

NAT'L INST. OF STAND & TECH
A11106 494930

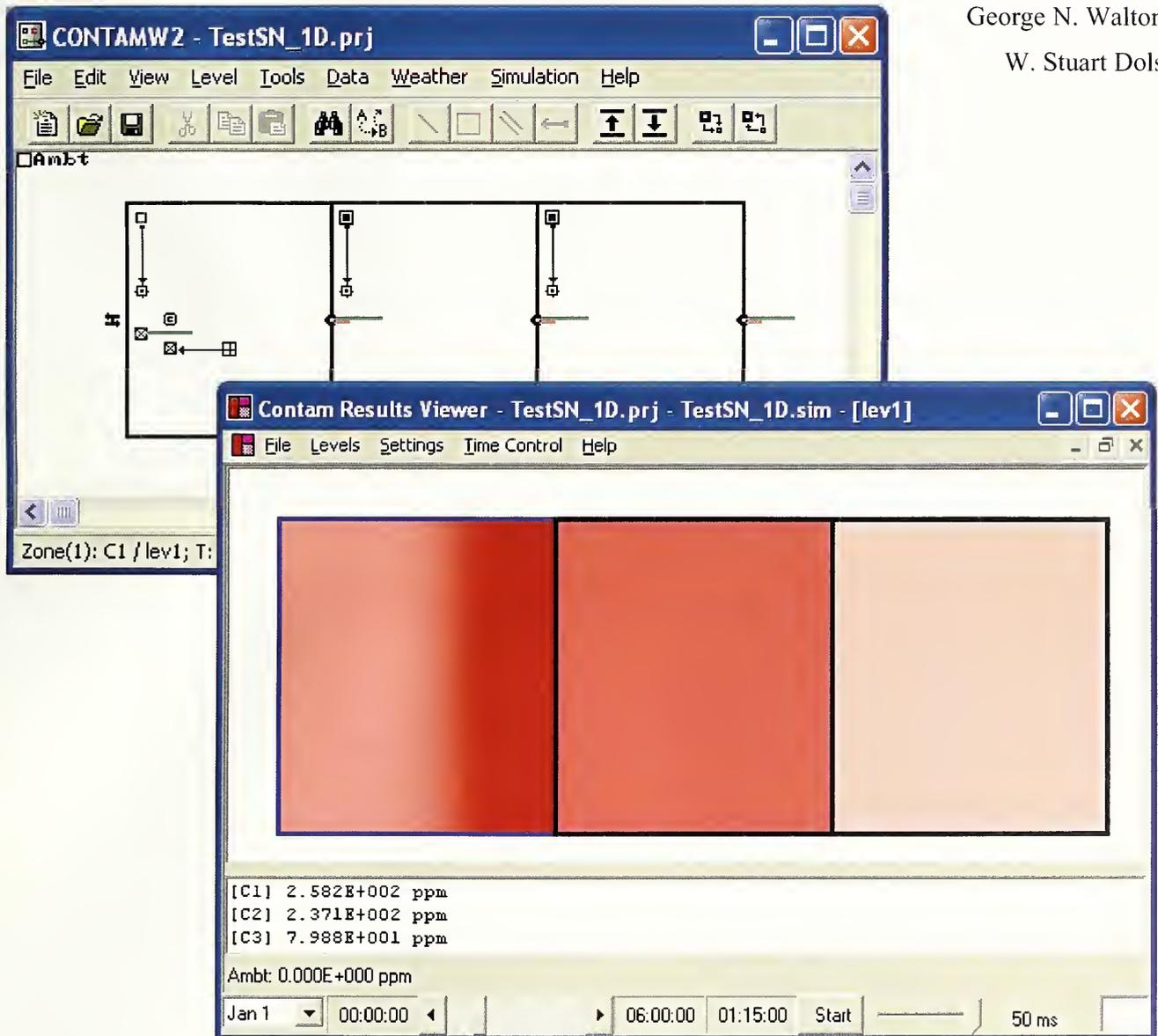
NIST
PUBLICATIONS

REFERENCE

NISTIR 7251

CONTAM 2.4 User Guide and Program Documentation

George N. Walton
W. Stuart Dols



QC
100
.456
#7251
2005

NISTIR 7251

**CONTAM 2.4 User Guide
and Program Documentation**

George N. Walton

W. Stuart Dols

*Building Environment Division
Building and Fire Research Laboratory
National Institute of Standards and Technology
Gaithersburg, MD 20899-8633*

Prepared for:
Naval Surface Warfare Center
Dahlgren, VA

October, 2005



U.S. Department of Commerce

Carlos M. Gutierrez, *Secretary*

Technology Administration

Michelle O'Neill, *Acting Under Secretary of Commerce for Technology*

National Institute of Standards and Technology

William A. Jeffrey, *Director*

1971

1972

1973

1974

1975

1976

1977

1978

1979

1980

1981

1982

1983

1984

1985

1986

1987

1988

1989

1990

Abstract

This manual describes the computer program CONTAM version 2.4 developed by NIST. CONTAM is a multizone indoor air quality and ventilation analysis program designed to help you determine: *airflows and pressures* – infiltration, exfiltration, and room-to-room airflows and pressure differences in building systems driven by mechanical means, wind pressures acting on the exterior of the building, and buoyancy effects induced by temperature differences between the building and the outside; *contaminant concentrations* – the dispersal of airborne contaminants transported by these airflows and transformed by a variety of processes including chemical and radio-chemical transformation, adsorption and desorption to building materials, filtration, and deposition to building surfaces; and/or *personal exposure* – the prediction of exposure of building occupants to airborne contaminants for eventual risk assessment.

CONTAM can be useful in a variety of applications. Its ability to calculate building airflows and relative pressures between zones of the building is useful for assessing the adequacy of ventilation rates in a building, for determining the variation in ventilation rates over time, for determining the distribution of ventilation air within a building, and for estimating the impact of envelope air-tightening efforts on infiltration rates. The program has also been used extensively for the design and analysis of smoke management systems. The prediction of contaminant concentrations can be used to determine the indoor air quality performance of buildings before they are constructed and occupied, to investigate the impacts of various design decisions related to ventilation system design and building material selection, to evaluate indoor air quality control technologies, and to assess the indoor air quality performance of existing buildings. Predicted contaminant concentrations can also be used to estimate personal exposure based on occupancy patterns.

Version 2.0 contained several new features including: non-trace contaminants, unlimited number of contaminants, contaminant-related libraries, separate weather and ambient contaminant files, building controls, scheduled zone temperatures, improved solver to reduce simulation times and several user interface related features to improve usability. Version 2.1 introduced more new features including the ability to account for spatially varying external contaminants and wind pressures at the building envelope, more new control elements, particle-specific contaminant properties, total mass released calculations and detailed program documentation. Version 2.4 introduces two new deposition sink models, a one-dimensional convection/diffusion contaminant model for ducts and user-selectable zones, new contaminant filter models, control super nodes, super filters and super airflow elements, a duct balancing tool, building pressurization and model validity tests and several other usability enhancements. Versions 2.2 and 2.3 were interim versions that were not released to the public.

Key Words: airflow analysis; building controls; building technology; computer program; contaminant dispersal; controls; indoor air quality; multizone analysis; smoke control; smoke management; ventilation

Software Disclaimer

This software was developed at the National Institute of Standards and Technology by employees of the Federal Government in the course of their official duties. Pursuant to title 17 Section 105 of the United States Code this software is not subject to copyright protection and is in the public domain. CONTAM is an experimental system. NIST assumes no responsibility whatsoever for its use by other parties, and makes no guarantees, expressed or implied, about its quality, reliability, or any other characteristic. We would appreciate acknowledgement if the software is used.

This software can be redistributed and/or modified freely provided that any derivative works bear some notice that they are derived from it, and any modified versions bear some notice that they have been modified.

Certain trade names or company products are mentioned in the text to specify adequately the experimental procedure and equipment used. In no case does such identification imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the equipment is the best available for the purpose.

Acknowledgements

Much of the work that went into developing this version of CONTAM was sponsored by the Naval Surface Warfare Center Dahlgren Division under Military Interdepartmental Purchase Request # N00178-05-MP-00139. The authors appreciate the interest and input of Kathrina Urann, Matthew Wolski and the entire IBTK team throughout this project.

The authors would like to acknowledge the programming skills of Brian Polidoro for his work on the CONTAM user interface, SimReadW and ContamRV, as well as his efforts in developing and maintaining the NIST IAQAnalysis Website. We would further like to acknowledge the testing of the beta versions of the software by the following people: NIST student interns Ryan Berke, Jacob Weber, Ivette Morazzani, and David Heinzerling; Steven Strege and Terry Fay of Hughes Associates Inc.; James Odasso and Kevin Good of Battelle Columbus; and Dr. Fred Loquasto III of Toyon Research Corporation. We also appreciate the continued collaboration with John Goforth and Mike Mercer of Lawrence Livermore National Laboratory in developing the socket communication capabilities of CONTAM.

Table of Contents

Abstract	iii
Software Disclaimer	iv
Acknowledgements	v
1 Introduction	1
1.1 What is CONTAM?.....	1
1.2 What's New in Version 2.4?	1
1.3 Enhancements for Version 2.1	3
1.4 Enhancements for Version 2.0	3
1.5 System Requirements	4
2 Getting Started	5
2.1 Installing CONTAM.....	5
2.2 Running CONTAM.....	6
2.3 User Tasks.....	6
2.4 The ContamW Graphical User Interface	9
2.5 Components of a CONTAM Project	11
3 Using CONTAM	14
3.1 Working with the SketchPad.....	14
3.2 Working with Project Files	24
3.3 Configuring ContamW	26
3.4 Working with Walls	29
3.5 Working with Levels.....	30
3.6 Working with Zones	33
3.7 Working with Airflow Paths	37
3.8 Working with Simple Air Handling Systems	55
3.9 Working with Ducts.....	62
3.10 Working with Controls.....	78
3.11 Working with Species and Contaminants.....	97
3.12 Working with Sources and Sinks	103
3.13 Working with Filters	111
3.14 Working with Kinetic Reactions	116
3.15 Working with Occupant Exposure	120
3.16 Working with Data and Libraries.....	123
3.17 Working with Weather and Wind	128
3.18 Working with WPC Files	136
3.19 Working with Schedules	145
3.20 Working with Simulations	149
3.21 Working with Simulation Results	161
3.22 Working with Project Annotations.....	191
3.23 Getting Help.....	192
4 Special Applications of CONTAM	193
4.1 Building Pressurization Test.....	193
4.2 Smoke Control Systems	194
4.3 Shafts.....	194
5 Theoretical Background	196
5.1 Model Assumptions	196
5.2 Contaminant Analysis.....	198
5.3 Airflow Analysis	207
6 References	224
Appendix A PRJ File Format	227
Appendix B ContamX Program Documentation	268
Appendix C ContamW Program Documentation	286
Appendix D SketchPad Arrays	299

1 Introduction

1.1 What is CONTAM?

CONTAM is a multizone indoor air quality and ventilation analysis computer program designed to help you determine:

- (a) *airflows*: infiltration, exfiltration, and room-to-room airflows (and pressure differences) in building systems driven by mechanical means, wind pressures acting on the exterior of the building, and buoyancy effects induced by the indoor and outdoor air temperature difference.
- (b) *contaminant concentrations*: the dispersal of airborne contaminants transported by these airflows; transformed by a variety of processes including chemical and radio-chemical transformation, adsorption and desorption to building materials, filtration, and deposition to building surfaces, etc.; and generated by a variety of source mechanisms, and/or
- (c) *personal exposure*: the predictions of exposure of occupants to airborne contaminants for eventual risk assessment.

CONTAM can be useful in a variety of applications. Its ability to calculate building airflows and interzone pressure differences is useful to assess the adequacy of ventilation rates in a building, to determine the variation in ventilation rates over time and the distribution of ventilation air within a building, to estimate the impact of envelope air tightening efforts on infiltration rates, and to design and assess smoke management systems. The prediction of contaminant concentrations can be used to determine the indoor air quality performance of a building before it is constructed and occupied, to investigate the impacts of various design decisions related to ventilation system design and building material selection, and to assess the indoor air quality performance of an existing building. Predicted contaminant concentrations can also be used to estimate personal exposure based on occupancy patterns in the building being studied. Exposure estimates can be compared for different assumptions of ventilation rates and source strengths.

This document addresses both the graphical user interface (referred to herein as **ContamW**) and the numerical solver (referred to herein as **ContamX**) of version 2.4 of the program, collectively referred to as **CONTAM**.

1.2 What's New in Version 2.4?

CONTAM 2.4 is an update to the previously released version 2.1 (versions 2.2 and 2.3 were interim versions not released to the public) with many enhancements, several of which are listed below. CONTAM 2.4 is backwards compatible with previous versions, meaning you can open existing project files created with previous versions. Once updated to version 2.4, project files can not be saved as or opened with previous versions. Therefore, you should maintain older copies of your existing project files.

Throughout this manual you will find new features of the program highlighted with either solid grey lines in the margins (printed version) or in a different color text (on-line version) as illustrated by this paragraph.

- Number of building components – The limit on the total number of each type of building component (i.e., zones, paths, ducts, sources/sinks, etc.) that can be created within a single project has been increased from 32 767 to 2 147 483 647.

- Short time-step method – A new simulation mode that solves contaminant dispersal equations using an explicit solution method.
 - o 1D Zones – The ability to simulate selected zones as one-dimensional convection/diffusion zones has been implemented. This provides the ability to more realistically simulate the transport of contaminants through “long” or “tall” zones, e.g., hallways and shafts, better accounting for transport times through these 1D zones.
 - o 1D duct model – Implementation of one-dimensional convection/diffusion elements in ductwork, thus providing more realistic contaminant transport results through the duct system. This requires the use of the short time-step method.
 - o Duct temperature calculations – When simulating 1D ducts in the short time-step mode, CONTAM can now calculate duct temperatures based on the "mixing" of air streams at the duct junctions and the source (zone) air temperatures from which air is introduced into the duct system.
- Simulation results data – Modified the zone and junction pressures as reported to the simulation results file to more closely resemble gage pressures. Zone pressures are now referenced to the ambient pressure at the level on which the zone is located. Junction static pressure is now referenced to the zone in which the junction is located at the height of the junction within the zone.
- Automated duct balancing – Simulation option to automatically balance duct systems. This will greatly simplify the task of defining detailed duct systems in CONTAM.
- Building model verification tests
 - o Building pressurization test – Simulation option to automatically determine building envelope airtightness based on a simulated fan pressurization test.
 - o Building airflow test – Simulation option that generates a set of data, mostly related to building ventilation, that can be used to gauge the reasonableness of model inputs before beginning analysis of a building.
- Contaminant Filters
 - o New filter models – CONTAM has added particle and gaseous filter models that greatly increases the user’s flexibility to create models based on measured filter performance data, e.g., MERV and breakthrough curves.
 - o Filter super elements – enables multiple filter models to be combined into a single filter element, e.g., a particle pre-filter combined with a gaseous filter.
- Contaminant summary file – This file is generated when simulating contaminants to provide information related to source/sink contaminant generation and removal, filter loading and breakthrough, and contaminant transport between building zones and ambient.
- Deposition sink models – Added Deposition Velocity and Deposition Rate sink models to simplify the definition of sinks based on more familiar deposition terminology.
- Control super elements – The task of creating building control systems has been improved by reducing the amount of user input required, increasing the amount of flexibility in defining control systems and enabling the sharing of control elements, e.g., sensors, within and between projects.
- TCP/IP socket simulation control – ContamX can be controlled via the TCP/IP Socket communication protocol based on a pre-defined set of messages that include control

commands and data exchange. ContamX can also be compiled and run under UNIX-based operating systems.

- Usability enhancements
 - o Find – search for items on SketchPad by Name and Number
 - o Find-and-replace – search for and replace properties of building components within a project
 - o SketchPad movement
 - Ctrl + arrow key => skip 10 cells in direction of arrow
 - Shift + arrow key => move to next icon in the direction of arrow
 - Mouse wheel for scrolling:
 - Wheel => up/down Shift+Wheel => page up/down
 - Alt+Wheel => left/right Shift+Alt+Wheel => page left/right
 - o Keyboard icon placement – use keyboard to place icons on the SketchPad
 - o Floating status bar
 - o User-defined zone colors
 - o User-defined duct colors and automated duct tracing/coloring tool
 - o Generate bitmap file (.bmp) of SketchPad

1.3 Enhancements for Version 2.1

Enhancements from version 2.0 to 2.1

- Wind pressure and contaminant fields – The ability to incorporate data from exterior airflow and pollutant transport models, e.g., plume and puff dispersion models, to utilize detailed ambient wind pressure and contaminant data fields to provide boundary conditions on the airflow paths of the envelope of built structures
- Control elements – New control elements to simulate time delays associated with spin-up/down of fans and the opening/closing of dampers and to perform integration, peak determination of sensor output over time, maximum, minimum and exponential operations
- Particle analysis – Modified contaminant properties to simplify the analysis of airborne particles
- Mass release calculation – The calculation of total mass released by contaminant sources during a simulation
- Program Documentation – Programming documentation of the software was produced which can be obtained from the NIST website. This documentation includes details of CONTAM input and output file formats [Walton and Dols 2003].

1.4 Enhancements for Version 2.0

Enhancements from version 1.0 to 2.0

- Building controls – Controls include sensors, actuators, modifiers and links. Control actuators can be used to modify various characteristics of building components based on control signals obtained from sensors and even modified by signal modifiers. For example, a sensor can be used to obtain a contaminant concentration within a zone, and a proportional control actuator can be used to adjust supply airflow into the zone based on the sensed concentration.

- Scheduled zone temperatures – Zone temperatures can now be varied through the use of user-defined schedules. This allows for the change in zone pressures due simply to the change in temperature within the zone according to the ideal gas relationship.
- Contaminants
 - o Non-trace contaminants – You can now account for the impact of contaminant concentrations on the density of the air, e.g., water vapor.
 - o Unlimited number of contaminants – CONTAM no longer restricts the number of contaminants you can simulate. The previous limitation was 10.
 - o Contaminant-related libraries – Contaminant related elements can now be shared through CONTAM library files. These elements include contaminant species, filters, source/sinks and kinetic reactions.
- Solver
 - o Variable air density – CONTAM now provides the ability to simulate non-flow related processes that can lead to the accumulation/reduction of mass within building zones, e.g., due to non-trace contaminant sources and to variations in the zone pressure due to the change in zone temperature.
 - o Improved numerical solver implementing sparse matrix techniques to greatly reduce transient simulation times for large problems.
 - o Separated solver from graphical user interface to provide for batch execution of simulations and directly utilize .PRJ files.
- Transient weather
 - o Separate transient weather and contaminant files – Weather files (.WTH) no longer contain contaminant concentrations (except for outdoor humidity ratio). This means you don't have to create different weather files depending on the types of contaminants you are simulating. CONTAM now provides you with the option of simulating transient ambient contaminant concentrations using a contaminant file (.CTM).
 - o Weather file creation/conversion software – NIST has developed a software tool that allows you to convert existing weather files to CONTAM 2.0 compatible weather files. You can convert your existing 1.0-compatible files, TMY2 and EnergyPlus weather files.
- User interface
 - o Longer zone names – Zone names can now be up to 15 characters long.
 - o SketchPad zooming feature – You can now reduce the icon size of the SketchPad to allow the display of larger projects on the screen.
 - o Display of net inter-zonal airflow results for highlighted zones
 - o Distinct simple air-handling system zones – The implicit zones of multiple simple air-handling system are now distinguished from each other to allow for the plotting of individual system zones.
 - o Airflow direction indicators are now displayed in the Status Bar when viewing airflow path results.

1.5 System Requirements

CONTAM runs under Windows 98, NT/2000, and XP.

2 Getting Started

2.1 Installing CONTAM

□ Obtaining CONTAM

CONTAM installs from a set of installation files that you can obtain from NIST. These files can be downloaded from the NIST website (www.bfrl.nist.gov/IAQanalysis).

□ Installing CONTAM 2.4

After downloading CONTAM 2.4 from the NIST website, double-click the Microsoft installer file "contam24.msi" to begin the installation process. Choose a folder into which you want to install the program, or simply select the default location which is "c:\Program Files\NIST\CONTAM 2.4." Follow the directions to complete the installation. This installation will not remove or overwrite existing versions of CONTAM.

□ Files Installed

The following table lists the files installed by the setup program. For each file, the directory to which it is installed, the name and a brief description are given. The <program> directory is selected when you install the program. The directory is that of the operating system fonts. The default is <windows>/FONTS, where <windows> depends on the operating system you are using, e.g., Windows XP <windows> = WINDOWS, Windows 2000 <windows> = WINNT and Windows 95/98 <windows> = WINDOWS.

Directory	File Name	Description
<program>	contamw2.exe	ContamW – User interface
	contamx2.exe	ContamX – Solver
	contam.cfg	Configuration file
	cwhelp23.hlp cwhelp23.cnt	Help files
	olch2d32.dll roboex32.dll	Charting and help display dynamic link libraries
<program>\samples	*.prj, *.wth and *.ctm	Sample project files
	*.lb?	CONTAM library files
	walton##.fnt	SketchPad fonts where ## ranges from 01 to 16 for different SketchPad resolutions

□ Uninstalling CONTAM

The CONTAM 2.4 setup program also provides you with an uninstall feature. You uninstall CONTAM 2.4 much as you would a typical Windows program. Access the **Control Panel** from the **Settings** selection of the **Start** menu. Select Add/Remove Programs from the Control Panel. Select CONTAM 2.4 from the list of installed programs and click the "**Add/Remove...**" button to uninstall CONTAM 2.4.

2.2 Running CONTAM

As indicated in the Installing CONTAM section, CONTAM actually consists of two executable programs: ContamW and ContamX. ContamW is the graphical user interface and ContamX is the simulation engine. Typically you will activate ContamX via a menu command from within ContamW. However, ContamX can be utilized as a command-line tool and even utilized within script or batch files.

❑ Running ContamW

Run ContamW by selecting CONTAM 2.4 from the NIST program group of the Start menu.

❑ Running ContamX as a Command-line Program

The easiest way to run ContamX as a command-line program is by locating the `contamx2.exe` file and related projects and supporting files within a common directory. The command-line format for running ContamX is as follows:

```
CONTAMX2 <project filename>
```

If you leave off the project file name, you will be prompted to enter one, therefore when running in batch mode be sure to provide a filename.

2.3 User Tasks

The use of CONTAM to analyze airflow or contaminant migration in a building involves five distinct tasks:

1. *Building Idealization*: Form an *idealization* or specific *model* of the building being considered,
2. *Schematic Representation*: Develop a schematic representation of the idealized building using the ContamW *SketchPad* to draw the building components,
3. *Data Entry*: Collect and input data associated with each of the building components represented on the *SketchPad*,
4. *Simulation*: Select the type of analysis you wish to conduct, set simulation parameters, and execute the simulation,
5. *Review & Record Results*: Review the results of your simulation and record selected portions of the results.

❑ Task 1 - Building Idealization

Building idealization refers to the simplification of a building into a set of zones that are relevant to the user's goal in performing an analysis. A building can be idealized in a number of ways depending on the building layout, the ventilation system configuration and the problem of interest. This idealization phase of analysis requires some engineering knowledge on the part of the user and is an acquired skill that you can develop through experience in airflow and indoor air quality analysis and by becoming familiar with the theoretical principles and details upon which indoor air quality analysis is based.

It is important to note that CONTAM provides a macroscopic model of a building. In this macroscopic view, each zone is considered to be well-mixed. Well-mixed means that a zone is characterized by a discrete set of state variables, i.e., temperature, pressure and contaminant concentrations. Temperature and contaminant concentration do not vary spatially within a zone, and contaminants mix instantly throughout well-mixed zones.

However, pressure does vary hydrostatically within all zones.

Beginning with CONTAM version 2.4, one-dimensional convection/diffusion zones can be implemented within CONTAM. This feature can be useful in simulating contaminant transport through long or tall zones that are characterized by a single, dominant flow direction. One dimensional convection/diffusion duct models can also be implemented to more realistically capture contaminant transport within entire duct systems.

CONTAM is well suited for analyzing the interaction between the zones of a building on a macroscopic level but is not well suited for the analysis of the microscopic airflow and contaminant characteristics within a given zone of a building. Computational Fluid Dynamics (CFD) analysis is better suited for analyzing the airflow and contaminant transport characteristics of a given zone of a building. However, the computational resources required to perform a CFD analysis for an entire building is currently prohibitive. The one-dimensional convection/diffusion model provides an intermediate level of detail between the well-mixed and CFD models.

□ **Task 2 - SketchPad Representation**

Developing the SketchPad representation will be the focus of your interaction with ContamW. With ContamW's SketchPad you will be able to draw a diagram – a *SketchPad diagram* – of your building idealization using drawing tools and libraries of icons to represent components of the building system. CONTAM translates your diagram into a system of equations that will then be used to model the behavior of the building when you perform a simulation.

See Working with the SketchPad in the Using CONTAMW section.

□ **Task 3 - Data Entry**

Data entry can be one of the more time-consuming parts of the process of using ContamW. It involves the determination and input of the numerical values of the parameters associated with each of the SketchPad icons. These icons represent the components of the building model and include air leakage paths (windows, doors, cracks), ventilation system components (fans, ducts, vents), contaminant sources, filters, and sinks and control network components. Each of these building components is associated with a number of parameters, and you must obtain the values of these parameters for entry into the building model. Depending on the component and the application, these values can be obtained from building-specific data, engineering handbooks, and product literature. In many cases, a degree of engineering judgment will be involved. ContamW allows you to create libraries of data elements that describe the common features of building components that you can share between different building models.

Detailed information is provided for the various components throughout the Using CONTAMW section.

□ **Task 4 - Simulation**

Simulation is the use of CONTAM to solve the system of equations assembled from your SketchPad representation of a building to predict the airflow and contaminant concentrations of interest. This step involves determining the type of analysis that is needed; steady- state, transient or cyclical, and a number of other simulation parameters. These parameters depend on the type of analysis you wish to perform (steady state or transient), and include

convergence criteria and in the case of a transient analysis, time steps and the duration of the analysis.

See Working with Simulations in the Using CONTAMW section.

□ **Task 5 - Review & Record Results**

ContamW allows you to view the simulation results on the screen and to output them to files for input to a spreadsheet program or other data analysis programs including those developed by NIST and available on the CONTAM website (e.g., **SimRead** and **ContamRV**). Airflows and pressure differences at each flow element can be viewed directly on the SketchPad. Contaminant concentrations for a zone can also be plotted as a function of time directly from the SketchPad. You can then decide which data you wish to examine more closely and export these to a tab-delimited text file that can then be imported into a spreadsheet for further analysis. There is also a controls-related feature that provide the ability to report the values of user-selected control nodes to a control "log" file for each time step of a transient simulation.

See Working with Results in the Using CONTAMW section.

2.4 The ContamW Graphical User Interface

The graphical user interface (GUI) of CONTAM is what you use to create and view your airflow and contaminant dispersal analysis projects. It consists mainly of a drawing region referred to as the SketchPad, a set of drawing tools, a title bar, a set of menus, and a status bar. The following sections provide brief explanations of each of the features of the CONTAM GUI. See the Using ContamW section for details on how to use these features.

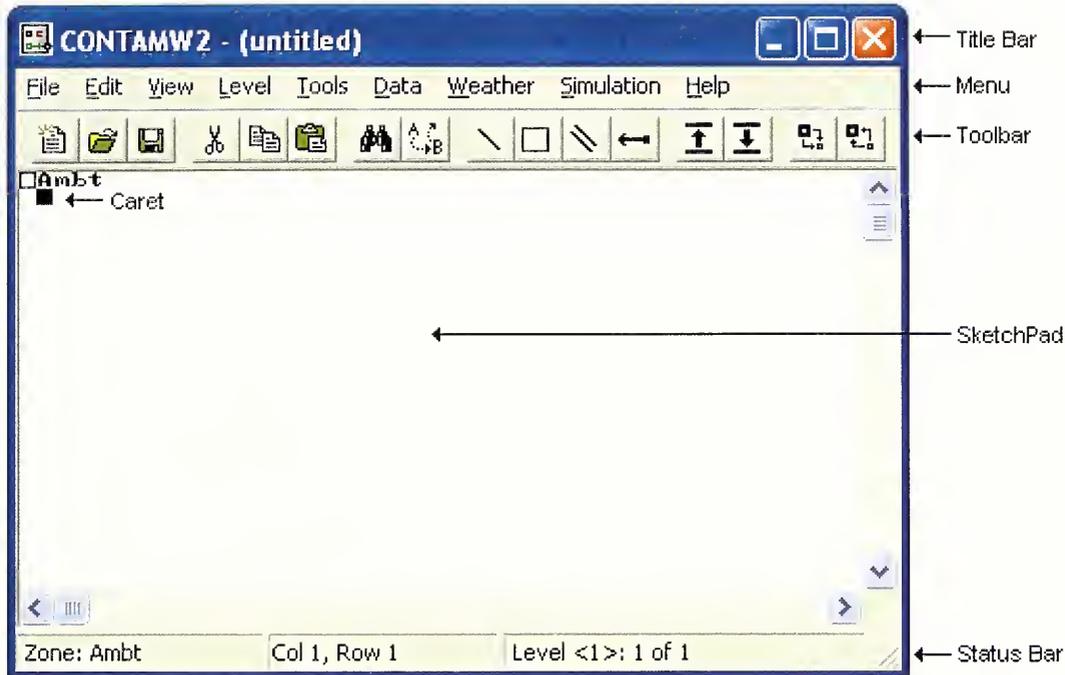


Figure – The CONTAM Graphical User Interface

2.4.1 SketchPad

The SketchPad is the region of the ContamW screen where you draw the schematic representation of a building you wish to analyze. This representation is in the form of a set of simplified floor plans that represent the levels of a building. The SketchPad is used to establish the geometric relationships of the relevant building features and is not intended to produce a scale drawing of a building. It should be used to create a simplified model where the walls, zones, and airflow paths are topologically similar to the actual building (See Working with the SketchPad).

2.4.2 Title Bar

The title bar is the typical rectangular region at the top of the main ContamW window. The CONTAM project filename will be displayed within this region.

2.4.3 Menu

The menu is typical of a Windows program with differences that provide functionality specific to the CONTAM application. It is through this menu that most ContamW operations can be performed including: saving and retrieving project files, selecting various modes of display, setting up and performing simulations, as well as accessing the on-line help system. Note that some of the menu items have shortcuts or hot-keys that enable quick access; for example, to save the current project file use the Ctrl+S key combination.

2.4.4 Toolbar

The toolbar, shown in the following figure, appears below the menu and provides convenient shortcuts to some of the menu items. Several of the toolbar buttons are similar to those found in other Windows applications. Other buttons provide a shortcut to functionality specific to ContamW.



2.4.5 Status Bar

The status bar, shown in the following figures, is the region displayed below the SketchPad at the bottom of the main window. This region is broken up into three separate panes that display various information depending on the current mode of the SketchPad.

ContamW 2.4 now provides you with a *floating status bar* that will display in the region of the caret whenever you select certain icons on the screen providing a more convenient means to review icon information.. This is similar to Tool Tips that appear when you hover over a toolbar button with the Windows cursor. You can turn this feature on and off via the **View → Floating Status Bar** or **Ctrl+T** for Tool tips.

□ Left Pane

This pane always displays the type and number of building component icon, e.g., zone, path, air handling system, etc.

In the *normal mode* of operation, the leftmost pane displays summary information of the currently highlighted cell or icon.

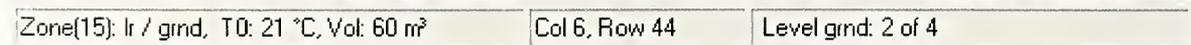
In the simulation *results mode*, the leftmost pane displays the results for the currently highlighted icon. For a zone this includes the temperature and pressure relative to the ambient pressure at the elevation of the level on which the zone is located. For duct junctions and terminals it displays the temperature and static pressure relative to the zone in which the junction or terminal is located. For paths and duct segments it will display the airflow and pressure drop across the path or segment along with symbols to indicate the direction of flow (>, <, ^, v). For a simple air handling system icon the outdoor airflow, recirculation airflow and exhaust airflows for the implicit flow paths will be displayed.

□ Center Pane

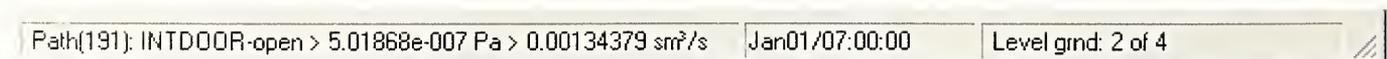
In the *normal mode* of operation, the center pane indicates the location of the currently highlighted icon or cell in SketchPad coordinates (numbered from the top-left corner). In the simulation *results mode* the center pane displays the current simulation time step for which the results are being displayed.

□ Right Pane

The rightmost pane displays the name and number of the current level and the total number of levels in the project.



Status Bar during Normal Mode of operation



Status Bar displaying airflow path results during Results Viewing Mode of operation

2.5 Components of a CONTAM Project

2.5.1 Project Files

All data related to the characteristics of the projects you work with are stored in a "project" file having a "PRJ" extension (See Working with Project Files). This is an ASCII file, which is intended to be "viewed" only by the ContamW program. You should keep careful records of your project files and establish a naming convention that is meaningful to you for the various versions of a project that you may wish to save.

There are other files utilized by CONTAM including: simulation results files, weather files, library files, and the log file. Simulation results are stored in files created automatically by CONTAM with the same name as the PRJ file except that the "PRJ" extension is replaced by the "SIM," "SUM," "EXP," "CSM," "ACH" and "AGE" extensions depending on the type of results generated by the simulation (see Working with Simulation Results). Weather files and contaminant files, typically given the WTH and CTM extension, respectively, may contain up to a years worth of weather and outdoor contaminant data and are used when performing transient simulations (see Working with Weather). Weather and contaminant files are ASCII files, but the data is of a format unique to CONTAM. Library files are the means by which you can share various types of data between CONTAM projects. Each type of library data has a different extension: LB0 (contaminants and source/sinks), LB1 (schedules), LB2 (wind pressure profiles), LB3 (airflow elements), LB4 (duct flow elements) or LB5 (control super elements). You create libraries of data using ContamW (see Working with Data and Libraries). The log file, "CONTAMW2.LOG," is created in the directory in which the executable program resides each time you run the program. This file keeps track of operations that you perform during your session with ContamW and is a useful tool in the event that you require technical support from the program developers (see Getting Help).

2.5.2 Building Components

Building components are the items that characterize the physical makeup of a building that you define using ContamW. This section briefly describes these components.

□ Levels

CONTAM represents buildings in terms of multiple levels, accounting for the communication of air and contaminants between these levels. Levels typically correspond to floors of a building, but a suspended ceiling acting as a return air plenum or a raised floor acting as a supply plenum may also be treated as a level.

□ Walls

Walls are used to designate zones which are regions surrounded by walls, floor and ceiling. These walls include the building envelope and internal partitions with a significant resistance to airflow.

□ Floors and Ceilings

Floors and ceilings are included implicitly by ContamW for building zones. When you draw a zone on the SketchPad, ContamW automatically includes the floor of the zone. To create a roof with penetrations into the floor below requires a blank level above the top floor. It is also possible to create a phantom zone with no floor or ceiling as might be required to create an atrium that spans multiple levels (see Working with Zones).

□ Zones

In CONTAM, a zone typically refers to a volume of air having a uniform temperature and contaminant concentration. However, beginning with version 2.4, zones can now be configured as one-dimensional *convection/diffusion zones* in which contaminant concentrations are allowed to vary along a user-specified axis.

There are three types of zones in CONTAM: *normal*, *phantom* and *ambient*. *Normal zones* are separated from the zone below by a floor. The *ambient zone*, which surrounds the building is implicitly defined and is identified by the symbol at the upper-left corner of the SketchPad. You can also use an *ambient zone* icon to define a courtyard. *Phantom zones* indicate that the area on the current level is actually part of the *normal zone* on the level immediately below. There is no floor between a *phantom zone* and the *normal zone* below. You could use *phantom zones* to define building features such as atriums. Only *normal zones* can be configured to be *convection/diffusion zones*, however, these zones can still be referenced by *phantom zones*.

□ Airflow Paths

An airflow path indicates some building feature by which air can move from one zone to another. Such features include cracks in the building envelope, open doorways, and fans. Path symbols placed on the walls are used to represent openings between zones or to ambient; any other placement represents an opening in the floor to the zone on the level below. CONTAM can implement several different models or airflow elements to define airflow paths. The basic categories of airflow elements or models are as follows: small and large crack/openings represented by power-law and quadratic pressure relationships, small and large doorways elements, and fan/forced airflow elements. (See Working with Airflow Paths)

□ Simple Air-handling Systems

The simple air-handling system (AHS) provides a simple means of introducing an air-handling system into a building without having to draw a duct system. It provides a reasonable model of an air-handling system that delivers user-specified flows where the system is properly balanced and the fan is not impacted by any other pressurizing effects in the building. The AHS consists of two implicit airflow nodes (return and supply), three implicit flow paths (recirculation, outdoor, and exhaust air), and multiple supply and return points that you place within the zones of the building. You can set the air flows of the AHS to vary according to a schedule.

□ Ducts

You can use ducts to implement a more detailed model of an air-handling system that handles a broader range of conditions. For example, when an air handler is off, the ductwork may provide flow paths between zones which are significant in relation to the normal construction cracks or openings. Ductwork consists of duct segments (paths) and junctions or terminal points (nodes). CONTAM can implement several different duct segment models or duct flow elements to define duct segments. The basic categories of duct flow elements are as follows: resistance models, fan performance curves, and back-draft dampers. (See Working with Ducts)

❑ **Contaminants, Sources and Sinks**

You can define an unlimited number of contaminants within a single project with a practically limitless number of sources associated with the contaminants. CONTAM can simulate contaminant transport via airflow between zones, removal by filtration mechanisms associated with flow paths, and removal and addition by chemical reaction. CONTAM can also implement several source and sink models to generate contaminants within or remove contaminants from a zone. These models include: constant generation, pressure driven, decaying source, cutoff concentration, reversible boundary layer diffusion, and burst models. (See *Working with Contaminants* and *Working with Sources and Sinks*)

❑ **Schedules**

Schedules are used to control or modify various properties of building components as a function of time. You can set schedules for airflow paths, duct flow paths; contaminant sources and sinks; and inlets, outlets and outdoor air delivery of simple air-handling systems. The effect of setting a schedule on a building component varies depending on the properties of the component. For example, you can set a schedule to adjust the airflow delivered to a zone by an inlet of a simple air-handling system. (See *Working with Simple Air Handling Systems*) CONTAM also provides the ability to schedule zone temperatures.

❑ **Controls**

Controls include sensors, actuators, modifiers and links. Control actuators can be used to modify various characteristics of building components based on control signals obtained from sensors and even modified by signal modifiers. For example, a sensor can be used to obtain a contaminant concentration within a zone, and a proportional control actuator can be used to adjust supply airflow into the zone based on the sensed concentration.

2.5.3 Occupants

Occupants can be used to determine the amount of contaminant exposure a person would be subjected to within a building. Occupants can also generate contaminants. You can set a schedule to establish each occupant's movement within a building. Occupant schedules can also be used to define periods of times when occupants are not in the building. (See *Working with Occupant Exposure*)

2.5.4 Weather

CONTAM enables you to account for either steady-state or varying weather conditions. Weather conditions consist of ambient temperature, barometric pressure, humidity ratio, wind speed and direction, as well as ambient contaminant concentrations.

2.5.5 Simulation

In CONTAM, simulation is the process of forming a set of simultaneous equations based upon the information stored in the project file, performing the numerical analysis to solve the set of nodal equations according to user-defined specifications, and creating simulation results files that can be viewed using the ContamW interface. There are three basic types of simulations that you can perform for airflow and contaminant analysis using CONTAM: steady state, transient and cyclical. (see *Working with Simulations*)

3 Using CONTAM

This section provides detailed information on how to use the features of the ContamW application as well as a detailed explanation of the terminology of the user interface. You should think of this section as your detailed conceptual and contextual guide to working with the ContamW program.

3.1 Working with the SketchPad

The SketchPad is the region of the ContamW screen where you draw the schematic representation of a building you wish to analyze. This representation is in the form of a set of simplified floor plans that represent the levels of a building. The SketchPad consists of an invisible array of cells into which you place various icons to form your schematics of a building. The SketchPad is used to establish the geometric relationships of the relevant building features and is not intended to produce a scale drawing of a building. It should be used to create a simplified model where the walls, zones, and airflow paths are topologically similar to the actual building.

□ The Caret

When working with ContamW you will notice a blinking square on the SketchPad. This is known as the system *caret*, and it is the size of a single SketchPad cell. This *caret* is the same thing as the blinking vertical bar that is common to many word processing applications. The *caret* indicates the currently selected cell of the SketchPad. Any icon-related information that appears in the status bar is associated with the location of the *caret*.

□ Moving the Caret

To move the *caret* around the SketchPad you can use the keyboard arrow keys or you can move the system cursor with the mouse and click the *left mouse button* to set the caret position. You can also use the arrow keys in conjunction with the **Shift** and **Ctrl** keys to move the *caret* more quickly.

Shift + Arrow Key moves the *caret* 10 cells in the direction of the arrow. The number of cells to move can be modified within the ContamW Configuration (See Cell/Icon Size in the Configuring ContamW section).

Ctrl + Arrow Key moves the *caret* to the next icon in the direction of the arrow

□ SketchPad Operations

The specific operations that you will perform using the SketchPad are as follows:

1. Drawing Walls, Ducts and Controls
2. Drawing building component icons
3. Defining building component icons
4. Viewing results
5. Viewing envelope pressure differentials due to wind effects

□ SketchPad Modes

A SketchPad mode basically refers to the type of information that is displayed upon the SketchPad. There are three basic modes of the SketchPad: *normal*, *results* and *wind pressure*.

You can tell what mode the program is in by looking at the items in the **View** menu to see which ones are checked.

In the *normal* mode ContamW displays only the building component icons. In this mode you can add, delete, copy, and move icons.

In the *results* mode, ContamW displays simulation results upon the SketchPad. In this mode you will not be allowed to add, delete, copy and move icons upon the SketchPad (See Viewing Results).

The *wind pressure* mode is provided to verify wind speed and direction information visually on the SketchPad. (See Checking Wind)

□ Control Super Element / Super Node SketchPad

With the advent of the new *control super element*, comes another use of the SketchPad to create and edit *super elements* and *super nodes* (See Control Super Elements in the Working with Controls sections). When working with super elements only the controls drawing tool is enabled, along with the ability to define the control network icons. This SketchPad is activated via the **Data→Controls→Super Elements...** menu item. When the Super Element SketchPad is active, the upper left corner of the SketchPad will display "**Super Element:**" followed by the name of the super element currently displayed on the SketchPad. When working with super elements the "Control Super Elements" dialog box will also be displayed.

The Super Node SketchPad enables only the modification of existing control sub-node icons, so the control drawing tool will be disabled as will be the ability to delete control network icons. The Super Node SketchPad is activated by *instantiating* an existing *control super element* or double-clicking on a Super Node icon. When the Super Node SketchPad is active, the upper left corner of the SketchPad will display "**Super Node:**" followed by the name of the super node currently displayed on the SketchPad. When working with super nodes the "Super Node Data" dialog box will also be displayed (See Control Super Nodes).

□ Zooming the SketchPad

You can change the size of the cells in which icons appear on the SketchPad in order to zoom the floor plan in and out on the SketchPad. The amount of information that will appear on your screen is dependent on the resolution and the cell size. Zooming can be accomplished using the two zoom buttons on the toolbar and their associated keyboard shortcuts shown below or changing the Current Cell/Icon Size on the Cell/Icon Size page of the Project Configuration Properties dialog box accessible via the **View→Options...** menu selection (see Cell/Icon Size).



Keyboard Shortcuts:

Ctrl + PageUp to increase cell size

Ctrl + PageDown to decrease cell size

□ Printing SketchPad Images

You can obtain images of your SketchPad drawings to print or edit using the Windows print screen feature. To do this, size the ContamW window and press **Alt+PrintScrn** on the keyboard to copy the current window to the Windows clipboard. Then you can immediately paste the image into the desired program. For example, you can paste the image into the Windows Paint

program for editing or directly into a word processing program. You can then print the image from either of these programs.

❑ **Exporting .BMP SketchPad Files**

You can save a SketchPad image of the currently displayed level to a Windows bitmap (.BMP) file using the **File → Save SketchPad to .BMP File...** menu item. The Save As dialog box will appear allowing you to name the file. The file will contain the entire SketchPad based on the current cell/icon size and color settings.

3.1.1 Drawing Walls, Ducts and Controls

You use various tools to draw the physical features of the building such as walls and ducts. All drawing can be accomplished with either the mouse or the keyboard. When drawing, the following mouse and keyboard keys produce common behavior. The left mouse button (**LMB**) corresponds to the **↵**, or **Enter** key; the right mouse button (**RMB**) to the **Esc** key. Vertical motion of the cursor corresponds to the **↑** and **↓** cursor keys, and horizontal motion corresponds to the **←** and **→** cursor keys.

The basic steps for drawing walls and ducts are as follows:

1. Activate drawing tool,
2. Set the initial location of the object,
3. Draw the object,
4. Undo current drawing,
5. Finalize drawing the object, and
6. Deactivate drawing tool.

❑ **Activating a Drawing Tool**

There are four drawing tools available, two for drawing walls, one for drawing ducts and one for drawing control networks. To activate a drawing tool either select it from the toolbar, select it from the "Tools" menu, or use the shortcut associated with the menu item. The toolbar button of a selected drawing tool will remain depressed as long as that tool is active. The toolbar buttons and associated menu shortcuts are shown here.

 rectangle (or box) drawing tool (**Ctrl+B**)

 free-form wall drawing tool (**Ctrl+W**)

 duct drawing tool (**Ctrl+D**)

 controls drawing tool (**Ctrl+L**)

Once you have selected the tool, the drawing cursor will be displayed. The drawing cursor is a pink square the size of a single SketchPad cell. Initially, the cursor appears with a transparent center.

❑ **Setting the Initial Location of the Object**

To begin drawing the object, you set the initial location by first moving the drawing cursor either with the mouse or the keyboard arrow keys and then you either click the **LMB** or press **↵** on the keyboard. This will anchor the beginning of the object at the nearest valid SketchPad cell for the

type of object you are drawing. When you select a valid beginning location, the drawing cursor will become solid and you can begin drawing your object.

❑ **Drawing the Object**

After you anchor the beginning of the object, you simply use the mouse and/or keyboard arrow keys to draw the desired shape. While you are drawing, the cursor will be restricted to the SketchPad region and constrained to specific movements in order to maintain drawing within valid cells of the SketchPad. As you move the cursor, a dark line will appear on the SketchPad representing the shape of the object you are drawing.

You can also move more quickly using the **Shift + Arrow Key** and **Ctrl + Arrow Key** keyboard combinations.

Shift + Arrow Key moves the drawing cursor 10 cells in the direction of the arrow. The number of cells to move can be modified within the ContamW Configuration (See Cell/Icon Size in the Configuring ContamW section).

Ctrl + Arrow Key moves the drawing cursor to the next icon in the direction of the arrow which may be a wall, duct or control.

❑ **Undoing the Current Drawing**

Prior to finalizing the drawing of the object, you can undo your drawing. To undo what you have drawn, either single-click the **RMB** or press the **Esc** key. This will erase the thick dark line, but you will still be in the drawing mode. To begin drawing again, set the initial location again and continue drawing another object.

❑ **Finalizing the Drawing Object**

Once you are satisfied with your drawing, you finalize the object by either single-clicking the **LMB** or pressing the **↵** key. This will replace the cells that had the thick dark line with the icons appropriate for completing the type of object you are drawing. If you have attempted to draw an invalid object, you will be prompted with a dialog box containing a message indicating the type of error, and then you will be allowed to repair the drawn object.

❑ **Deactivating Drawing Tools**

When you are finished using a drawing tool you click the **RMB** or press **Esc** until your system cursor reappears. Also, selecting a different drawing tool will automatically deactivate the current tool if you do not currently have an object anchored.

3.1.2 Drawing Building Component Icons

A set of icons is used to represent various building components such as airflow paths (representing doors, windows, cracks, etc.), contaminant sources and sinks, occupants and air-handling systems. You *place icons on the SketchPad using the Right Mouse Button* to display a pop-up menu and then selecting the desired building component from the menu. You can also use the **Tools → Draw Icon** menu and associated keyboard shortcuts to place building icons that can be place with the pop-up menu.

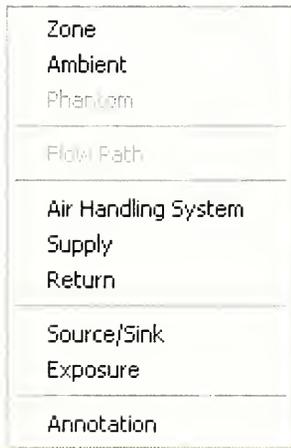
The icon placement menus are context-sensitive. The menu selections that you can choose from the icon placement menus will depend on the contents of the cell occupied by the caret and whether the SketchPad is currently displaying simulation results. *When results are being displayed, you will not be allowed to place additional icons on or delete icons from the SketchPad.* This is done to prevent the display of misleading results on the SketchPad due to a

mismatch in the number of icons on the SketchPad and the number of icons for which results are available from the last simulations. The icon placement menu will also prevent you from placing icons on invalid SketchPad locations. For example, you cannot place a supply or return of a simple air-handling system within the ambient zone.

Icon Placement Main Menu Commands:

Icon	Keyboard Shortcut
Zone	Z
Ambient	A
Phantom	P
Flow Path	F
Air Handling System	H
Supply (Inlet)	I
Return (Outlet)	O
Source/Sink	S
Exposure	X
Annotation	N

The following figure shows the pop-up icon placement menu.



The following table is a list of the icons that you will see on the ContamW SketchPad. This list shows the icons by categories of building components. Some of these icons are placed on the SketchPad using the drawing tools, and others are placed using the pop-up icon placement menu.

Icon Category	Component Icons
Walls	
Zones	
Duct Segments	
Duct Junctions	
Duct Terminals	
Simple AHS	
Airflow Paths	
Source/Sinks	
Occupants	
Controls	

Table – CONTAM SketchPad Icons (shown in default colors)

3.1.3 Defining Building Component Icons

When a new building component icon is drawn upon the SketchPad, it appears as a red icon. The color red indicates that the parameters of the component must still be defined. To define a building component, you double-click (**LMB**) on the icon or move the caret (represented by a black square) to the icon and press the **Enter** key. This will activate the associated dialog box or property sheet for the selected component, into which you enter the parameters that define the component. After you have entered the parameters of the component, click the **OK** button to accept the parameters or the **Cancel** button not to accept the parameters. The detailed properties of each component icon will be presented throughout the "Using ContamW" section of this manual. You can also access detailed descriptions of the component properties by pressing the **F1** key (Help) while running the ContamW application.

3.1.4 Deleting Building Component Icons

To delete undefined building component icons, you highlight the icon then delete it. You highlight the icon by either clicking (**LMB**) on the icon or moving the caret (represented by a black square) to the icon using the arrow keys. You delete the highlighted icon by either pressing the **Delete** key or using the menu command: **Edit → Delete**.

3.1.5 Find and Replace Building Component Data

ContamW provides the ability to perform both simple searches for building components and to perform find-and-replace operations. Both features involve the use of a dialog box that can be displayed simultaneously with the SketchPad to allow interaction between both the SketchPad and the dialog boxes.

□ Find

Use the Find feature to locate various building component icons on the SketchPad by either their *Number* or *Name* properties. Be aware that numbers may change whenever the total number of each type of component is modified by either adding or deleting components. Whenever the project file is saved, the components will be renumbered based on their SketchPad position starting from the top level and working from left to right then top to bottom on the SketchPad down through the building levels.

Menu Command: **E**dit → **F**ind...

Keyboard Shortcut: **C**trl+**F**

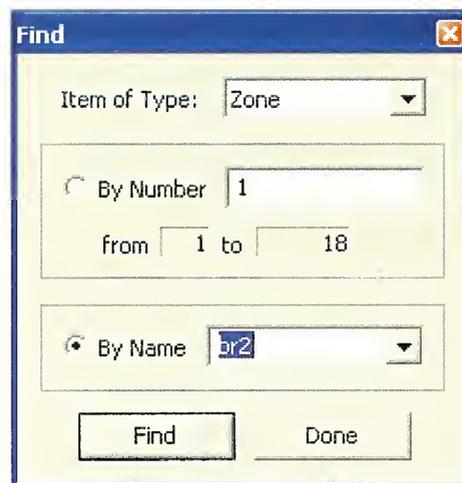


Figure – Find dialog box

Simply select the type of component icon you wish to locate, select either a number or name from the lists provided and click the *Find* button. The icon will be located and highlighted on the SketchPad. If there are no components of a given type, the search controls will be disabled.

Searchable items:

- Air Handling Systems
- Airflow Paths – including Supplies and Returns of Simple Air Handling Systems
- Control Nodes – sub-nodes of super nodes will defer to the super node
- Ducts: Segments, Junctions and Terminals
- Occupants
- Source/Sinks
- Undefined – undefined SketchPad icons
- Zones – including implicit supply and return zones of simple air handling systems

□ **Find and Replace**

The *Find and Replace* feature consists of a more advanced *find* feature that enables you to generate lists of items based on user-selected ranges of property values. You can then replace parameters of the items in the list of items found.

Menu Command: **Edit → Replace...**

Keyboard Shortcut: **Ctrl+R**

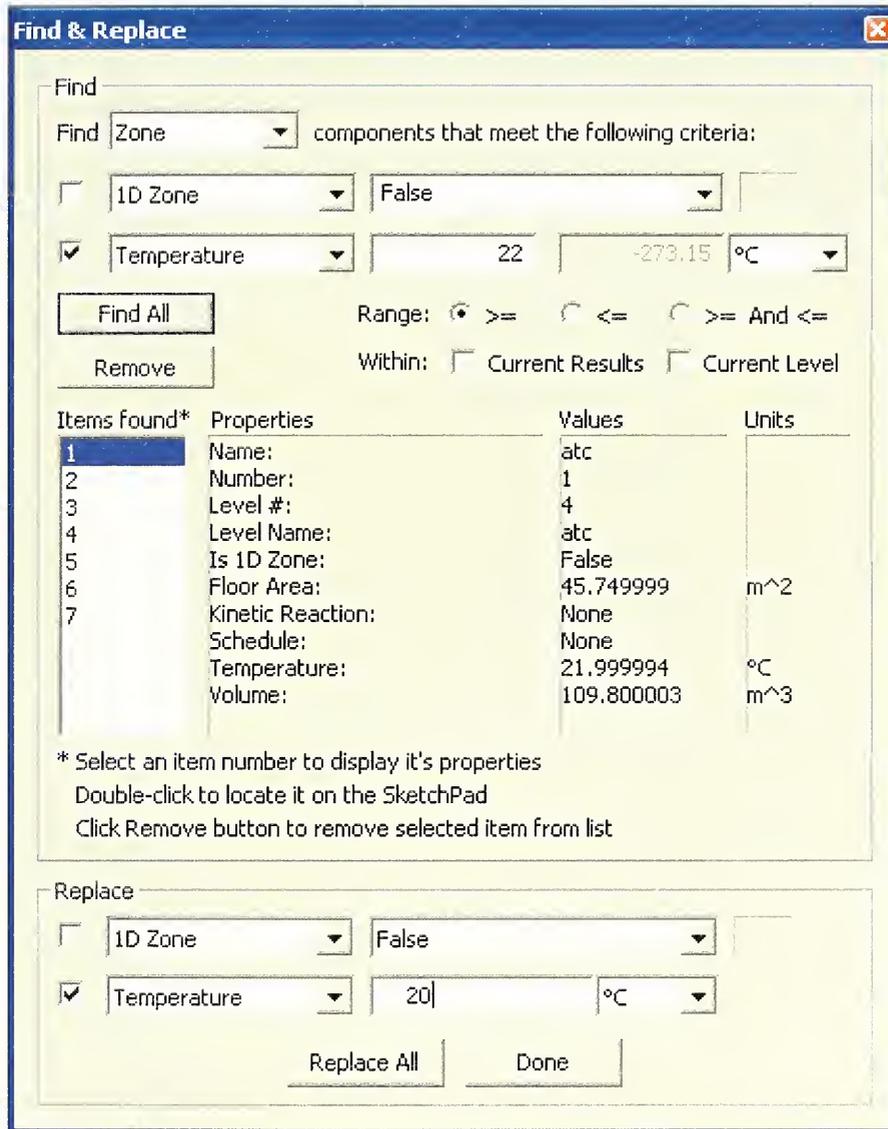


Figure – Find and Replace dialog box

There are two sections of the Find and Replace dialog box: Find and Replace. Use the dialog box controls within the Find section to generate a list of items that match the desired search criteria. Once you have generated a list of items found, then select the items within the Replace section to replace the desired parameters of the items within the list. You are not required to use the replace feature; you can simply use this tool to locate specific items within a project file.

Each type of building component has two predefined sets of parameters upon which searches can be performed: list-based and value-based. The list-based search parameters are accompanied by a

list of each parameter type contained in the project, e.g., elements, schedules, filters, etc. You can either select from the list to search for an exact match or type in a case-sensitive search string of text to match all items that contain the search string. Value-based parameters allow you to search for items based on a range of values of the selected parameter. It is possible to search using either of these sets of parameters or both. Use the check boxes that are located adjacent to the search parameters to select which ones to use.

Once you have set your search criteria, click the *Find All* button to initiate the search. This will generate a list of *Items found*. You can then select items in the list to display a summary of their properties, or double-click to locate the item on the SketchPad. You can also remove individual items from the list by selecting the item and clicking the *Remove* button. This allows you to refine the set of items for replacement.

Building Component	List Parameters	Value Parameters
Air Handling System	Kinetic reaction in return Kinetic reaction in supply Outdoor air filter Return air filter Outdoor air schedule	Minimum outdoor airflow Return system volume Supply system volume
Air Handling System Supply and Return	Air handling system Filter Schedule	Airflow rate
Airflow Path	Airflow element Filter Schedule	Multiplier Relative elevation
Control Node *	Control type	N/A
Duct Junction	Color Schedule	Relative elevation Temperature
Duct Segment	Color Duct flow element Filter Schedule	Length
Duct Terminal	Balanced Color Filter Schedule	Balance loss coefficient Design flow rate Free face area Relative elevation Temperature Terminal loss coefficient
Occupant	Inhalation schedule	Body weight Multiplier Peak inhalation rate
Source/Sink	Source/sink element Schedule Species	Multiplier
Zone	ID zone Color Kinetic reaction Name Schedule	Floor area Temperature Volume

* Control nodes are not replaceable using this feature

Table – Find and Replace options

3.1.6 Viewing Results on the SketchPad

ContamW provides some graphical displays of simulation results upon the SketchPad. You can display airflow rates and pressure differences across airflow paths on the SketchPad if a simulation has been performed and results of the simulation are available (see Viewing Results in the Working with Results section of the manual).

3.1.7 Viewing Envelope Wind Pressure Differentials

ContamW provides a feature that allows you to visualize the effects of wind upon a building envelope. A graphical display of wind pressures at each of the flow paths that are adjacent to the ambient zone can be viewed on the SketchPad (see Checking Wind Pressure Data in the Defining Steady State Weather and Wind section). This feature enables you to verify input of wind speed and direction data with respect to the orientation of the building envelope to the wind.

3.2 Working with Project Files

All information related to the layout, building components, and occupant information of your CONTAM project are stored in a "project" file. When you save the project file, it will be stored with the ".PRJ" extension. CONTAM project file names must conform to the file naming conventions of the Windows operating system. This project name is displayed in the Title bar of the ContamW application window.

All project file operations are contained under the **File** menu. These operations include: **New Project**, **Open Project...**, **Save Project**, and **Save Project As...** Note that some of these file operations have keyboard short-cuts and/or tool bar buttons associated with them.

□ Creating a New Project

Menu Command: **File → New Project**

Keyboard Shortcut: None

Toolbar Button: 

Use this command if you are currently working on a project and you wish to create a new one. This command will clear the SketchPad and all data related to the current project from ContamW's program memory. If your current project has not been saved prior to executing this command, you will be asked whether or not you wish to save it prior to clearing it.

A dialog box will appear enabling you to enter a project description (See **Editing the Project File Description** below)

□ Opening an Existing Project

Menu Command: **File → Open Project...**

Keyboard Shortcut: **Ctrl+O**

Toolbar Button: 

Use the menu item to open an existing CONTAM project file. This command will display the File Open dialog box typical of Windows applications. This dialog box is set to display only those files having the ".PRJ" extension in order to simplify your search for CONTAM project files. Follow the typical procedure for opening a file using the Windows operating system.

The Open dialog box provides you with both the Version and Description of the highlighted project file. You can edit the project file description via the **File → Project Description...** menu item.

When a project file is successfully read by ContamW, a backup copy of the project file is created with the name CONTAM.BKP in the program directory.

□ Converting Projects from Previous Versions

Menu Command: **File → Open Project...**

Keyboard Shortcut: **Ctrl+O**

This is the same as opening an existing project. However, after you open a project of an earlier version it will be untitled, so you must give the project a name when saving it. Once you save a project with the most recent format, you will no longer be able to open it using an earlier version of CONTAM.

A utility is also provided on the NIST website that will enable you to quickly update a set of project files to the current version. For example PrjUp24.exe will upgrade to the CONTAM 2.4

format. To utilize this utility, copy prjup24.exe and the desired project file(s) to a new directory, select the projects to update and drag and drop them onto the prjup24.exe file. This will replace the existing files with updated versions.

❑ Editing the Project File Description

Menu Command: **File → Project Description...**

Use this feature to enter a project file description which is provided to help you manage your project files. This description will be available when you are opening project files from within ContamW.

❑ Saving a Project

Menu Command: **File → Save Project**

Keyboard Shortcut: **Ctrl+S**

Toolbar Button: 

This option saves the building description to the project file under its current name.

❑ Renaming a Project

Menu Command: **File → Save Project As...**

Keyboard Shortcut: None

Toolbar Button: None

Use this command to save a copy of the current project file under a new name. This will display the Windows "Save As" dialog box and allow you to specify both a storage location and a file name for the project file. ContamW will automatically append the ".prj" extension to the filename, so you do not have to give the file an extension. If you do give the file an extension, ".prj" will be appended as well as the extension you provide. If you specify the name of an existing file, you will be warned that saving the file will overwrite the existing file. If you try to save an untitled or new project file prior to giving it a name, the "Save As" dialog box will be displayed.

❑ Exiting ContamW

Menu Command: **File → Exit**

Keyboard Shortcut: None

Toolbar Button: None

You can also exit the program using the standard window-closing button located in the upper right-hand corner of the program window.

3.3 Configuring ContamW

The project configuration settings of ContamW are available from the **Options...** selection of the **View** menu. This set of configuration properties will allow you to select the overall size of the SketchPad (i.e. the number cells that make up the height and width), the size of a single SketchPad cell, the default system of units (SI or IP), the default units of mass flow, the default zone temperature and the file path of an optional EWC to WPC file converter. All of these settings are saved with the CONTAM project with the exception of the cell size that resets back to the default size whenever you exit ContamW. You can also save the default units and values and the available cell sizes to a configuration file, CONTAM.CFG. ContamW looks for this file when starting and loads the saved settings if the file exists. This way you can set ContamW to start with the same default settings each time you run the program. The path name of the EWC-to-WPC file converter does not appear in the configuration dialog boxes, however if a converter is selected when you click the Save Configuration File button it will be saved to the CONTAM.CFG file. The value saved will be that which you have set in the **WPC File...** selection of the **Weather** menu. To reset the converter name back to "null," you must edit the configuration file manually.

The following sections provide detailed descriptions of the specific configuration properties. They are the context-sensitive help topics that you can access by pressing **F1** when working with property pages of the "Project Configuration Properties" property sheet.

3.3.1 Default Units

The following parameters define the units and selected values that will be used by default throughout the project when you are providing input to the program. You do not have to use these units throughout the project; however, you should select the primary set of units for convenience when entering data throughout a project. You can change these defaults at any time while working with a project, and you can select different units for individual parameters as you enter them.

Default System of Units: You can select the appropriate radio button to set the system of units that you would like the program to use. You can either select the International System of Units (SI) or the Inch-Pound (IP) system of units.

Default Units of Flow: You can select the appropriate radio button for the most commonly used units of flow in the current project. These are the units in which airflow simulation results will be displayed within the status bar. ContamW converts all airflow rates to and performs all calculations in mass flow units of kg/s. Note that all volumetric airflow rate units are at standard temperature and pressure.

Default Zone Temperature: This value will be the default temperature used whenever you create a new zone, duct junction or terminal within the current project. If you change this value, you will be prompted as to whether or not you would like to update the temperatures of all existing zones, junctions and terminals.

3.3.2 SketchPad Size

Sketchpad Size (cells): This is the height and width of the SketchPad. The units (cells) represent each space where an icon may be placed upon the SketchPad. The default (and minimum) values are 66 cells wide and 58 cells high.

Shift Drawing (reposition on SketchPad): If you need more space on one side of a drawing you can shift the entire project on the SketchPad. You may need to increase the size of the SketchPad to make room for the project before shifting it. Entering a positive number in the "Horizontally" edit box will shift the project to the right relative to the SketchPad, and entering a positive number in the "Vertically" edit box will move the project down relative to the SketchPad. A negative number will have the opposite effect.

3.3.3 Cell/Icon Size

Current Cell/Icon Size: There are seven options for displaying icons on the SketchPad (1x1, 2x2, 3x3, 4x4, 5x5, 8x8, and 16x16). The default option is 8x8. This is a "zooming" feature that provides a visual aid to enable you to view sketches in more detail or to fit larger sketches within the program window. The cell size has no effect on the project simulation. There are two "zooming" buttons provided on the toolbar to quickly increase and decrease the cell size according to the available cell sizes below.

Available Cell Sizes: Use these check boxes to select which sizes you want available. The 1x1 size causes problems with some versions of the Window 98 operating system. If this is the case for you, then deselect the 1x1 check box and click the "Save Configuration" button.

Cell Jump: Use this to set the number of SketchPad cells the cursor will jump when you use the Shift + Arrow Keys to maneuver around the SketchPad.

3.3.4 SketchPad Appearance

If you save the configuration file, then these options will default to the values set on this property page whenever you start ContamW.

SketchPad Options

Floating Status: Check this box to display the Floating Status box on the SketchPad. You can also turn the floating status box on and off via the View → **Floating Status (Ctrl+T)** menu command.

Color Zones: Check this box to display zone fill colors on the SketchPad. If this box is not checked, then all zones will be filled using the "Background" palette color. You can also turn the color zones option on and off via the View → **Color Zones (Ctrl+Z)** menu command.

Color Ducts: Check this box to display duct colors on the SketchPad. If this box is not checked, then all ducts will be displayed using the "Ducts" palette color. You can also turn the color ducts option on and off via the View → **Color Zones (Ctrl+Z)** menu command.

Palettes: ContamW provides the ability to modify the color of items displayed on the SketchPad. You can select one of four available palettes. You can not modify the colors of the "Default" palette, but you can modify the other three "User Palettes." Select a palette to display the colors of the palette in the **Palette Colors** section below.

Palette Colors: These are the colors use to display items on the SketchPad. Click the color box to change the color used to display each type of item. Only the colors of the User Palettes can be changed. The following is a brief description of each category.

- Foreground – Defined icons including walls
- Background – Background of SketchPad cells, default zone fill and fill of all zones when **Color Zones** option is not active.
- Flow Results – Lines displaying scaled flow results for airflow paths and duct terminals
- Pres Results – Lines displaying scaled pressure difference results across airflow paths in the Results and Wind Pressure view modes
- Drawing Line – Cell highlight when drawing on the SketchPad
- Undefined – Undefined icons
- Controls – Defined control icons
- Ducts – Default color of defined ducts and color of all defined ducts when **Color Ducts** view option is not active
- Errors – Background color of cells when an error condition is being highlighted on the SketchPad. Errors are indicated when a building check fails. Building checks are performed when you choose **Run Simulation** or **Create ContamX Input File** from the **Simulation** menu.
- Cursor – SketchPad drawing and icon selection cursor
- Highlight – Icon highlighting color for find operation
- Sub Level – All icons on sub-level when sub-level display option is activated via the **Level** → **Reveal Level Below (Ctrl+Shift+B)** menu command.

3.4 Working with Walls

In ContamW, walls are used to create enclosed regions referred to as zones on a given level. Walls include the building envelope and internal partitions with a significant resistance to airflow. You will note when drawing on the SketchPad that you can only draw walls in odd numbered rows and columns. This is done to avoid conflicts with ducts that can only be drawn on the even rows and columns of the SketchPad.

It is important to realize that ContamW is not meant to provide drawing capabilities similar to that of a Computer Aided Drawing (CAD) program. The SketchPad drawing functions were designed to provide a strict environment for creating the input to the underlying multizone model upon which the CONTAM simulation engine is based.

3.4.1 Drawing Walls

You can draw walls using either of the two wall drawing tools previously described in the "Working with the SketchPad" section. Use the *box drawing tool* to quickly draw a rectangular region and the *wall drawing tool* to draw a free form wall. Use the wall drawing tool to draw almost any shape wall. Walls must always form complete enclosures. Therefore, a wall cannot have an open or dangling end. It also may not be drawn across building component icons. You will receive a warning message if you attempt to draw an invalid wall.

3.4.2 Deleting Walls

You can delete a wall by moving the caret to any portion of the wall and selecting **Delete** from the **Edit** menu or using the keyboard shortcut **Del** key. ContamW will highlight the section of wall to be deleted and request confirmation to delete the indicated section. If the caret is on the intersection of three or more walls, you will be given multiple options of wall segments to delete. If you select "No" when asked to confirm deletion, the next option for deletion will become highlighted until you either delete a section or all of the options are exhausted.

3.4.3 Modifying Walls

You modify the positions or shapes of walls by adding and deleting wall sections. For example in the following figures, if you want to modify the zone in figure (a) to obtain the zone in figure (d), you would first add the dark line in figure (b) then delete the dark line in figure (c).



When modifying walls you may need to move or delete other icons. For example in the following figures, to create drawing (c) from drawing (a), you would delete the lighter line and the lighter zone icon from figure (b). You must delete one of the two zone icons, or you will end up with two zone icons within the same enclosed wall area. This is not permitted, and you will receive a message indicating a zone definition error.



3.5 Working with Levels

CONTAM projects are organized by levels. Each level of a CONTAM project contains a plan view drawing. Typically, each level of a CONTAM project corresponds to a floor of a building. Depending on the detail required for your modeling purposes, you could represent a building floor using multiple levels. For example, a level could be used to represent a ceiling or floor plenum.

3.5.1 Creating Levels

When you first start ContamW, a default level is created so that you can begin working on a drawing right away without having to create a new level. There are three commands in the **L**evel menu that you use when creating new levels. These are the **C**opy Level, **P**aste Level, and **I**nsert **B**lank Level commands. Whenever you create a new level, whether it is blank or a copy of another level, ContamW will give it a default name that will consist of a number enclosed within the "<" and ">" characters. You can modify this name later by editing the data associated with a level.

You must be careful when copying levels within ContamW. For example, you might have an air handler defined within a duct system on a level and then copy it to another level. This would create another air handler on the new level. If the ductwork is connected between the two levels, the two air handlers may act against each other. You must be careful to make connections between building levels in a manner that makes sense for your purposes.

□ Creating Blank Levels

Menu Command: **L**evel → **I**nsert **B**lank Level (**A**bove/**B**elow Current Level)
 Keyboard Shortcut: None
 Toolbar Button: None

Use these commands to create a blank level. When you create a blank level, you must select whether you want it created above or below the current level.

□ Copying and Pasting levels

Menu Command: **L**evel → **C**opy Level along with
Level → **P**aste Level (**A**bove/**B**elow Current Level)
 Keyboard Shortcut: None
 Toolbar Button: None

With ContamW you can copy an entire level and insert it as an entire level anywhere within the current project. Use the Copy Level command to copy the level currently displayed on the SketchPad; move to either the level above or below where you wish to insert the copied level and use one of the Paste Level commands to insert the copied level.

3.5.2 Viewing Levels

If a project has more than one level, you can specify the level that you want to view or modify on the SketchPad by using the level commands. All level-related commands are located under the **L**evel menu. There are also some shortcut keys and toolbar buttons provided for your convenience.

□ **Moving Up/Down a Level**

Menu Command: **Level → Go to (Level Above/Below)**

Keyboard Shortcut: **Page Up, Page Down**

Toolbar Button: 

Use this command to change which level you want the SketchPad to display.

□ **Changing the Currently Active Level**

Menu Command: **Level → Go to (Level Above/Below)**

Keyboard Shortcut: **Page Up, Page Down**

Toolbar Button: 

Use this command to change which level you want the SketchPad to display.

□ **Displaying Multiple Levels**

Menu Command: **Level → Reveal Level Below**

Keyboard Shortcut: **Ctrl+Shift+B**

Toolbar Button: None

Use this command to see both the current level and the level below the current level at the same time. ContamW will display the walls and building component icons of the level below in gray. This feature is useful for aligning building features between adjacent levels.

3.5.3 Deleting Levels

Menu Command: **Level → Delete Level**

Keyboard Shortcut: None

Toolbar Button: None

Use this command to delete an entire level. Once you have deleted a level, you cannot undo the deletion. You may want to save a copy of the file prior to deleting a level; this is the only way to prevent losing your work.

3.5.4 Modifying Levels

To modify level data, you access the Level Data dialog box using the **Edit Level Data...** command of the **Level** menu. You can also use the keyboard shortcut **F8** to display the dialog box. The following section shows the information contained on the "Level Data" dialog box.

3.5.5 Level Data

This is the information associated with each level that you create. ContamW will provide default values for this data, but you can modify it as required for your particular building.

Name: Name to identify the level. All level names must be unique.

Elevation of this level: The elevation of the base of the level above ground.

Distance to level above: This is the height of the level from floor to ceiling. ContamW will use this value to calculate zone volumes based on the floor area of each zone. If you change this value, a dialog box will appear asking if you would like ContamW to adjust the volumes of the zones on the level based on the new level height. If you select Yes, then the zone volumes will be adjusted by $V_{new} = V_{old} * H_{new}/H_{old}$. If there are *phantom zones* on the level, you will be advised that their volumes were not adjusted, so you should handle those zones individually.

3.5.6 Checking Levels

Menu Command: **Level → Check Current Level**

Keyboard Shortcut: None

Toolbar Button: None

Use this feature to check the data for the currently displayed level and make sure the building components are defined. Errors will be highlighted in red on the SketchPad.

3.6 Working with Zones

A *zone* is a volume of air separated from other volumes of air by walls, floor and ceiling. Generally, zones are characterized by having a uniform temperature and pressure (that varies hydrostatically) and contaminant concentration. However, the concentration within user-selected zones can be allowed to vary in one dimension (See 1D Zone Data Properties within this section) via the 1D convection/diffusion model (See the Theoretical Background section). In addition, there are three different types of zone icons: *normal*, *phantom* and *ambient*. A *normal* zone icon represents a zone that is separated by a floor from the zone below. A *phantom* zone is used to indicate that there is no floor below the enclosed region on a given level. That is, the region on this level is actually part of the zone below and has the same temperature and contaminant concentration. An *ambient* zone icon represents the outdoor air. The ambient zone, which surrounds the building, is already identified by default by the icon at the upper-left corner of the SketchPad.

You create zones on the SketchPad by first drawing walls that enclose a region, placing a single zone icon within the enclosed region, and then defining the zone icon. Eventually, every zone must be connected to either the ambient zone or a constant pressure zone either directly or indirectly via any set of paths that prevents the zone from being *isolated* from a zone of "known" pressure. This connectivity is required to enable a solution to the set of airflow equations solved by ContamX.

3.6.1 Creating Zones

You create zones by drawing walls upon the SketchPad. The wall drawing operations are described in the Drawing Walls section of this manual. The shape and size of zones as drawn upon the SketchPad do not provide the underlying model with any scaling information. However, the manner in which the enclosed regions border each other is significant. Any zones between which you wish to provide a direct connection (via an airflow path) must share a common wall. Therefore, when drawing a building floor plan, you should try to maintain the general topology of the actual floor plan. While scale is not significant, ContamW does provide you with the SketchPad coordinates that may help you when laying out your project.

The dimensions of your zones are determined when you define them. If you change the shape or size of a region on the SketchPad that contains a zone icon, the dimensions do not change according to the model unless you actually modify the properties of the associated zone icon.

3.6.2 Deleting Zones

You delete a zone by first deleting the zone icon. After you delete the zone icon, delete the wall or walls necessary to eliminate the enclosed region that is left behind. You must avoid the existence of an enclosed region without a zone icon; CONTAM will not perform a simulation with an undefined zone icon in a project.

3.6.3 Modifying Zones

You modify the SketchPad representation of a zone by adding and removing walls. This procedure is described in the Modifying Walls section of this manual. To modify the parameters of a zone, you use the icon definition procedure (See Defining Building Component Icons) to display its properties and make the desired changes.

Once you define a zone, you can move the zone icon anywhere within the zone using the **C**ut and **P**aste functions of the **E**dit menu. These commands will only allow you to paste the icon into the zone in which it is currently located.

3.6.4 Defining Zones

After you draw the enclosed region of a zone, you must define the zone. To define a zone you must first draw a zone icon within the enclosed region (See Drawing Building Component Icons) and then use the icon definition procedure (See Defining Building Component Icons) to display and edit the properties of the zone.

The properties of each zone include a name, temperature (constant or scheduled), pressure (variable or constant), volume, information describing contaminant behavior within the zone, and 1D zone information. You must provide each zone with a name that is unique to each level of a project. For this reason, you cannot copy zone data within a level. However, you can copy entire levels of data, including zone data, from one level to another (See Creating Levels). Detailed descriptions of zone properties are given under Zone Properties below.

Only *normal zones* require definition. *Ambient* and *phantom* zone icons appear as defined icons as soon as you place them onto the SketchPad. However, you must include the volume of *phantom zones* in the volume of the associated *normal zone* below it.

3.6.4.1 Zone Properties

This section provides detailed descriptions of the specific zone properties. The following sections are the context-sensitive help topics that you can access by pressing **F1** when working with property pages of the "Zone Properties" property sheet.

Zone – Zone Data Properties

These are the basic properties that describe a zone.

Zone Name: This is the symbolic name of the zone. Enter a name up to 15 characters in length. Zone names must be unique within each level of a building.

Dimensions

Volume: Zone volume is used in the dynamic contaminant calculations. For phantom zones, the zone volume is set at the standard zone icon on the lowest level for the entire height of the zone and includes the volume of the phantom zones on the levels above.

Floor Area: Instead of the volume, you may enter a floor area. Floor area is then multiplied by the height of the current building level to compute a volume. This area is not used for any other purpose by CONTAM.

Color: Select a zone fill color to help you distinguish zones on the SketchPad. You must activate the **Color Zones** option from the **V**iew menu in order to display zone colors on the SketchPad.

Include in Building Volume: This refers to the calculation of whole building *air change rates* that you can now have CONTAM perform during simulation. The associated check box is checked by default, so if you do not want the zone to be included in the whole *building volume* you should uncheck this box. The *air change rate* and *building pressurization* calculations will calculate the total rate of airflow from *ambient* into all of the zones having

this box checked. Typically, you would include all *conditioned space* in the building volume; for example, you might not want to include an attic or crawl space.

Temperature

Constant or Scheduled: Select to either maintain zone temperatures at a **constant** value or to change according to a user-defined **schedule**. If you select Scheduled, then you must associate a temperature schedule with the zone. Selecting Scheduled will also set the simulation run control to vary the density within building zones during simulation (See Airflow Numerics Properties in the Working with Simulations section).

Temperature: Set the value you want ContamW to use as the constant temperature of the zone when you select the "Constant" radio button. Whenever you create a new zone, ContamW will use the default value which you can override by entering another value. You can set the default zone temperature via the **Options...** command of the **View** menu.

Temperature Schedule: Select the temperature schedule you want to associate with this zone when you select the "Scheduled" radio button. (See Working with Schedules).

Pressure: The zone air pressure relative to ambient can be specified as either variable or constant. Typically you would set the zone pressure to be variable and allow it to be calculated during the simulation. However, you might set the pressure to be constant if you want to simulate a *fan pressurization (blower door)* test of a building (see Building Pressurization Test in the Applications section of the manual) or to perform analytic test cases of CONTAM.

Zone – Contaminant Properties

These are the contaminant-related properties of a zone. Contaminants must be defined prior to defining contaminant data for a zone. To define contaminants, select **Data** then **Contaminants...** from the main program menu.

Zone Name: This is the symbolic name of the zone as entered on the Zone Data property page.

Contaminant Concentrations: Select *variable* or *constant*. Typically you would set this to *variable* and allow ContamX to calculate the contaminant concentration within the zone during simulation. However, you may wish to set this to *constant* as a simple means of creating a simple contaminant source within a zone. If you set this to *constant*, the contaminant concentrations in this zone will begin and remain at the values you set for the initial concentrations.

Initial Concentration: Select a contaminant from the list and enter the *initial contaminant concentrations* for dynamic (transient) simulations. Note that only those species you've selected to be contaminants (to use during simulation) appear in the list. If a species for which you wish to set an initial concentration does not appear in the list you must set "Use in simulation" property of the corresponding species to be true (See Creating Species and Contaminants). You can reset these initial values through the Run Control Properties of the Simulation Parameters.

NOTE: Changing the number of contaminants, i.e. those species used in the simulation, will reset the *initial concentrations* of all zones to the default contaminant concentrations. You can reset all of these concentrations via the Run Control Properties as indicated above or individually here.

Sum of Non-trace Initial Concentrations: This value is provided as a check for you to insure that you input a reasonable set of initial values when using *non-trace contaminants*. This value should be very close to 1.0.

Kinetic Reactions: If there are previously defined kinetic reactions within the current project, you may select one from the list of names. Click the "New Reaction" button to define a new kinetic reaction. To view or modify existing kinetic reaction data click the "Edit Reaction" button. Setting this field to <none> indicates no reactions in the zone. You can also import kinetic reactions from a contaminant-related library file (See Working with Data and Libraries).

Zone – 1D Zone Data Properties

These are the properties used to define 1D convection/diffusion zones in which contaminant concentrations can vary along a user-specified axis. This option is provided for cases in which you need to account for the delay in contaminant transport within long or tall zones. If this feature is enabled, ContamX will subdivide this zone along the specified axis into multiple cells or sub-nodes for the purposes of contaminant transport calculations.

Zone Name: This is the symbolic name of the zone as entered on the Zone Data property page.

Set 1D Zone

1D Zone: Check this box to treat as a 1D convection/diffusion zone. Otherwise the zone will be treated as a well-mixed zone.

Axial Dispersion Coef: Enter the axial dispersion coefficient and units for the zone. Note that the zone dispersion coefficient is not necessarily the molecular diffusion properties of the defined contaminants. This coefficient is empirically based and should account for turbulent mixing as well.

Axis End Point Coordinates

Define the end points, **Point 1** and **Point 2**, of the axis of the 1D zone. These coordinates will define the direction along which contaminant transport will be calculated when treating this zone as a 1D convection diffusion zone. These values are required to establish the relationship of other building components with respect to this zone, i.e., flow paths, duct junctions and terminals, simple air-handling unit supply and return points, contaminant source/sinks, sensors and occupants.

While the coordinate system is user-defined, ContamW will verify the relationship of 1D zones to associated building components prior to running a simulation to insure that the coordinates “match up.” This verification is performed automatically when you select either the **Run Simulation** or **Create a ContamX Input File** command from the **Simulation** menu. If there are discrepancies, messages will be displayed notifying you of the nature of the discrepancies.

X and Y: The X and Y values are in absolute coordinates. These values can be based on scale drawings and/or a user-defined coordinate system. Unlike walls on the SketchPad, axes are not required to be rectilinear, i.e., you may specify axes that are not horizontal or vertical in the x-y plane.

Rel Elevation: Enter the Z-coordinate as the height relative to the base of the zone.

Cell Size: This is the cell size into which ContamX will sub-divide the zone during simulation if the zone is to be treated as a 1D convection/diffusion zone. If the axis length is not evenly divisible by the cell size, then the cell size will be adjusted as needed. While there is no strict guidance on the selection of cell size, it should generally be on the order of the flow velocity along the axis times the simulation time step.

Units: Set the units of the above items.

3.7 Working with Airflow Paths

An airflow path is a CONTAM building component through which air can move between two adjacent zones. These components can be cracks in the building envelope, open doorways, exhaust fans, etc. The air pressures in the adjacent zones and the flow characteristics of the flow path itself determine the flow through most of these paths. The location of the SketchPad icon that represents a flow path determines the two zones that are connected by a flow path. If the path icon is placed on a wall, it connects the zones on opposite sides of the wall. If it is placed in a blank cell of a zone (i.e. on the floor), it connects the zone containing the flow path icon to the zone located directly below the flow path icon.

You can place airflow paths on the SketchPad, define their flow characteristics, move, copy, and delete them. You must provide each airflow path with specific information that describes its flow characteristics. Much of the information that describes the flow characteristics of airflow paths is contained in *airflow elements* that you must create when defining the airflow paths. The details of drawing, defining and modifying airflow paths are described in the following sections.

3.7.1 Airflow Elements

Each airflow path must refer to an airflow element. *Airflow elements* describe the mathematical relationship between the flow through an airflow path and the pressure drop across the path. ContamW provides you with several mathematical models or types to choose from. Each of these airflow element types is described in detail in the Airflow Path Properties section. While every airflow path must refer to a single *airflow element*, multiple paths can refer to the same *airflow element*. *Airflow elements* can also be stored within a CONTAM library file and shared between different CONTAM project files (See Working with Data and Libraries).

3.7.2 Creating Airflow Paths

You create airflow paths by placing an airflow path icon on the SketchPad (See Drawing Building Component Icons). Because flow paths connect only two adjacent zones, there are several restrictions on the placement of flow paths. You may not place airflow path icons on the corner of a zone or on the blank cell of a level that does not have a zone located directly below it. Note that you can place an airflow path icon on the "roof" of a building that connects a zone below to the ambient. In this case, the roof level would be a level that does not contain walls but contains airflow path icons located in the floor of the roof level (above zones on the level below). The context-sensitive feature of the pop-up icon placement menu enforces these restrictions. The "Flow Path" menu selection will be disabled (grayed out) if you pop-up on a location where ContamW does not allow the icon to be placed.

This table shows the various flow path icons and provides a brief description of each.

Icons	Description
	Small and large one-way flow paths
	Small and large two-way flow paths
	Directional fan flow paths

3.7.3 Deleting Airflow Paths

You delete airflow path icons using the icon deletion procedure (See Deleting Building Component Icons). Deleting airflow path icons from the SketchPad does not delete any airflow elements that you may have defined. These flow elements can only be deleted using the CONTAM Library Manager, accessible through the **Data** menu for each type of data element.

3.7.4 Modifying Airflow Paths

To modify the parameters of an airflow path, you use the icon definition procedure (See Defining Building Component Icons) to display its properties and make the desired changes. You can also move and copy airflow paths.

□ Moving Airflow Path Icons

Once you define an airflow path, you can move the path icon using the **Cut** and **Paste** functions of the **Edit** menu. ContamW will constrain the movement of path icons according to the following rules. You can only move a flow path through a wall onto a wall, not onto the floor of a zone. You can only move a flow path through a floor to another location on the floor, not onto a wall. You can only move a path icon within the same level upon which it is located.

□ Copying Airflow Path Icons

Once you define an airflow path, you can copy the path icon using the **Copy** and **Paste** functions of the **Edit** menu. ContamW will constrain the copying of path icons according to the following rules. You can only copy a flow path through a wall onto a wall, not onto the floor of a zone. You can only copy a flow path through a floor to another location on the floor, not onto a wall. You may copy a path icon from one level to another.

3.7.5 Defining Airflow Paths

After you draw a *Flow Path icon* on the SketchPad you must define it using the icon definition procedure (See Defining Building Component Icons) to display and edit the properties of the airflow path. This will display the "Airflow Path Properties" property sheet. Detailed descriptions of all airflow path properties are given in the Airflow Path Properties section of this manual. Once you have defined the properties of an airflow path, the icon will be displayed in black. The icon representing the defined airflow path may be slightly different from the original undefined icon depending on the type of airflow element you associate with the path (See the list below). From now on, you use this icon to access the properties of the airflow path.

Some of the properties associated with an airflow path are optional: *contaminant filtration*, *schedule*, *wind pressure* and *pressure limit*. You can associate contaminant filters with any flow path, and you can define a filter for each contaminant contained in your CONTAM project. A filter could be used, for example, to simulate the penetration of particles through a wall cavity. You can use a schedule to vary the flow through a path as a function of time. For example, you could use a schedule to open and close a path that represents a window. CONTAM allows you to account for the direct effects of wind pressure on the envelope of a building. The envelope is considered to be any wall that is adjacent to the ambient zone. If an airflow path is directly connected to the ambient zone, a "Wind Pressure" property page will be displayed to allow you to account for the effects of wind pressure on the path through the building envelope. Pressure limits are useful when you are using CONTAM to design a *smoke control system*. For example, you could set pressure limits (maximum and minimum) for stairwell doors so that those

doorways for which the pressure limits are violated will be highlighted upon the SketchPad when you view simulation results.

3.7.5.1 Airflow Path Properties

This section provides detailed descriptions of the specific airflow path properties. Each of the following subsections are the context-sensitive help topics that you can access by pressing **F1** when working with property pages of the "Airflow Path Properties" property sheet.

Airflow Path – Airflow Element Properties

Path Number: This is the number that appears in the status bar when you highlight a path icon. ContamW automatically assigns this number to each *airflow path* once they are defined. This number could change as you add and remove paths to or from a project. Each time you save a project, the path numbers are reassigned. Numbers are assigned beginning on the top level, starting from the upper-left corner of the SketchPad, moving left-to-right and top-to-bottom of the SketchPad. Paths include supplies and returns of simple air handling systems.

Airflow Element Name: <required> You may select a previously defined airflow element from those contained in the Name list. Click the "New Element" button to define a new airflow element. To view or modify airflow element data click the "Edit Element" button. You can also access the CONTAM Library Manager to import airflow elements from library files by clicking the "Library" button (See Working with Data and Libraries). Every airflow path must have an airflow element associated with it. This field may not be left blank.

Model Summary: This summarizes the information associated with the airflow element whose name is currently displayed in the Name field.

Airflow Path – Filters and Schedules

Filter: You may either create, edit or delete the filter associated with this flow path. Creating a new filter or modifying an existing one will display the Filter Properties dialog box. Use this dialog box to create and edit filter elements as needed. For an explanation of contaminant filters see Contaminant Filters in the Using CONTAMW section of the manual.

Schedule: If there are previously defined schedules within the current project, you may select one from the Name list. Click the "New Schedule" button to define a new schedule. To view or modify an existing schedule click the "Edit Schedule" button. Setting this field to <none> indicates no schedule. (See Working with Schedules)

Airflow Path – Flow Path Properties

Relative Elevation: The elevation of the path will determine how it responds to, and influences, the building stack effect. Enter the elevation of the flow path relative to the elevation of the building level on which it is located.

For most flow paths (those not associated with a two-way flow model) enter the elevation of the mid-height of the flow path. Enter the elevation of the bottom of the opening for flow paths associated with a two-way flow element (See Airflow Element Types).

Multiplier: The multiplier is a constant value allowing you, for example, to use a flow element description of one window and a multiplier of 5 to describe 5 windows in a particular wall. If

you have specified a "leakage area" flow element, complete the description here by giving the reference area, length, or number of units.

Positive Flow Direction: In the cases of fans and flow/pressure limits (see below) it is necessary to know which direction is defined as positive. The arrows indicate the positive direction for flow of fans (forced flow element types) and flow and pressure drop for flow/pressure limits.

Limits: In the design of smoke control systems there are often limits on the maximum and minimum pressure or flow on a path. For example, a stairwell door can require a minimum pressure drop from the stairwell to the zone to prevent smoke entering the evacuation route, but there is also a maximum pressure drop above which some people may not be able to open the door. You may specify either type of limits (default: no limits) and set the values. When you display the results of a simulation, those paths with pressures or flows outside the limits will be highlighted.

Location: These are the absolute X and Y coordinates and units for the selected airflow path. The Z coordinate will be taken as the *Relative Elevation* of the airflow path with respect to the building level on which the path is located.

These values are required for flow paths through the building envelope when using a *Wind Pressure and Contaminant (WPC)* file (see Wind Pressure and Ambient Contaminant Files) or for any path that connects to a *1D convection/diffusion zone* (see 1D Zone Data). You can provide both coordinate values; however, when working with 1D convection/diffusion zones only the value corresponding to the 1D axis of the zone will be relevant during the simulation. These coordinates will be verified by ContamW against a WPC file or the axis of the 1D zones to which the path connects. This verification is performed automatically when you select either the **Run Simulation** or **Create a ContamX Input File** command from the **Simulation** menu.

Airflow Path Wind Properties

Wind Pressure Option: There are three wind pressure options for openings in the building envelope:

1. No wind pressure
2. Constant pressure
3. Pressure dependent on wind speed and direction

When you select one of the above options, the following data entry options that correspond to the wind pressure option that you selected will be made available for your input.

Constant Pressure Data: Enter a constant wind pressure (wind speed and direction independent).

Variable Pressure Data: Variable wind pressures are computed from the product of the following three parameters:

1. The dynamic pressure of the wind at some reference height. ContamX will determine this pressure based on either the steady state wind data (See Wind Properties) or the transient wind data from a weather file (See Defining Transient Weather and Wind).
2. A coefficient accounting for local terrain effects (Wind Pressure Modifier).
3. A pressure coefficient accounting for relative wind direction (Wind Pressure Profile).

Wind Pressure Modifier: CONTAM uses this value to account for differences between wind velocity profiles at the building site and that at a measurement location, e.g. an airport weather station (See Working with Weather and Wind). CONTAM only requires this value if you are implementing Variable Wind Pressure for an airflow path. CONTAMW will provide a default value for this based on the data entered via the Weather and Wind Parameters Property Page or you can override the default value for a particular airflow path.

Wall Azimuth Angle: Enter the direction the wall faces (clockwise from north).

Profile: The wind pressure profile accounts for the wind direction effects. You may select a previously defined wind pressure profile from those contained in the Name list. Click the "New Profile" button to define a new wind pressure profile. To view or modify wind pressure profile data, click the "Edit Profile" button. After you press the "New Profile" or "Edit Profile" button, the Wind Pressure Profile page will be displayed with a graphical representation of the profile.

Wind Pressure Profile

Wind Pressure Profiles are used to account for the wind direction effects in flow elements. CONTAM refers to the function relating the average wind pressure coefficient for the face of a building to the angle of incidence of the wind on the face of the building as the *wind pressure profile* or $f(\theta)$. A more detailed explanation of this function is given in the Working with Weather and Wind section. The profile is displayed graphically at the bottom of the screen for a visual review of the data.

Name: This is the name you give to the wind pressure profile. Wind profile names must be unique within a project.

Description: Field for entering a more detailed description of the specific wind pressure profile.

Data Points: You may enter up to 16 angle/pressure coefficient pairs. Pressure coefficients are in the range -1 to 1. The first value must be at a wind direction of zero degrees. The angle zero indicates a wind blowing directly toward the surface in which the opening lies. An identical value at angle 360° is automatically assumed. 90° indicates a wind blowing parallel to the wall from the right, and 270° is from the left. By making these angles relative to the wall, the same profile might be used on different walls around the building without modification depending on building geometry.

To update entered data points on the graph press the "Redraw" button in the curve fit frame. Pressing this button will re-plot the data on the graph and will sort the data points by angle.

Select Curve Fit:

Curve Fit 1: This curve fit simply connects the data points and linearly interpolates between the data points.

Curve Fit 2: This curve fit connects all the user data points using a nonlinear curve fit between the points (cubic spline).

Curve Fit 3: This curve fit requires five data points – one for each wind direction angle 0°, 90°, 180°, 270° and 360°. The curve fits a trend line to these five data points (ignoring any others) according to the following equation developed by Walker and Wilson [Walker and Wilson 1994 and ASHRAE 2005 p 27.6].

$$C_p(\theta) = \frac{1}{2} \left[(C_p(0^\circ) + C_p(180^\circ))(\cos^2 \theta)^{\frac{1}{2}} + (C_p(0^\circ) - C_p(180^\circ))(\cos \theta)^{\frac{3}{2}} + (C_p(90^\circ) + C_p(270^\circ))(\sin^2 \theta)^{\frac{1}{2}} + (C_p(90^\circ) - C_p(270^\circ))(\sin \theta)^{\frac{3}{2}} \right]$$

Airflow Element Types

This screen provides you with the option to choose which airflow model you want to associate with the airflow element that you are about to create. For an explanation of the various types of airflow models that CONTAM implements, see Airflow Analysis in the Theoretical Background section of the manual.

□ One-way Flow using Powerlaw Models

These one way flow models permit flow in the direction of the pressure drop. CONTAM provides you with the following powerlaw flow models.

1. $Q = C(\Delta P)^n$: This is the general form of the powerlaw model in volumetric flow form that allows you to directly input the coefficient, C , and exponent, n .
2. $F = C(\Delta P)^n$: This is the general form of the powerlaw model in mass flow form that allows you to directly input the coefficient, C , and exponent, n .
3. Leakage Area Data: This model refers to effective leakage areas as described in Chapter 26 of the *2001 ASHRAE Handbook of Fundamentals*.
4. Connection (ASCOS) Data: Refers to the airflow description used in the ASCOS program [Klote 1982].
5. Orifice Area Data: Relates the opening description to the orifice area data.
6. Crack Description: A narrow opening described by its length and width.
7. Test Data (1-point): Uses a single flow rate and pressure drop along with an estimate for the pressure exponent, n .
8. Test Data (2-points): Uses two flow rates with their corresponding pressure drops to define the flow using the powerlaw model.
9. Stairwell: Data corresponding to a stairwell is fit to the powerlaw model.
10. Shaft: Describes the flow in a large shaft.

□ One-way Flow using Quadratic Models

These models permit flow in only one direction through the path based on the pressure drop.

1. $\Delta P = aQ + bQ^2$ and $\Delta P = aF + bF^2$: These models allow you to directly input the "a" and "b" coefficients for the Quadratic models.
2. Crack Description: This model describes a narrow opening in greater detail than the powerlaw version of the crack model.
3. Test Data (2- point): This model uses two flow rates and their corresponding pressure drops to define the flow using the quadratic model.

□ Two-way Flow Models

These models are used to represent tall openings where temperature differences across the opening may cause two-way flow across the opening, e.g., doorways.

1. Single Opening Model with 2-way Flow: This models the flow through a path as a single large opening.

2. Two-opening Model: This model divides an opening vertically and models the flow using two powerlaw models, one for each opening. Results are obtained in the form of the net flow rate in each direction.

❑ **Backdraft Damper Flow Models**

The models allow you to model a feature that has different resistances depending on the direction of the pressure drop, e.g., a smoke control damper. CONTAM implements both a volume flow and a mass flow version of the backdraft damper model, both of which are based on the powerlaw model.

1. Backdraft Damper Model: $F = C(\Delta P)^n$: This is the volumetric flow form of the Backdraft Damper model. You input separate values of C and n depending on the sign of the pressure drop across the damper.
2. Backdraft Damper Model: $Q = C(\Delta P)^n$: This is the mass flow form of the Backdraft Damper model. You input separate values of C and n depending on the sign of the pressure drop across the damper.

❑ **Fan and Forced Flow Models**

These models allow you to create airflow paths that represent fans or other paths for which you want to designate a flow rate.

1. Constant Mass Flow Fan: This model provides for a constant mass flow rate.
2. Constant Volume Flow Fan: This model provides for a constant volumetric flow rate, but the mass flow will vary based on the actual density.
3. Cubic Polynomial Fan Model: This model implements a user-defined fan performance curve to relate pressure vs. flow rate.

❑ **Cubic Spline Flow Models**

These models provide the ability to create user-defined airflow elements based on a set of data points to which a cubic spline curve is fit. There are four versions that relate either mass flow or volume flow to pressure difference across the flow element (or vice versa). The cubic spline curve fit insures that the relationship between flow and pressure is differentiable as required by the CONTAM airflow solver.

1. F vs P: *Mass flow rate as a function of Pressure difference*
2. Q vs P: *Volume flow rate as a function of Pressure difference*
3. P vs F: *Pressure difference as a function of Mass flow rate*
4. P vs Q: *Pressure difference as a function of Volume flow rate*

One-way Flow using Powerlaw Models

CONTAM performs airflow calculations using the mass flow powerlaw formula, $F = C(\Delta P)^n$, for the following types of airflow elements. Where F is the mass flow rate, ΔP is the pressure difference across a flow path, C is the flow coefficient and n is the flow exponent. However, for some of the powerlaw models, ContamW enables you to input parameters that describe an opening in terms other than the coefficient C and exponent n. These parameters are then converted internally by ContamW to determine C and n for the mass flow version of the powerlaw model.

Powerlaw Model: $Q = C(\Delta P)^n$

This model allows you to directly enter the coefficient C and exponent n for the volume flow version of the powerlaw model.

Name: Enter the name you want to use to identify the airflow element. The airflow element will be saved within the current project and can be associated with multiple airflow paths.

Flow Coefficient (C): The coefficients may only be expressed in SI units due to the conversion method used. Use the following conversion to convert from IP units to SI units.

To convert from units of $\frac{cfm}{in H_2O^n}$ to the SI units of $\frac{m^3 / s}{Pa^n}$ multiply by $\frac{1}{2119 \times 249^n}$

Flow Exponent (n): Flow exponents vary from 0.5 for large openings where the flow is dominated by dynamic effects, and 1.0 for narrow openings dominated by viscous effects. Measurements usually indicate a flow exponent of 0.6 to 0.7 for typical infiltration openings.

Description: Field for entering a more detailed description of the specific airflow element.

Icon: Choose either the small or large opening icon as appropriate for the specific airflow element. The icon has no effect on the simulations.

Powerlaw Model: $F = C(\Delta P)^n$

This model allows you to directly enter the coefficient C and exponent n for the mass flow version of the powerlaw model.

Name: Enter the name you want to use to identify the airflow element. The airflow element will be saved within the current project and can be associated with multiple airflow paths.

Flow Coefficient (C): The coefficients may only be expressed in SI units due to the conversion method used. Use the following conversion to convert from IP units to SI units.

To convert from units of $\frac{lbm / s}{in H_2O^n}$ to the SI units of $\frac{kg / s}{Pa^n}$ multiply by $\frac{0.4536}{249^n}$

Flow Exponent (n): Flow exponents vary from 0.5 for large openings where the flow is dominated by dynamic effects, and 1.0 for narrow openings dominated by viscous effects. Measurements usually indicate a flow exponent of 0.6 to 0.7 for typical infiltration openings.

Description: Field for entering a more detailed description of the specific airflow element.

Icon: Choose either the small or large opening icon as appropriate for the specific airflow element. The icon has no effect on the simulations.

Powerlaw Model: Leakage Area

Leakage area refers to a description of airflow features given in Chapter 26 of the 2001 ASHRAE Handbook of Fundamentals [ASHRAE 2001]. Table 1 of this reference provides typical leakage areas for residential buildings.

Name: Enter the name you want to use to identify the airflow element. The airflow element will be saved within the current project and can be associated with multiple airflow paths.

Leakage Area: There are three possible ways to enter the leakage area: per item, per unit length, and per area. Select the radio button for the type of leakage area to be described, then enter the appropriate value for the type of leakage selected.

Per Item: Enter a total leakage value for an item, this is usually used for a doorway or window – something that can be classified as an item.

Per Unit Length: Commonly used to describe an interface such as a wall/ceiling junction.

Per Unit Area: Used to describe an area such as a wall or floor.

Reference Conditions: Be sure to check the reference condition for the reported leakage areas. Two sets of reference conditions are common [ASHRAE 2005 p 27.13]:

Discharge coefficient of 1.00 at a reference pressure difference of 4.0 Pa and

Discharge coefficient of 0.611 at a reference pressure difference of 10 Pa.

Discharge Coefficient: Enter the discharge coefficient for the leakage area at the reference pressure difference.

Flow Exponent: Enter the flow coefficient for the leakage area at the reference pressure difference. The flow exponent is not reported and therefore must be estimated. For openings associated with infiltration, measurements usually indicate an exponent between 0.6 and 0.7.

Pressure Difference: Enter the reference pressure difference for the associated leakage rating.

Description: Field for entering a more detailed description of the specific airflow element.

Icon: Choose either the small or large opening icon as appropriate for the specific airflow element. The icon has no effect on the simulations.

Powerlaw Model: Connection (ASCOS)

The ASCOS connection element is provided for compatibility with the ASCOS program [Klote 1982]. It is an implementation of the more general orifice flow element based upon the orifice equation.

$$Q = K \cdot A_o \cdot \sqrt{2 \Delta P / \rho}$$

The inputs for the ASCOS connection are the opening area, A_o , and the dimensionless flow coefficient, K . Data describing a connection are reduced to the powerlaw model with an exponent of 0.5. The orifice flow element (see Powerlaw Model: Orifice Area) provides a more general implementation.

Name: Enter the name you want to use to identify the airflow element. The airflow element will be saved within the current project and can be associated with multiple airflow paths.

Flow Coefficient: The flow coefficient K is related to the dynamic effects and is typically close to 0.6 for an orifice and slightly higher for other openings in buildings.

Flow Area: A_o , refers to the observable area of the opening.

Description: Field for entering a more detailed description of the specific airflow element.

Icon: Choose either the small or large opening icon as appropriate for the specific airflow element. The icon has no effect on the simulations.

Powerlaw Model: Orifice Area

This airflow element allows you to input the description of an orifice.

Name: Enter the name you want to use to identify the airflow element. The airflow element will be saved within the current project and can be associated with multiple airflow paths.

Cross-sectional Area: This refers to the observable area of the opening.

Flow Exponent (n): Flow exponents vary from 0.5 for large openings where the flow is dominated by dynamic effects, and 1.0 for narrow openings dominated by viscous effects. Measurements usually indicate a flow exponent of 0.6 to 0.7 for typical infiltration openings.

Discharge Coefficient: The discharge coefficient, C, is related to the dynamic effects and is typically close to 0.6 for an orifice and slightly higher for other openings in buildings.

Hydraulic Diameter: The hydraulic diameter is equal to $(4 \cdot \text{Area} / \text{Perimeter})$. For square openings this equals the square root of the area, and for long thin openings it is two times the width.

Reynolds Number: The transition from laminar flow to turbulent flow occurs over a very broad range of Reynolds numbers with the flow being fully laminar approximately below 100.

NOTE: The hydraulic diameter and Reynolds number have little impact on the calculations. Generally you should use the default values except for special circumstances where you need them to be modified. The parameters above describe the flow characteristics of an orifice in typical operation. At extremely low pressure drops the use of the powerlaw model leads to a division by zero during the network solution process. ContamX avoids this problem by changing to a linear model in this region. The model is based conceptually on the flow changing from turbulent to laminar at very low pressures. The Hydraulic diameter and Reynolds number are used to determine a point where the model changes from the powerlaw to linear.

Description: Field for entering a more detailed description of the specific airflow element.

Icon: Choose either the small or large opening icon as appropriate for the specific airflow element. The icon has no effect on the simulations.

Powerlaw Model: Crack Description

This airflow element allows you to enter the description of a crack and is based upon the relationships developed between flow through cracks and the powerlaw model [Clark 1985].

Name: Enter the name you want to use to identify the airflow element. The airflow element will be saved within the current project and can be associated with multiple airflow paths.

Crack Length/Width: Enter the length and width of the crack.

Description: Field for entering a more detailed description of the specific airflow element.

NOTE: ContamW automatically uses the icon for a small opening for the powerlaw model crack description.

Powerlaw Model: Test Data (1 point)

This airflow element allows you to provide a single test data point that ContamW reduces to the powerlaw model by calculating the flow coefficient based on input values (See Fitting Powerlaw Coefficients in the Theoretical Background section).

Name: Enter the name you want to use to identify the airflow element. The airflow element will be saved within the current project and can be associated with multiple airflow paths.

Pressure Difference (ΔP): Enter the pressure difference for the data point.

Flow Rate (F): Enter the flow rate for the data point.

Pressure Exponent (n): The flow exponent is a guess used to fit the single point to the powerlaw model. A value of 0.6 to 0.7 is typical for an infiltration element. A value of 0.5 is more appropriate for large openings.

Description: Field for entering a more detailed description of the specific airflow element.

Icon: Choose the style icon appropriate for the specific airflow element. Note that the icon has no effect on the simulations; the choice is based on user preference. There are two icon options for this Powerlaw model, one for a large opening and one for a small opening.

Powerlaw Model: Test Data (2 points)

This airflow element allows you to provide two test data points that ContamW reduces to the powerlaw model by calculating the flow coefficient based on input values (See Fitting Powerlaw Coefficients in the Theoretical Background section).

Name: Enter the name you want to use to identify the airflow element. The airflow element will be saved within the current project and can be associated with multiple airflow paths.

Pressure Difference (ΔP): Enter the pressure difference for each of the two data points.

Flow Rate (F): Enter the flow rate for each data point.

Description: Field for entering a more detailed description of the specific airflow element.

Icon: Choose either the small or large opening icon as appropriate for the specific airflow element. The icon has no effect on the simulations.

Powerlaw Model: Stairwell

This airflow element allows you to enter a description of a stairway, which ContamW converts to a powerlaw relationship assuming a pressure exponent of 0.5. A stairwell will normally be modeled as a vertical series of zones connected by low resistance openings (this stairwell flow element) through the floors. The resistance is based on a fit to experimental data [Achakji & Tamura 1998]. This model should only be used between zones on different levels of a building (paths through floors), not between zones on the same level (paths through walls).

Name: Enter the name you want to use to identify the airflow element. The airflow element will be saved within the current project and can be associated with multiple airflow paths.

Distance Between Levels: The vertical distance between doorways of the stairwell. This is typically equal to the “Distance to level above” of the level below the path with which this element is to be associated.

Cross-sectional area: This is the horizontal, cross-sectional, area of the shaft.

Density of People: A large number of people in the stairwell influence the flow resistance of the stairwell. The experiment used densities of 1, 2, and 3 people per square meter. These are the only units available in ContamW. The following conversion is provided for your convenience.

$$1 \text{ m}^2 = 10.76 \text{ ft}^2$$

Stair Treads: There are two options for this field: open tread and closed tread. This refers to the front of the tread whether or not the tread is open effects flow.

Description: Field for entering a more detailed description of the specific airflow element.

Icon: Choose either the small or large opening icon as appropriate for the specific airflow element. The icon has no effect on the simulations.

Powerlaw Model: Shaft

This airflow element allows you to enter a description of a shaft, and ContamW converts the information to a powerlaw relationship assuming a pressure exponent of 0.5. A shaft will normally be modeled as a vertical series of zones connected by low resistance openings (this shaft airflow element) through the floors. The resistance is based on a conduit friction model using the Darcy-Weisbach relation and Colebrook’s equation for the friction factor [ASHRAE 2005, p2.7].

Name: Enter the name you want to use to identify the airflow element. The airflow element will be saved within the current project and can be associated with multiple airflow paths.

Distance Between Levels: This is the distance between openings for the stairwell.

Cross-sectional area: This is the horizontal, cross-sectional, area of the shaft, not the opening.

Perimeter: The perimeter of the horizontal cross-section of the shaft. This number is used in conjunction with the area to create the hydraulic diameter.

Roughness: This refers to the average size of the protrusions from the shaft wall into the airflow.

Description: Field for entering a more detailed description of the specific airflow element.

Icon: Choose either the small or large opening icon as appropriate for the specific airflow element. The icon has no effect on the simulations.

One-way Flow using Quadratic Models

ContamX performs airflow calculations using the mass flow quadratic relationship ($\Delta P = aF + bF^2$) between mass flow rate and pressure difference across a flow path for the following types of airflow elements. Where F is the mass flow rate, ΔP is the pressure difference across a flow path, and a and b are flow coefficients.

Quadratic Model: $\Delta P = aQ + bQ^2$

This model allows you to directly enter the coefficients, a and b, into the volume flow version of the quadratic flow model. Where ΔP is the pressure drop and Q is the volumetric flow rate.

Name: Enter the name you want to use to identify the airflow element. The airflow element will be saved within the current project and can be associated with multiple airflow paths.

Coefficients: The coefficients must be expressed in SI units because of the way ContamW handles unit conversions. The units of the coefficients at standard conditions are as follows:

$$a \left[\text{Pa} \cdot \text{s}/\text{m}^3 \right]$$

$$b \left[\text{Pa} \cdot (\text{s}/\text{m}^3)^2 \right]$$

Description: Field for entering a more detailed description of the specific airflow element.

Icon: Choose either the small or large opening icon as appropriate for the specific airflow element. The icon has no effect on the simulations.

Quadratic Model: $\Delta P = aF + bF^2$

This model allows you to directly enter the coefficients, a and b, into the mass flow version of the quadratic flow model. Where ΔP is the pressure drop and F is the mass flow rate.

Name: Enter the name you want to use to identify the airflow element. The airflow element will be saved within the current project and can be associated with multiple airflow paths.

Coefficients: The coefficients must be expressed in SI units because of the way ContamW handles unit conversions. The units of the coefficients are as follows:

a [Pa · s/kg]

b [Pa · (s/kg)²]

Description: Field for entering a more detailed description of the specific airflow element.

Icon: Choose either the small or large opening icon as appropriate for the specific airflow element. The icon has no effect on the simulations.

Quadratic Model: Crack Description

This model employs a quadratic relationship of the form $\Delta P = aQ + bQ^2$ where Q is the volume flow rate [Baker, Sharples, and Ward 1987]. CONTAM uses the mass flow version of that formula: $\Delta P = aF + bF^2$ where ΔP is the pressure drop and F is the mass flow rate.

Name: Enter the name you want to use to identify the airflow element. The airflow element will be saved within the current project and can be associated with multiple airflow paths.

Crack Dimensions:

Length: The overall length of the crack

Width: Width of the crack

Depth: The distance along the direction of airflow

Number of Bends: The number of bends in the flow path.

Description: Field for entering a more detailed description of the specific airflow element.

Quadratic Model: Test Data (2 points)

Measured data (2 points) is reduced to the mass flow version of the quadratic model.

Name: Enter the name you want to use to identify the airflow element. The airflow element will be saved within the current project and can be associated with multiple airflow paths.

Data: Data consists of two sets of pressure drops and corresponding flow rates.

Description: Field for entering a more detailed description of the specific airflow element.

Icon: Choose either the small or large opening icon as appropriate for the specific airflow element. The icon has no effect on the simulations.

Two-way Flow Models

These models enable you to simulate openings through which two-way airflow might occur (e.g. doorways and open windows). When viewing results of these models, the bidirectional flow will be indicated on the SketchPad if it occurs. You can also use ContamW's airflow plotting feature to plot both airflows or to simply plot the net airflow. There must be a density difference across the associated airflow path in order for bidirectional flow to occur across these types of airflow elements. See Doorways (Large Openings) in the Theory section for a more detailed explanation.

The *Relative Elevation* of flow paths that implement these models should be set to the bottom of the flow path as opposed to the mid-height used for other element types.

Two-way Flow Model: Single Opening

This is a model for flow through large openings such as doorways through which air can flow in two directions simultaneously throughout the opening.

Name: Enter the name you want to use to identify the airflow element. The airflow element will be saved within the current project and can be associated with multiple airflow paths.

Height: The overall height of the opening, not to be confused with the *Relative Elevation* of flow paths with which the element is associated.

Width: The width of the opening.

Flow Coefficient: An experimentally determined value. Experiments by Weber and Kearney [Weber and Kearney 1989] have shown the default value of 0.78 to work well for most applications.

Minimum Temperature Difference for Two-Way Flow: A two-way flow is driven by the temperature (actually air density) difference between the two zones. When this temperature difference approaches zero the algorithm used for solving the flow tends towards a division by zero problem. To avoid this undefined situation the two-way model reverts to a one-way power law model at this "minimum temperature difference" using the opening size to define the orifice at ΔT set in this field. ContamW uses a default value of 0.01 °C.

Description: Field for entering a more detailed description of the specific airflow element.

Icon: Choose either the small or large opening icon as appropriate for the specific airflow element. The icon has no effect on the simulations.

Two-way Flow Model: Two-opening

This model accounts for two-way flow due to the stack effect acting over the height of a tall opening. It uses two power law flow models at different heights to approximate a single tall opening.

Name: Enter the name you want to use to identify the airflow element. The airflow element will be saved within the current project and can be associated with multiple airflow paths.

Height: The overall height of the opening, not to be confused with the *Relative Elevation* of flow paths with which the element is associated.

Width: The width of the opening.

Flow Coefficient: This is an experimentally determined value. Experiments by Weber and Kearney have shown the default value of 0.78 to work well for most applications.

Description: Field for entering a more detailed description of the specific airflow element.

Icon: Choose either the small or large opening icon as appropriate for the specific airflow element. The icon has no effect on the simulations.

Backdraft Damper Flow Models

The Backdraft Damper models make it possible to model a feature that has different flow resistances depending on the direction of the pressure drop, with greatly reduced (or zero) flow in one direction. Note that there are similar models available for duct flow elements.

Backdraft Damper Model: $Q = C(\Delta P)^n$

This is the volumetric flow form of the backdraft damper airflow model.

Name: Enter the name you want to use to identify the airflow element. The airflow element will be saved within the current project and can be associated with multiple airflow paths.

Pressure Difference: Enter a flow coefficient and exponent for both the positive and negative flow directions of the flow path. $\Delta P > 0$ refers to the pressure difference across the flow element that would lead to an airflow in the positive flow direction (See Airflow Path Properties) and $\Delta P < 0$ would lead to a flow in the opposite direction.

Flow Coefficients: The coefficients may only be expressed in SI units due to the conversion method used. Use the following conversion to convert from IP units to SI units.

To convert from units of $\frac{cfm}{in H_2O^n}$ to the SI units of $\frac{m^3/s}{Pa^n}$ multiply by $\frac{1}{2119 \times 249^n}$

Flow Exponents: Flow exponents vary from 0.5 for large openings where the flow is dominated by dynamic effects, and 1.0 for narrow openings dominated by viscous effects. Measurements usually indicate a flow exponent of 0.6 to 0.7 for typical infiltration openings.

Description: Field for entering a more detailed description of the specific airflow element.

Backdraft Damper Model: $F = C(\Delta P)^n$

This is the mass flow form of the backdraft damper airflow model.

Name: Enter the name you want to use to identify the airflow element. The airflow element will be saved within the current project and can be associated with multiple airflow paths.

Pressure Difference: Enter a flow coefficient and exponent for both the positive and negative flow directions of the flow path. $\Delta P > 0$ refers to the pressure difference across the flow element that would lead to an airflow in the positive flow direction (See Airflow Path Properties) and $\Delta P < 0$ would lead to a flow in the opposite direction.

Flow Coefficients: The coefficients may only be expressed in SI units due to the conversion method used. Use the following conversion to convert from IP units to SI units.

To convert from units of $\frac{lbm/s}{in H_2O^n}$ to the SI units of $\frac{kg/s}{Pa^n}$ multiply by $\frac{0.4536}{249^n}$

Flow Exponents: Flow exponents vary from 0.5 for large openings where the flow is dominated by dynamic effects, and 1.0 for narrow openings dominated by viscous effects. Measurements usually indicate a flow exponent of 0.6 to 0.7 for typical infiltration openings.

Description: Field for entering a more detailed description of the specific airflow element.

Fan and Forced-Flow Models

These airflow element models enable you to easily create forced airflow elements as airflow paths between two zones, as opposed to implementing a duct model or simple air handling system. There are three types of fan models to choose from: constant mass flow rate, constant volumetric flow rate, and fan performance curve.

Fan Model: Constant Mass Flow Fan

This model describes an airflow element having a constant mass flow rate. This airflow element will provide the specified constant mass flow rate regardless of the density of the air being delivered by the fan.

Name: Enter the name you want to use to identify the airflow element. The airflow element will be saved within the current project and can be associated with multiple airflow paths.

Design (maximum) Flow: Enter the maximum mass flow rate. This value can be modified by the path schedule.

Description: Field for entering a more detailed description of the specific airflow element.

Fan Model: Constant Volume Flow Fan

This model describes an airflow element having a constant volume flow rate. The design flow rate you specify is in terms of standard air. CONTAM uses a density of 1.204 kg/m³ based on conditions specified by ASHRAE for dry air: 101.325 kPa and 20 °C (ASHRAE 2004, p 18.4). Therefore, if actual conditions during simulation do not match these standard conditions, the results will differ from specified flow due to differences in air density.

Name: Enter the name you want to use to identify the airflow element. The airflow element will be saved within the current project and can be associated with multiple airflow paths.

Design (maximum) Flow: Enter the maximum volume flow rate. This value can be modified by the path schedule.

Description: Field for entering a more detailed description of the specific airflow element.

Fan Model: Fan Performance Curve

This airflow element allows you to create a fan based on a fan performance curve. You provide ContamW with a set of pressure rise and airflow rate data points, and ContamW performs a polynomial curve fit to the data. Airflow rates are specified in mass flow units for standard air.

Name: Enter the name you want to use to identify the airflow element. The airflow element will be saved within the current project and can be associated with multiple airflow paths.

Fan off conditions: These values specify how the fan is modeled when the speed of the fan is insufficient to provide a flow.

Cut-off Ratio: The *fan speed ratio* is the actual speed (RPM) of the fan divided by the rated

speed of the fan (fan is defined for a speed ratio of 1.0). When this ratio is below the specified cut-off limit the fan is modeled as a simple orifice based on the "equivalent orifice" defined for this fan element. During a simulation, ContamX obtains the actual ratio from the control value being applied (in the form of a schedule or control signal) to the airflow path with which this fan element is associated and compares it to the *cut-off ratio*. See Chapter 18 Fans in [ASHRAE 2004].

Equivalent Orifice: The size of the orifice used to model the fan opening when the control value drops below the specified limit.

Description: Field for entering a more detailed description of the specific airflow element.

Fan Performance Data: Click the "Edit Fan Curve Data" to input/modify the fan curve for this airflow element. A detailed explanation of this data is given in the next section, Fan Performance Data.

Fan Performance Data

Between four and ten data points are required for ContamW to fit a cubic polynomial to create a performance curve for the fan. The fitted curve may not contain a point of contraffecture; if it does an error message will be displayed. If this error message is displayed the pressure rise data must then be modified to eliminate the point of contraffecture.

Entering Fan Curve Data Points

You only need to fill in the "Flow rate" and "Pressure rise" data entry when entering fan curve data. When adding a new data point, the flow rate must be entered in the "Flow rate" edit box and the pressure rise must be entered in the "Pressure rise" edit box. Use the "Revised dP" box when making changes to a pressure rise that has already been entered. This will preserve the original data point but use the revised value when performing the curve fit. After entering the flow rate and the pressure rise into the appropriate edit boxes, the next step is to press the "<< Insert <<" button. Once you have entered four data points a cubic fit will automatically be generated and displayed as a line on the graph.

Revising Fan Curve Data Points

To edit existing data points, highlight the line of data to revise in the list of Fan Curve Data to the left of the Insert, Replace and Delete buttons. The values from that data set will then appear in the edit boxes to the right of the buttons.

If you need to modify the flow rate, enter the new flow rate in the "Flow rate" edit box and then press the "<< Replace <<" button. The new data will now replace the old data.

If the Pressure Rise needs to be modified there are two available options:

1. Replace the data point completely. To overwrite an existing pressure rise and refit the fan curve, select the data set to be modified by highlighting it in the data list and then type the new pressure rise in the "Pressure rise" edit box then press the "<< Replace <<" button.
2. Change the point used to fit the cubic polynomial but leave the marker for the original data point intact. With this option you may "tweak" the curve in the event there is some issue with the originally entered fan curve, such as a point of contraffecture.

Cubic Spline Models

Cubic spline models enable you to create airflow elements based on a curve fit to a user-defined set of data points. The cubic spline fit used to generate the curve guarantees a first-order differentiable relationship between flow and pressure as required by the CONTAM solver. During simulation, the sign of the pressure difference will be based on the *Positive Flow Direction* defined for the flow paths with which the spline elements are associated. For example, a drop in pressure in the direction of positive flow (i.e. Pressure in from zone – Pressure in to zone > 0) will utilize a positive pressure from the spline data.

There are some basic requirements for each of these elements. They require a minimum of four data points, so a curve will not be displayed until the minimum number of points is entered. All models require that the slope be greater than zero for all segments of the curve fit. If there is an error in the curve fit when you click the OK button, an error message will be displayed indicating the offending segments of the curve and a reason for the error. Segments are numbered from zero to the number of data points minus one. For example – “seg 0: $y' \leq 0$ ” will be displayed if the segment between the first two data points has a slope less than or equal to zero.

Name: Enter the name you want to use to identify the airflow element.

Description: Field for entering a more detailed description of the specific airflow element.

Curve Data: Create and edit the list of data points to define the curve for the airflow element. The type of cubic spline element you are editing determines the independent and dependent variables for the curve. The labels of the list, data entry fields and associated units will change accordingly, as will the axes of the plot. Use the data entry fields along with the “Add” button to create new and edit existing data points. Use the “Delete” button to remove the currently selected data point from the list.

Icon: Choose either the small or large opening icon as appropriate for the specific airflow element. The icon has no effect on the simulations.

Cubic Spline: F vs P

Mass flow as a function of Pressure drop across the element.

Cubic Spline: Q vs P

Volume flow as a function of Pressure drop across the element.

Cubic Spline: P vs F

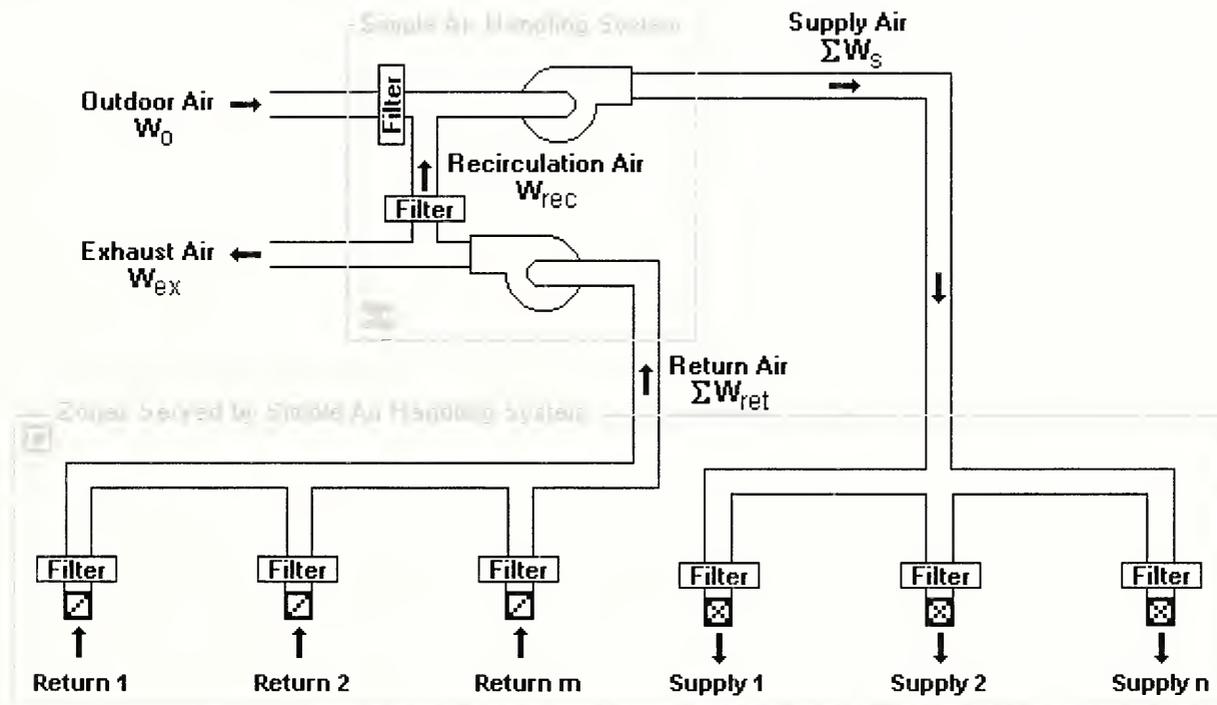
Pressure drop as a function of Mass flow through the element.

Cubic Spline: P vs Q

Pressure drop as a function of Volume flow through the element.

3.8 Working with Simple Air Handling Systems

The *simple air-handling system* (AHS) provides a convenient means of incorporating an air-handling system into a building without having to draw and define an entire duct system. Each simple air-handling system consists of two implicit airflow nodes or zones (*supply* and *return* sub-systems), three implicit flow paths (*recirculation*, *outdoor*, and *exhaust*), and multiple *zone supplies* (inlets to zones) and *zone returns* (outlets from zones) that you can place within zones throughout the building. You specify the airflow rates of each *supply* and *return* point. *Simple air-handling systems* do not require you to associate both zone supplies and zone returns with them. You may use the *simple air-handling system* to only supply outdoor (ambient) air to a building or to only exhaust air from the building. The following figure shows a schematic representation of a simple air-handling system.



□ Determination of System Airflows

ContamX determines the airflow rates associated with each air-handling system according to the following algorithm. More detailed explanations for the values that you must input for this algorithm are explained in the Defining Air-handling Systems section that follows.

First, all of the user-defined supply (supply air to the zones) and return (return air from the zones) airflow (mass) rates are summed.

$$\Sigma W_s \equiv \text{the sum of all supplies}$$

$$\Sigma W_{ret} \equiv \text{the sum of all returns}$$

The amount of outdoor air that the system requires is then determined by

$$W'_o = \max(f_o \cdot \Sigma W_s, \min(\Sigma W_s, W_{o_min})),$$

where, $f_o \equiv$ the fraction outdoor air = W_o / W_s

You input this parameter as the *Outside Air Schedule* of a simple air handling system (See Air Handling System – AHS Properties), and

W_{o_min} ≡ the "minimum outdoor airflow" parameter that you input.

The rate at which air is recirculated via the implicit recirculation flow path, W_{rec} , is determined by

$$W_{rec} = \min(\Sigma W_{ret}, \Sigma W_s - W'_o).$$

The rate at which outdoor air is brought in by the system via the implicit outdoor airflow path, W_o , is determined by

$$W_o = \Sigma W_s - W_{rec}.$$

And the rate at which exhaust airflows to the ambient via the implicit exhaust flow path, W_{ex} , is determined by

$$W_{ex} = \Sigma W_{ret} - W_{rec}.$$

Based on this algorithm the amount of outdoor air the system will provide, W_o , will be between W_{o_min} and $f_o \cdot \Sigma W_s$ as long as the demand for supply air, ΣW_s , is sufficient to provide this value. Otherwise the system will provide ΣW_s . When the sum of the supply flows, ΣW_s , exceeds the sum of the return flows, ΣW_{ret} , the balance is made up of *outdoor* air. Any excess return air is exhausted via the implicit exhaust flow path.

3.8.1 Creating Simple Air-handling Systems

You create *simple air-handling systems* by placing an *Air-handling System (AHS) icon* on the SketchPad along with some combination of *supply* and/or *return icons* (See Drawing Building Component Icons). You may place an *AHS icon* on any blank cell of the SketchPad on any level. You do not have to place it within a zone. However, you must place *Supply icons* and *Return icons* within a zone. The pop-up menu selections will be disabled (grayed out) if you pop-up on a location where ContamW does not allow these icons to be placed.

3.8.2 Deleting Air-handling Systems

You delete icons related to simple air-handling systems using the icon deletion procedure (See Deleting Building Component Icons). If you delete an AHS icon, , all of the supplies and returns associated with that system will be deleted. You will be warned of this when you attempt to delete the icon. However, you can delete individual supplies and returns without affecting any other icons. Deleting any AHS-related icons from the SketchPad does not delete any filter, schedule, or kinetic reaction elements that you may have defined for the AHS. These flow elements can only be deleted using the CONTAM Library Manager, accessible through the **Data** menu for each type of data element.

3.8.3 Modifying Air-handling Systems

To modify the parameters of a simple air-handling system, you use the icon definition procedure (See Defining Building Component Icons) to display its properties and make the desired changes. You can move air-handling system icons and move and copy supply and return icons; however, you cannot copy air-handling system icons.

❑ Moving Air-handling System Icons

You can move AHS icons using the **Cut** and **Paste** functions of the **Edit** menu. ContamW will only allow you to move an AHS icon within the same level upon which it is located.

❑ Moving Supply and Return Icons

Once you define a supply or return icon, you can move them using the **Cut** and **Paste** functions of the **Edit** menu. ContamW will only allow you to move a supply or return icon within the same zone in which it is currently located.

❑ Copying Supply and Return Icons

Once you define a supply or return icon using the **Copy** and **Paste** functions of the **Edit** menu. You can copy supply and return icons into any zone on any level.

3.8.4 Defining Air Handling Systems

After you place an *AHS-related* icon (*AHS*, *Supply* and *Return*) onto the SketchPad you must define them using the icon definition procedure (See Defining Building Component Icons) to display and edit the properties of the icon. This will display the property sheet associated with the icon. Detailed descriptions of all AHS properties are given in the Air-handling System Properties section of this manual.

This list shows the various AHS-related icons and provides a brief description of each.

Icon	Description
	Air-handling system
	Room air supply (inlet) of an air-handling system
	Room air return (outlet) of an air-handling system

❑ Air-handling System Icon

The properties associated with each AHS include outdoor air intake data, supply and return contaminant information, and system filters. Outdoor air intake data consists of a minimum outdoor air intake rate, W_{oa_min} and a schedule for varying the outdoor air intake rate as a percentage of the supply airflow rate. The schedule will set the value of f_o in the algorithm above. The default value of f_o is 1.0 (i.e., 100 % outdoor air) if you don't use a schedule. Supply and return system data refers to the implicit zones that are created for each AHS. Just as for any zone you create, you can provide a zone volume, initial zone concentration and a kinetic reaction for each contaminant defined for the project. You can also associate contaminant filters with the implicit outdoor and recirculation flow paths.

❑ Supply and Return Icons

The properties associated with each AHS include airflow rate and contaminant filtration information. You must enter a design airflow rate for each supply and return icon that you create. Enter the design maximum value that you anticipate requiring. You can also provide a schedule to the design airflow rate to adjust down the design flow rate. As with other airflow paths, you can provide a contaminant filter to these items. Be careful not to place filters in both the individual returns and the recirculation path of the air handling system unless this is what you really intend to do.

3.8.4.1 Air Handling System Properties

This section provides detailed descriptions of the specific properties of simple air-handling systems. Each of the following subsections are the context-sensitive help topics that you can access by pressing **F1** when working with property pages of the "Simple Air-handling System Properties," "Supply Point Properties," and "Return Point Properties" property sheets.

Air-handling System – AHS Properties

System Name: The name used to identify the Air-handling System. Air-handling system names must be unique from one another.

Minimum OA Flow: This is the minimum *outdoor air* introduced into the AHS *supply* flow. If the sum of the *return* flows plus the *minimum outside airflow* exceeds the total supply flow then the excess return air is vented outside. Similarly if the return flow plus the minimum outside airflow do not sum to the supply flow then outside air is added to make up the difference.

Outside Air Schedule: This is a schedule of the fraction of outdoor air, f_o , to be introduced to the simple air handling system. If the schedule is set to "<none>", then the fraction of outdoor air defaults to 100 % (100 % outdoor air, 100 % exhaust and no recirculation). You define a new Outside Air schedule by pressing the "New Schedule" button and following the procedure for editing a Week Schedule.

Name: This drop down box contains the names of all previously defined schedules within the current project. You can select an existing schedule from the list and use the "Edit Schedule" button to view and modify its properties, create a "New Schedule" and even select a schedule from a CONTAM Library via the "Library..." button. You can also choose to not apply a schedule by selecting "<none>" from the list.

Description: Displays the detailed description of the selected outdoor air schedule.

Outdoor Air Inlet Location: These are the absolute X, Y and Z coordinates and units that define the location of the outdoor air intake of the selected air handling system. These values are required when using a *Wind Pressure and Contaminant (WPC)* file (see Wind Pressure and Ambient Contaminant Files).

NOTE: You must first check either the *Wind Pressures* or *Contaminant Concentrations* check boxes on the *Wind Pressure and Contaminants (WPC) File Parameters* dialog box to be able to access these coordinate input fields (see WPC Usage Parameters).

Air-handling System – Supply System Properties

System Volume: Enter a value for the supply sub-system (implicit supply zone) of the simple air handler (e.g., supply-side duct work etc.). This volume is very similar to a zone volume and will be used in the simulation of contaminant transport.

Contaminant Data: All of the defined contaminants in the project will be displayed in this list box.

Initial Concentration: Select a contaminant from the list and enter the *initial contaminant concentrations* for the *supply air sub-system* of this *simple air handling system*. Note that only those species you've selected to be contaminants (to use during simulation) appear in the list. If a species for which you wish to set an initial concentration does not appear in the list you must set the "Use in simulation" property of the corresponding species to be true (See Creating Species

and Contaminants). You can reset these initial values through the Run Control Properties of the Simulation Parameters.

NOTE: Changing the number of contaminants, i.e. those species used in the simulation, will reset the *initial concentrations* of all zones to the default contaminant concentrations. You can reset all of these concentrations via the Run Control Properties as indicated above or individually here.

Sum of Non-trace Initial Concentrations: This value is provided as a check for you to insure that you input a reasonable set of initial values when using *non-trace contaminants*. This value should be very close to 1.0.

Kinetic Reaction: You define a new kinetic reaction by pressing the "New Reaction" button and then completing the kinetic reaction matrix. You can edit an existing matrix by selecting the proper matrix from the drop down box and pressing the "Edit Reaction" button. You must have previously defined contaminants in order to implement a kinetic reaction.

Description: Field that shows a more detailed description of the specific reaction if entered by the user. This description can be modified by pressing the "Edit Reaction" button next to the description field and then changing the description field for the reaction.

Name: This contains a list of names of all previously created reactions. The reaction selected from this box will define the behavior of the specific supply system being modified.

Air-handling System – Return System Properties

System Volume: Enter a value for the return sub-system (implicit return zone) of the simple air handler (e.g., supply-side duct work etc.). This volume is very similar to a zone volume and will be used in the simulation of contaminant transport.

Contaminant Data: All of the defined contaminants in the project will be displayed in this list box.

Initial Concentration: Select a contaminant from the list and enter the *initial contaminant concentrations* for the *supply air sub-system* of this *simple air handling system*. Note that only those species you've selected to be contaminants (to use during simulation) appear in the list. If a species for which you wish to set an initial concentration does not appear in the list you must set the "Use in simulation" property of the corresponding species to be true (See Creating Species and Contaminants). You can reset these initial values through the Run Control Properties of the Simulation Parameters.

NOTE: Changing the number of contaminants, i.e. those species used in the simulation, will reset the *initial concentrations* of all zones to the default contaminant concentrations. You can reset all of these concentrations via the Run Control Properties as indicated above or individually here.

Sum of Non-trace Initial Concentrations: This value is provided as a check for you to insure that you input a reasonable set of initial values when using *non-trace contaminants*. This value should be very close to 1.0.

Kinetic Reaction: You define a new kinetic reaction by pressing the "New Reaction" button and then completing the kinetic reaction matrix. You can edit an existing matrix by selecting the proper matrix from the drop down box and pressing the "Edit Reaction" button. You must have previously defined contaminants in order to implement a kinetic reaction.

Description: Field that shows a more detailed description of the specific reaction if entered by the user. This description can be modified by pressing the "Edit Reaction" button next to the description field and then changing the description field for the reaction.

Name: This contains a list of names of all previously created reactions. The reaction selected from this box will define the behavior of the specific supply system being modified.

Air-handling System – Filter Properties

As the names suggest, *outdoor air filters* affect the air brought in by the simple air-handling system from outside the building, while *recirculation air filters* affect the return air being circulated back through the air handler. You may either create, edit or delete the filters associated with the implicit outdoor air and recirculation flow paths of the simple air-handler. Creating a new filter or modifying an existing one will display the Filter dialog box. Use this dialog box to create, edit and filter elements as needed. For an explanation of contaminant filters see Contaminant Filters in the Using CONTAMW section of the manual.

Supply/Return Point - System Properties

Design Flow Rate: This value is design flow rate for the supply/return element (i.e. the maximum flow rate specified for a particular vent).

AHS: You must associate each supply and return point with an existing air-handling system.

Name: This drop down box contains the names of all previously created Air-handling Systems. The AHS selected from this box will define the behavior of the specific supply/return system being modified. Once an existing AHS is selected it may be edited by pressing the "Edit AHS" button.

Description: Field that shows a more detailed description of the specific schedule if entered by the user. This description can be modified by pressing the "Edit AHS" button next to the description field and then changing the description field for the AHS.

Location: These are the X, Y and Relative Elevation (Z) coordinates and units for the selected supply/return point. X and Y are absolute coordinates and Rel Elevation is relative to the level on which the supply/return point is located.

These values are required for any supply/return point that is located within a *1D convection/diffusion zone* (see 1D Zone Data). These coordinates will be verified by ContamW against the axis of the 1D zone in which it is located. This verification will be performed automatically when you select either the **Run Simulation** or **Create a ContamX Input File** command from the **Simulation** menu.

Supply/Return Point - Filter and Schedule Properties

Filter: You may either create, edit or delete the filter associated with a supply/return point. This filter will act upon the associated contaminant(s) as air flows to/from the zone in which the supply/return point is located. Creating a new filter or modifying an existing one will display the Filter dialog box. Use this dialog box to create, edit and filter elements as needed. For an explanation of contaminant filters see Contaminant Filters in the Using CONTAMW section of the manual.

Schedule: You can associate a schedule with the supply/return point to control the fraction of the design airflow rate that flows to/from the zone in which the supply/return point is located according to the time of day and day of the week.

Description: Field that shows a more detailed description of the specific schedule if entered by the user. You can modify this description by pressing the "Edit Schedule" button next to the description field and then changing the description field for the schedule.

Name: This drop down box contains the names of all previously created schedules. The schedule selected from this box will define the behavior of the specific supply/return system being modified.

3.9 Working with Ducts

As an alternative to the simple air-handling system, CONTAM allows you to model HVAC systems using detailed duct systems. Duct systems are made up of duct segments, junctions, terminal points, and *forced air components*. A *duct segment* is a section of a duct between any two junctions, terminal points or combination of the two. A terminal point is an endpoint of a duct segment, and a junction is the connection between two duct segments. Only horizontal segments can be displayed on the ContamW SketchPad, but vertical segments can be implemented as well. Vertical segments are connected with special junctions and terminal points that reveal the location of the vertical segments on the SketchPad.

You draw the layout of the duct system, and ContamW will insert junctions and terminal points where appropriate. Once you have drawn a duct system, you must define the drawing by providing detailed segment, junction and terminal point information. You must provide each duct segment with specific information that describes the flow characteristics and physical dimensions of that segment. Much of this information is in the form of *duct flow elements* that you must create when defining the duct segments. The details of drawing and defining a duct system are described in the following sections.

3.9.1 Duct Flow Elements

Each duct segment must refer to a *duct flow element*. *Duct flow elements* describe the mathematical relationship between flow through and pressure drop along the duct, the flow resistance or forced flow characteristics, cross-sectional geometry, and optional leakage per unit length of a duct. ContamW provides you with several different mathematical models or element types to choose from. Each of these duct flow element types is described in detail in the Duct Segment Properties section. While every duct segment must refer to a single *duct flow element*, multiple duct segments can refer to the same *duct flow element*. *Duct flow elements* can also be stored within a CONTAM library file and shared between different CONTAM project files.

3.9.2 Drawing Ducts

You draw ducts using the duct drawing tool previously described in the Working with the SketchPad section. After you complete the "finalize drawing object" stage of duct drawing, the completed duct will appear as an undefined set of red duct icons. The color red indicates that the duct is not yet defined. Once you have defined the duct components, they will be displayed in blue. A directional duct segment icon will be displayed within each duct segment, and each junction and terminal point will be replaced with an icon indicating the type of junction or terminal point you have defined.

□ Drawing Duct Segments

ContamW provides certain constraints when drawing ducts to insure that a valid duct system will be drawn that conforms to the underlying model of nodal equations upon which CONTAM is based. When drawing a duct, you cannot cross over a line that you are currently drawing. Ducts can cross over walls, but you cannot draw ducts over any other building component icons. After each duct segment is drawn, the ends of the segment will be automatically replaced by either an undefined terminal point or junction icon. You can only begin or end a duct segment drawing in an unoccupied SketchPad cell or on an undefined duct icon. For this reason, you are advised to draw the entire duct system, on a given level, before you define the individual duct segments. If

you cross over a previously drawn duct segment, either defined or undefined, while drawing a new duct segment, no junction will be placed where the two segments cross.

Vertical duct segments do not appear directly on the SketchPad, because the SketchPad displays only plan-view drawings. Vertical duct segments are associated with vertically connected junctions or terminal points. These vertical junction and terminal icons provide access to the properties of the vertical segments that are located below the level upon which the vertical junction or terminal appears.

□ **Drawing Duct Junctions**

Undefined horizontal duct junction icons are drawn automatically at the intersections of the ducts after you finalize drawing a duct. Each time you begin or end drawing a duct segment upon an undefined duct segment, junction or terminal point, ContamW will automatically provide an undefined junction icon if there is not yet one at the junction location.

All vertical junctions are connected to vertical duct segments that are located below the level upon which the vertical junction appears. You can create a vertical junction that is isolated from other ducts on a level that is connected either up, down or both up and down to ductwork on adjacent levels. To do this, finalize the drawing object immediately after setting the initial location of the duct object. To do this, you simply press **LMB** (double click) or ↵ twice at the desired location on the SketchPad when you have the duct drawing tool selected. This will place a single undefined terminal point icon within the SketchPad cell. You can later define this undefined terminal point icon as the specific type of junction that you need and associate it with a vertical duct segment.

□ **Drawing Duct Terminal Points**

Undefined horizontal terminal point icons are automatically drawn by ContamW at the end of each duct segment that does not end on another previously drawn duct icon.

You can also create terminal points that are isolated from other ducts on a level but connected either up or down to ductwork on an adjacent level by finalizing the drawing object immediately after setting the initial location of the duct object. To do this, you simply press **LMB** (double click) or ↵ twice at the desired location on the SketchPad when you have the duct drawing tool selected. This will place a single undefined terminal point icon within the SketchPad cell. You can later define this terminal point as the specific type that you need and associate it with a vertical duct segment.

□ **Coloring Duct Systems**

You can use the *Duct System Coloring Tool* to automatically color sections of the duct system that are associated with fans (forced flow elements) within the duct. You access the *Duct System Coloring Tool* via the **View → Duct Coloring...** menu command. This will display the Duct System Coloring dialog box that contains a list of all duct segments that contain fans (forced flow elements). Select the fan whose ducts you wish to color. Set the inlet side, outlet side and fan segment colors, then click the Set Colors button to have ContamW trace the duct system on the inlet and outlet sides of the fan segment and set the colors.

Use the **View → Color Ducts** menu command to toggle the colors in which ducts are displayed on the SketchPad between the custom duct colors and the default color.

3.9.3 Deleting Ducts

Deleting currently defined ducts is a two-step process. That is, if you want to delete a duct segment, junction or terminal, you must first undefine the item. *To undefine a duct segment* you must highlight the special duct segment icon (that indicates the positive flow direction) and press the **Delete** key or select Delete from the Edit menu. This will highlight the entire duct segment from its two end-points and prompt you to confirm the undefine operation by displaying a message box. Once you undefine a duct segment, it will again be displayed in red and the special duct segment icon will be converted back to a straight segment icon. *To delete the duct segment* you highlight any portion of the undefined segment or associated undefined terminal and perform the deletion. The entire segment to be deleted will be highlighted, and you will be asked to confirm the deletion.

There are several items you should consider when deleting ducts. You cannot delete an undefined junction that is still connected to a defined duct segment. If you delete a junction that is between two undefined segments, the junction will be removed to form a single segment. You cannot delete a terminal without deleting its associated segment.

Deleting the ducts defined on the SketchPad does not delete any duct flow elements that you may have defined. These flow elements can only be deleted using the CONTAM Library Manager, accessible through the **Data** menu for each type of data element.

3.9.4 Modifying Ducts

In order to modify the layout of a duct system, you must undefine any portions to which you want to connect a new duct (i.e. form a junction). To modify the parameters of a duct segment, junction or terminal, you use the icon definition method to display its properties and make the desired changes. You do not have to undefine the item to change its properties, only to change the physical layout.

To remove a junction from between two segments, you should first undefine the two duct segments, remove the junction and then redefine the newly formed longer segment. To split a segment into two segments, double-click the **LMB** with the duct drawing tool on the undefined segment where you wish to place the junction.

3.9.5 Defining Ducts

After you draw a duct, you must define each duct segment, junction, and terminal point. Each of these duct components is defined using the icon definition procedure (See Defining Building Component Icons) to display and edit the properties of the component.

□ Defining Duct Segments

You define each duct segment by using the icon definition procedure on any portion of an undefined duct segment. This will display the "Duct Segment Properties" property sheet. Detailed descriptions of all duct segment properties are given in the Duct Segment Properties section of this manual. Once you have defined the properties of a duct segment, a special icon will be displayed indicating the positive flow direction of the duct segment (see Directional Duct Segment Icons). From now on, you use this icon to access the properties of the duct segment.

When defining a duct segment, you must associate the segment with a duct flow element. CONTAM combines duct flow element data with segment specific data such as length and

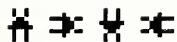
dynamic losses to determine the frictional resistance, volume, and leakiness of a particular duct segment.

You can also define contaminant filtering properties of duct segments. You can define a filter for each contaminant contained in your CONTAM project. A duct filter could be used, for example, to simulate the deposition of particles on the inside surface of a duct.

As previously mentioned, duct flow elements contain duct leakage information. The CONTAM model implements all leakage at the junctions and terminals of a duct segment. This means that half of the leakage associated with a duct segment occurs at each end of the segment. The leakage between a junction and the zone in which the junction is located (as determined on the SketchPad) is a function of the duct element leakage characteristics and the pressure difference between the junction and the zone. You should consider this leakage model when accounting for leakage of a duct that passes through multiple zones. You should put a least one junction (or terminal) in each zone within which you want to account for duct leakage.

□ Directional Duct Segment Icons

These are the icons that indicate a defined duct segment. The direction that the small arrow points indicates the positive flow direction of the duct segment.



□ Defining Duct Junctions and Terminals

You define each duct junction and terminal by using the icon definition procedure on a duct junction or terminal icon. This will display the "Duct Junction Properties" property sheet. If a terminal icon is located in the ambient zone, a Wind Pressure property page will be displayed to allow you to account for the effects of wind pressure on the exterior terminal. The Wind Pressure property page will not be displayed for junction and terminal icons located within non-ambient zones. Detailed descriptions of duct junction properties are given in the Duct Junction and Terminal Properties section of this manual. Once you have defined the properties of a duct junction, a special icon will be displayed indicating the specific type of junction you have selected (see the following list). From now on, you use this icon to access the properties of the duct junction.

□ Junction and Terminal Icons

Icon	Description
	Junction connected to ducts on the same level
	Junction connected to the vertical duct of a junction on the level above
	Junction and downward vertical duct connected to a junction on the level below
	Junction and downward vertical duct connected to a junction on the level below and connected to the vertical duct of a junction on the level above
	Terminal connected to a duct on the same level
	Terminal connected to the vertical duct of a junction on the level above
	Terminal and downward vertical duct connected to a junction on the level below

3.9.5.1 Duct Segment Properties

This section provides detailed descriptions of the specific duct segment properties. Each of the following subsections are the context-sensitive help topics that you can access by pressing **F1** when working with property pages of the "Duct Segment Properties" property sheet.

Duct Segment – Flow Element Properties

Duct Segment Number: This is the number that appears in the status bar when you highlight a duct segment icon. ContamW automatically assigns this number to each *duct segment* once they are defined. This number could change as you add and remove duct segments to or from a project. Each time you save a project, the duct segment numbers are reassigned. Numbers are assigned beginning on the top level, starting from the upper-left corner of the SketchPad, moving left-to-right and top-to-bottom of the SketchPad.

Duct Flow Element Name: You may select a previously defined airflow element from those contained in the Name list. Click the "New Element" button to define a new airflow element. To view or modify airflow element data click the "Edit Element" button. Every duct segment must have an airflow element associated with it. This field may not be left blank.

Model Summary: This summarizes the information associated with the duct flow element currently displayed in the Name field.

Duct Segment – Segment Properties

Segment Data:

Duct Segment length: Enter the length of the duct segment. This will be used along with the cross-sectional area to determine the volume of the duct segment.

Sum of loss coefficients: This is the sum of all dynamic loss coefficients due to junction losses and all fittings in the segment. This term only applies to the Darcy-Colebrook duct flow element model (See Ducts in the Airflow Analysis section).

Positive Flow Direction: In the case of fans and backdraft dampers it is necessary to know which direction is defined as positive. The arrows represent the positive direction for pressure drop and airflow.

Color: You can set the color of each segment individually here, or you can use the automated *Duct System Coloring Tool* to color sections of ducts related to forced flow elements within the duct system. You access the *Duct System Coloring Tool* via the **View → Duct Coloring...** menu command (See Drawing Ducts).

Segment Summary Information: This field displays information about the selected duct flow element.

Duct Segment – Filter & Schedule Properties

Filter: You may select a previously defined filter from those contained in the associated Name list. Click the "New Filter" button to define a new filter. Click the "Edit Filter" button to view or modify existing filter data. Setting this field to <none> indicates no filter for this airflow path. (See Contaminant Filters under Working with Contaminants)

Schedule: If there are previously defined schedules within the current project, you may select one from the Name list. Click the "New Schedule" button to define a new schedule. To view or

modify an existing schedule click the "Edit Schedule" button. Setting this field to <none> indicates no schedule. (See Working with Schedules)

Duct Flow Element – Shape, Size and Leakage

Duct Shape: You can implement several different shapes of ducts including: round, rectangular, oval or other. Select the shape you want and the appropriate dimensions will be made available for you to enter under the Duct Dimension parameters.

Duct Dimensions: The data entered here is dependent on the duct shape. For circular ducts, the diameter is the only parameter needed. Rectangular ducts require you to enter the width and height, and oval ducts require you to input the major and minor dimensions. The other option allows you to enter a perimeter and a flow area to define the dimensions for a duct of any other shape.

Leakage: Leakage of air between the duct and the surrounding zone may be described in terms of a leakage rate at a given pressure or by use of the leakage classification. If you enter a leakage rate and a pressure difference, the value for the leakage class is automatically calculated by ContamW and placed in the "Leakage Class" field. Similarly, if you input a leakage class, a default value of 250 Pa is used for the pressure difference, and a leakage rate will be calculated based on the pressure difference and the leakage class you entered. See [ASHRAE 2005, Chapter 35]

Duct Flow Element Types

You select the type of duct flow element from this page that you want CONTAM to utilize to model airflow through this particular duct segment.

The most common model is the Darcy-Weisbach relation and Colebrook's natural roughness function.

Darcy-Colebrook Model

Three versions of the *powerlaw model* are included:

Orifice resistance model

Resistance: $F = C(\Delta P)^n$ (mass flow)

Resistance: $Q = C(\Delta P)^n$ (volume flow)

Three types of *forced flow models* are included:

Fan - performance curve

Constant mass flow

Constant volume flow

The Backdraft Damper models make it possible to model a feature (e.g. a smoke control damper) that has different flow resistances depending on the direction of the pressure drop, with greatly reduced (or zero) flow in one direction.

Backdraft Damper: $F = C(\Delta P)^n$ (mass flow)

Backdraft Damper: $Q = C(\Delta P)^n$ (volume flow)

Darcy-Colebrook Model

This model uses the Darcy-Weisbach relation and Colebrook’s natural roughness function [ASHRAE 2005 p 35.7].

Name: Enter a unique name you want to use to identify the duct flow element. The element will be saved within the current project and can be associated with multiple duct segments.

Description: Field for entering a more detailed description of the specific duct flow element.

Roughness: The flow resistance due to friction is calculated from Colebrook’s function and the roughness factor [Table 1 ASHRAE 2005 p 35.7]. Some typical values are given here.

Smooth	0.03 mm	0.0001 ft
Medium Smooth	0.09 mm	0.0003 ft
Average	0.15 mm	0.0005 ft
Medium Rough	0.90 mm	0.0030 ft
Rough	3.00 mm	0.0100 ft

Shape Size and Leakage: You must enter data to physically describe each duct airflow element. You input these values on the Shape Size and Leakage property page associated with each duct airflow element.

Powerlaw Model: Orifice Area

This airflow element allows you to describe the airflow through an orifice, and ContamW converts it to a powerlaw relationship.

Name: Enter a unique name you want to use to identify the duct flow element. The element will be saved within the current project and can be associated with multiple duct segments.

Description: Field for entering a more detailed description of the specific duct flow element.

Flow Exponent (n): Flow exponents vary from 0.5 for large openings where the flow is dominated by dynamic effects, and 1.0 for narrow openings dominated by viscous effects. Measurements usually indicate a flow exponent of 0.6 to 0.7 for typical infiltration openings.

Discharge Coefficient (C): The discharge coefficient is related to the dynamic effects and is typically close to 0.6 for an orifice and slightly higher for other openings in buildings.

Hydraulic Diameter: The hydraulic diameter is equal to $(4 \cdot \text{Area} / \text{Perimeter})$. For square openings this equals the square root of the area, and for long thin openings it is two times the width.

Reynolds Number: The transition from laminar flow to turbulent flow occurs over a very broad range of Reynolds numbers with the flow being fully laminar approximately below 100.

NOTE: The hydraulic diameter and Reynolds number have little impact on the calculations. Generally you should use the default values except for special circumstances where they need to

be modified. The values above describe the flow characteristics of an orifice in typical operation. At extremely low pressure drops the use of the powerlaw model leads to a division by zero during the network solution process. ContamX avoids this problem by changing to a linear model in this region. The model is based conceptually on the flow changing from turbulent to laminar at very low pressures. The Hydraulic diameter and Reynolds number are used to determine a point where the model changes from the powerlaw to linear.

Shape Size and Leakage: You must enter data to physically describe each duct airflow element. You input these values on the Shape Size and Leakage property page associated with each duct airflow element.

Powerlaw Model: $Q=C(\Delta P)^n$

This airflow element allows you to directly enter the coefficients C and n for the volumetric flow version of the powerlaw model.

Name: This is the name you give to this airflow element. This name must be unique within a project.

Description: Field for entering a more detailed description of the specific duct flow element.

Flow Coefficient (C): The coefficients may only be expressed in SI units due to the conversion method used. Use the following conversion to convert from IP units to SI units.

To convert from units of $\frac{cfm}{in H_2O^n}$ to the SI units of $\frac{m^3/s}{Pa^n}$ multiply by $\frac{1}{2119 \times 249^n}$

Flow Exponent (n): Flow exponents vary from 0.5 for large openings where the flow is dominated by dynamic effects, and 1.0 for narrow openings dominated by viscous effects. Measurements usually indicate a flow exponent of 0.6 to 0.7 for typical infiltration openings.

Shape Size and Leakage: You must enter data to physically describe each duct airflow element. You input these values on the Shape Size and Leakage property page associated with each duct airflow element.

Powerlaw Model: $F=C(\Delta P)^n$

This airflow element allows you to directly enter the coefficients C and n for the mass flow version of the powerlaw model.

Name: Enter a unique name you want to use to identify the duct flow element. The element will be saved within the current project and can be associated with multiple duct segments.

Description: Field for entering a more detailed description of the specific duct flow element.

Flow Coefficient (C): The coefficients may only be expressed in SI units due to the conversion method used. Use the following conversion to convert from IP units to SI units.

To convert from units of $\frac{lbm/s}{in H_2O^n}$ to the SI units of $\frac{kg/s}{Pa^n}$ multiply by $\frac{0.4536}{249^n}$

Flow Exponent (n): Flow exponents vary from 0.5 for large openings where the flow is dominated by dynamic effects, and 1.0 for narrow openings dominated by viscous effects. Measurements usually indicate a flow exponent of 0.6 to 0.7 for typical infiltration openings.

Shape Size and Leakage: You must enter data to physically describe each duct airflow element. You input these values on the Shape Size and Leakage property page associated with each duct airflow element.

Fan Model: Performance Curve

This airflow element allows you to create a fan based on a fan performance curve. You provide ContamW with a set of pressure rise and airflow rate data points, and ContamW performs a polynomial curve fit to the data point.

Name: Enter a unique name you want to use to identify the duct flow element. The element will be saved within the current project and can be associated with multiple duct segments.

Fan off conditions: These values specifies how the fan is modeled when the speed of the fan is insufficient to provide a flow.

Cut-off Ratio: The actual speed of the fan divided by the rated speed of the fan. When this ratio is below the specified cut-off limit the fan is modeled as a simple orifice based on the "equivalent orifice" defined below. During a simulation, ContamX obtains the actual ratio from the control value being applied to the duct segment with which this fan element is associated and compares it to the *cut-off ratio*. See Chapter 18 Fans in [ASHRAE 2004].

Equivalent Orifice: The size of the orifice used to model the fan opening when the control value drops below the specified limit.

Description: Field for entering a more detailed description of the specific airflow element.

Fan Performance Data: Click the "Edit Fan Curve Data" to input/modify the fan curve for this duct flow element. A detailed explanation of this data is given in the Fan Performance Data section under Working with Airflow Paths.

Shape Size and Leakage: You must enter data to physically describe each duct airflow element. You input these values on the Shape Size and Leakage property page associated with each duct airflow element.

Fan Model: Constant Mass Flow

This airflow element allows you to describe the airflow through a duct as a constant mass airflow rate.

Name: Enter a unique name you want to use to identify the duct flow element. The element will be saved within the current project and can be associated with multiple duct segments.

Description: Field for entering a more detailed description of the specific duct flow element.

Design (max) Flow: Enter the maximum mass flow rate. You can use the duct segment schedule to modify this value.

Shape Size and Leakage: You must enter data to physically describe each duct airflow element. You input these values on the Shape Size and Leakage property page associated with each duct airflow element.

Fan Model: Constant Volume Flow

This duct flow element allows you to describe the airflow through a duct as a constant volumetric flow rate. The design flow rate you specify is in terms of standard air. CONTAM

uses a value of 1.204 kg/m^3 based on conditions specified by ASHRAE for dry air: 101.325 kPa and 20 °C (ASHRAE 2000). Therefore, if actual conditions during simulation do not match these standard conditions, the results will differ from specified flow due to differences in air density.

Name: Enter a unique name you want to use to identify the duct flow element. The element will be saved within the current project and can be associated with multiple duct segments.

Description: Field for entering a more detailed description of the specific duct flow element.

Design (max) Flow: Enter the maximum volume flow rate. You can use the duct segment schedule to modify this value.

Shape Size and Leakage: You must enter data to physically describe each duct airflow element. You input these values on the Shape Size and Leakage property page associated with each duct airflow element.

Backdraft Damper: $Q=C(\Delta P)^n$

This is the volumetric flow version of the Backdraft Damper duct flow element.

Name: Enter a unique name you want to use to identify the duct flow element. The element will be saved within the current project and can be associated with multiple duct segments.

Description: Field for entering a more detailed description of the specific duct flow element.

Pressure Difference: Enter a flow coefficient and exponent for both the positive and negative flow directions of the duct segment. $\Delta P > 0$ refers to the pressure difference across the flow element that would lead to an airflow in the positive flow direction (See Airflow Path Properties) and $\Delta P < 0$ would lead to a flow in the opposite direction.

Flow Coefficients: The coefficients may only be expressed in SI units due to the conversion method used. Use the following conversion to convert from IP units to SI units.

$$\text{To convert from units of } \frac{\text{cfm}}{\text{in } H_2O^n} \text{ to the SI units of } \frac{\text{m}^3 / \text{s}}{\text{Pa}^n} \text{ multiply by } \frac{1}{2119 \times 249^n}$$

Flow Exponents: Flow exponents vary from 0.5 for large openings where the flow is dominated by dynamic effects, and 1.0 for narrow openings dominated by viscous effects. Measurements usually indicate a flow exponent of 0.6 to 0.7 for typical infiltration openings.

Shape Size and Leakage: You must enter data to physically describe each duct airflow element. You input these values on the Shape Size and Leakage property page associated with each duct airflow element.

Backdraft Damper: $F=C(\Delta P)^n$

This is the mass flow version of the Backdraft Damper duct flow element.

Name: Enter a unique name you want to use to identify the duct flow element. The element will be saved within the current project and can be associated with multiple duct segments.

Description: Field for entering a more detailed description of the specific duct flow element.

Pressure Difference: Enter a flow coefficient and exponent for both the positive and negative flow directions of the duct segment. $\Delta P > 0$ refers to the pressure difference across the flow

element that would lead to an airflow in the positive flow direction (See Airflow Path Properties) and $\Delta P < 0$ would lead to a flow in the opposite direction.

Flow Coefficients: The coefficients may only be expressed in SI units due to the conversion method used. Use the following conversion to convert from the IP units to SI units.

To convert from units of $\frac{lbm/s}{in H_2O^n}$ to the SI units of $\frac{kg/s}{Pa^n}$ multiply by $\frac{0.4536}{249^n}$

Flow Exponents: Flow exponents vary from 0.5 for large openings where the flow is dominated by dynamic effects, and 1.0 for narrow openings dominated by viscous effects. Measurements usually indicate a flow exponent of 0.6 to 0.7 for typical infiltration openings.

Shape Size and Leakage: You must enter data to physically describe each duct airflow element. You input these values on the Shape Size and Leakage property page associated with each duct airflow element.

3.9.5.2 Duct Junction and Terminal Properties

This section provides detailed descriptions of the specific duct junction and terminal properties. The following sections are the context-sensitive help topics that you can access by pressing **F1** when working with property pages of the "Duct Junction Properties" property sheet.

Duct Junction and Terminal – Junction Properties

Junction Number: This number is automatically generated by ContamW for identification purposes and is unique to each junction. ContamW will renumber junctions and terminals when saving a project if you have added or removed any junctions or terminals. These icons are numbered starting at the top level in the upper left hand corner of the SketchPad moving left to right and down the SketchPad then proceeding down through each level in the same manner.

Relative Elevation & Temperature: The elevation and temperature of the junction are used to determine how the duct flow responds to and influences the building stack effect. Enter the height of the midpoint of the junction or terminal point relative to the current building level. Enter the temperature you wish the junction or terminal to use during simulation.

Note that you can have ContamX calculate duct temperatures when performing simulations using the 1D Duct Model with the Short Time Step Method. See Contaminant Numerics Properties in the Working with Simulations section.

Junction: If you are creating a junction that connects two duct segments, then select the type of connection to create here. Items will be enabled/disabled depending on the valid connection types.

Define Downward Duct: You define vertical duct segments as downward running ducts accessed via the duct junction at the top of the downward segment. Press the "Define Downward Duct" button to define vertical segments. The SketchPad duct icons that are connected to the defined vertical duct will change to indicate the type of vertical connections associated with the junction, i.e., connected downward, upward or both (see Junction and Terminal Icons).

Terminal: If you are creating a terminal at the end of a duct segment, then select the type of connection to create here. Items will be enabled/disabled depending on the valid connection types.

Location: Enter the absolute X and Y coordinates and units for the selected junction. The Z coordinate will be taken as the *Relative Elevation* of the junction with respect to the building level on which the junction is located.

These values are required for *terminals* located in the Ambient zone when using a *Wind Pressure and Contaminant (WPC)* file (see Wind Pressure and Ambient Contaminant Files) or for *junctions* and *terminals* located within a *1D convection/diffusion zone* (see 1D Zone Data). These coordinates will be verified by ContamW against a WPC file or the axis of the 1D zone in which the junction or terminal is located. This verification will be performed automatically when you select either the **Run Simulation** or **Create a ContamX Input File** command from the **Simulation** menu.

Duct Junction and Terminal – Terminal Properties

Prior to CONTAM 2.4, duct terminals merely provided connectivity of the duct network to zones and did not have user-defined properties associated with them. Beginning with CONTAM 2.4, duct terminals have been revised to be more like duct flow segments, having geometric properties, loss coefficients and flow balancing capabilities.

Terminal flow is determined by the *equivalent loss coefficient* (C_e) of the terminal. C_e is determined during simulation by the combination of the Terminal Loss Coefficient (C_t), Balance Coefficient (C_b) and schedule value or control signal (S_c). As implied here, terminals can now be acted upon via the control network or scheduled to allow C_e to range between ($C_t + C_b$) for a control signal of 1.0 and a very large loss coefficient that essentially reduces the flow to zero for a control signal of 0.0. (See Ducts in the Airflow Analysis section of the Theoretical Background).

You can use the terminal properties to define the loss characteristics of a duct terminal to be used during simulation; to provide geometric data that will enable ContamW to provide you with more meaningful flow velocities at the terminal; and to define duct balancing properties that CONTAM can use to establish user-defined design airflow rates at the duct terminals. After setting the design conditions below, you can have CONTAM perform the duct balancing by selecting the Duct Balance airflow simulation method of the Run Control Properties.

Junction Number: This number is automatically generated by ContamW for identification purposes and is unique to each junction. ContamW will renumber junctions and terminals when saving a project if you have added or removed any junctions or terminals. These icons are numbered starting at the top level in the upper left hand corner of the SketchPad moving left to right and down the SketchPad then proceeding down through each level in the same manner.

Terminal Data: These properties are the characteristic data that describe the terminal itself regardless of whether it is balanced or not.

Terminal Loss Coefficient: Characteristic loss coefficient of the terminal. This defaults to the minimum value of 0.001 that effects a very small loss at the terminal. This provides consistency with terminal flow results for previous versions of CONTAM that did not implement loss coefficients at the terminals. See Chapter 35 of [ASHRAE 2005] and the

Duct Fitting Database [ASHRAE 2002] for data on dynamic losses in duct fittings.

Free Face Area: This is the total minimum area of the openings in the air outlet or inlet through which air can pass. This area will be used to calculate the velocity from the airflow rate at the terminal when displaying simulation results on the SketchPad.

Duct Area: The area of the duct to which the terminal is connected. Typically, this would be the same area as the duct segment just downstream/upstream from the supply/return air terminal. This value is used during simulation to determine the airflow rate through the terminal.

Balancing Data: These properties are used for the automated duct balancing capabilities implemented in CONTAM version 2.4. This feature will enable you to select a set of duct terminals for which you provide design airflow rates and have ContamX calculate loss coefficients that will enable the associated supply fan to provide the design flow rate.

Balance Terminal: Check this box if you would like ContamX to include this terminal in the duct balancing procedure.

Design Flow Rate: Provide the design airflow rate for which you want ContamX to determine a *Balance Loss Coefficient*.

Max Balance Coefficient: Provide a maximum value for the Balance Loss Coefficient below. This will be used to provide warning information if the *Balance Loss Coefficient* determined by the automated balancing procedure of ContamX exceeds this value.

Balance Loss Coefficient: This is the loss coefficient determined via the automated balancing procedure of ContamX. You can also modify it “manually.” As noted above, this value will be used to determine the effective loss coefficient (C_e) for the terminal.

Duct Junction and Terminal – Wind Pressure Properties

This set of properties is available only for those *terminals* that are located in the ambient zone and thus potentially subject to wind effects.

Wind Pressure Option: There are three wind pressure options for duct terminals located in an ambient zone:

1. No wind pressure
2. Constant pressure
3. Pressure dependent on wind speed and direction

When you select one of the above options, the following data entry options that correspond to the wind pressure option that you selected will be made available for your input.

Constant Pressure Data: Enter a constant wind pressure (wind speed and direction independent).

Variable Pressure Data: Variable wind pressures are computed from the product of the following three parameters:

1. The dynamic pressure of the wind at some reference height. ContamX will determine this pressure based on either the steady state wind data (See Wind Properties) or the transient wind data from a weather file (See Defining Transient Weather and Wind).

2. A coefficient accounting for local terrain effects (Wind Pressure Modifier).
3. A pressure coefficient accounting for relative wind direction (Wind Pressure Profile).

Wind Pressure Modifier: CONTAM uses this value to account for differences between wind velocity profiles at the building site and that at a measurement location, e.g. an airport weather station (See Working with Weather and Wind). CONTAM only requires this value if you are implementing Variable Wind Pressure for an airflow path. ContamW will provide a default value for this based on the data entered via the Weather and Wind Parameters Property Page or you can override the default value for a particular airflow path.

Wall Azimuth Angle: Enter the direction the wall faces (clockwise from north).

Profile: The wind pressure profile accounts for the wind direction effects. You may select a previously defined wind pressure profile from those contained in the Name list. Click the "New Profile" button to define a new wind pressure profile. To view or modify wind pressure profile data, click the "Edit Profile" button. After you press the "New Profile" or "Edit Profile" button, the Wind Pressure Profile page will be displayed with a graphical representation of the profile.

Duct Junction and Terminal – Filter and Schedule

Junctions and terminals differ with respect to Filters and Schedules. Junctions can not have filters associated with them and only the temperature of the junction can be scheduled or controlled. Beginning with CONTAM version 2.4, terminals can be both filtered and scheduled. The terminal temperature and loss coefficient can be scheduled (or controlled), although only one type of schedule can be applied at a time.

Filter: You may either create, edit or delete the filter associated with this terminal. Creating a new filter or modifying an existing one will display the Filter Properties dialog box. Use this dialog box to create and edit filter elements as needed. For an explanation of contaminant filters see Contaminant Filters in the Using CONTAMW section of the manual.

Schedule: As of CONTAM version 2.4, terminals consist of both a duct segment and a terminal junction during simulation. Each of these are schedulable items. You can select to schedule either the *Loss coefficient* of the segment or the *Temperature* of the junction. The list of schedules will contain those types (unitless or temperature) of schedules that exist in the project.

If there are previously defined schedules within the current project, you may select one from the Name list. Click the "New Schedule" button to define a new schedule. To view or modify an existing schedule click the "Edit Schedule" button. Setting this field to <none> indicates no schedule. (See Working with Schedules)

3.9.6 Balancing Duct Systems

In CONTAM you can balance duct systems yourself by providing balance points, e.g., orifice flow elements, at desired locations, or you can have CONTAM balance the system for you. Versions of CONTAM prior to version 2.4 required you to balance the system manually by adjusting the resistance of duct flow elements or local loss coefficients at desired balancing points. Beginning with version 2.4, CONTAM provides the ability to automatically balance duct systems using CONTAM duct terminals. See Duct Junction and Terminal - Terminal Properties and Working with Simulations.

□ **Automated Duct Balancing Procedure**

Automated duct balancing is another mode of simulation in ContamX. When you initiate a balance, ContamW will perform a pre-simulation check on the PRJ file and, if the check is successful, save the PRJ file and execute ContamX. ContamX will perform its own set of pre-balance checks to determine if a valid set of conditions has been defined. If this check is successful, then the automated balancing mode of ContamX will be initiated. If the pre-check of ContamX is unsuccessful, an error message will be displayed indicating the nature of the pre-check failure.

The automated procedure establishes relationships between fan elements and duct terminals by first performing a steady state airflow simulation for the entire building system as defined, i.e., duct balance coefficients having not yet been modified. If there are no errors in the system configuration, it then replaces fan elements with constant flow balancing fans, initializes fan flow rates to the summation of design flows of associated terminals (the minimum flow required for balancing), performs an airflow simulation, and checks for convergence of the duct balance by comparing calculated airflows with design airflows for all balance terminals. If convergence is not met, then the balance coefficients are adjusted (only increased) and the procedure is repeated until either convergence or the maximum number of iterations are performed. Upon completion of the balancing, ContamX will create a simulation results file (SIM file) that contains the calculated airflow results of the balanced system (see items 0 and 1 below).

The automated balance procedure generates a *duct balance result file* which has the same name as the project file but having the BAL extension. The contents of the balance report file will vary depending on the results of the automated balance procedure (See Results Files in the Working with Results section).

The basic steps to use the duct balancing feature are outlined below:

- Draw duct system – define segments, terminals and fans. Be sure to use separate fan flow elements for each fan, as the balance procedure may need to adjust the flow rates individually to achieve desired results. In order to obtain as realistic and reasonable results as possible, you should provide known loss coefficients at each terminal and junction throughout the system. Otherwise, the balancing procedure might not produce desired pressures within the system – although design flows will be achieved.
 - Designate balancing terminals and set design airflow rates for each.
 - Select airflow simulation balance procedure and run simulation.
 - Observe feedback from balance procedure. There are six possible results. If the one of the first two results is obtained (Success or Non-convergence), a simulation results file will be created that contains the airflow rates calculated based on the calculated balance coefficients. If one of the four error results is obtained, then the simulation results are for the pre-balance steady state calculation of the system as defined. The following list provides the six different outcomes of the balance and their associated completion code as it will appear in the first line of the *duct balance results file*.
0. Success: The balance procedure completed successfully. You will be prompted to copy balance coefficients to terminals and balance flow rates to constant flow fan elements when using such elements to model the fan. Once you have responded to the prompts, you can review the results of the balance on the SketchPad. Airflow rates from the terminals will be displayed via the SketchPad results display method. Balance

coefficients will not be saved until you save the project, so you can maintain the original pre-balance file by saving the post-balance file under a new name.

Performance curve fan elements are treated differently from constant flow fan elements, because they can not be directly modified to provide the balance flow rates. If using fan curve elements, consult the balance report file for setting the fan speed ratio to provide design flow rates. Use the value reported to provide a speed ratio via a control signal or schedule value to achieve the balance flow with the fan. If a value of 0.0 is reported, then the fan curve is not capable of providing the desired flow via a reasonable fan speed ratio and pressure rise. In this case it is likely that a different fan should be used.

1. Non-convergence: The balance procedure was not able to obtain the desired balance flow rates for all of the terminals within the preset number of iterations. You can choose to accept the calculated balance coefficients and run the balance procedure again with the modified coefficients as initial values. The balance report file will provide a relative flow coefficient equal to the calculated flow rate divided by the design flow rate.
2. Error – No terminals to balance: There are no terminals having the *Balance Terminal* option selected. Be sure to check the *Balance Terminal* check box to designate terminals to be balanced.
3. Error – Invalid flow direction(s): This error occurs when the balance terminals associated with a single fan have opposing design flow directions. Be sure the design flow direction of all terminals matches that of the associated fan: positive design flow rates are out of the terminal into the zone in which the terminal is located, and negative design flow rates are into the terminal from the zone. The fan flow direction is user-selectable via the duct segment icon of the associated fan.
4. Error – Failed to terminate link(s) at a fan: This error occurs when one or more balance terminals can not be traced back through the duct system (links) to a forced flow element (fan) within the duct system.
5. Error – Fan system includes supply and return terminal: This error occurs when a given fan has balance terminals both upstream and downstream of the fan. Be sure to only attempt to balance terminals on either the upstream or downstream side of a fan.

DUCT BALANCING NOTES:

- Duct balancing is performed with the entire building system considered. When the airflow calculation is performed during each iteration of the balancing procedure, the resistances between zones are included. Interzonal resistances that are relatively high can have a significant impact on the balancing results.
- During duct balancing, all junction temperatures are set to 20 °C. Ambient conditions are established by the steady state weather data with the exception that wind is set to 0 m/s.

3.10 Working with Controls

Controls allow the characteristics of one building element to be modified during transient simulations in response to a change in characteristics of another building element. You can also use controls to simply monitor building performance by polling building components for values and logging them to a control log file during a simulation.

Controls include sensors, links, modifiers and actuators. Sensors are control elements that are used to obtain a value, calculated during simulation, of a building component and then provide it to a control network. You can use sensors to obtain contaminant concentrations and temperatures of zones, duct junctions and duct terminals and to obtain pressure differences and airflow rates of airflow paths, duct segments and duct terminals. You can use actuators to control schedulable items including airflow paths, duct segments, junctions and terminals, simple air-handling systems, and contaminant source/sinks. Note that actuator input to a controlled building component overrides schedules associated with the component.

The connection between sensors and other nodes in a control network are made up of control links. Control modifiers are nodes within the control network that are used to change or modify their input signals. Control actuators are used to modify various characteristics of building components based on control signals obtained from sensors and even modified by signal modifiers. For example, a sensor can be used to obtain a contaminant concentration within a zone, and a proportional control actuator can be used to adjust supply airflow through a supply airflow path into the zone based on the sensed concentration.

3.10.1 Control Elements

A control network can consist of many different control nodes, each of which is defined by a control element. There are several different types of control elements provided, such as a report element, that can be used to convert dimensionless sensor values to engineering units; mathematical elements to perform simple mathematical and logical operations on input signals; and classical control elements such as proportional/integral, band and limit controls. There is a phantom control element that can be used to reference an existing control node, which is convenient for linking nodes that may be on different levels of a building or simply too far apart to conveniently draw a physical link. Each of these element types is described in detail in the Control Element Types section. While every node in the control network must refer to a single *control element*, multiple nodes can refer to the same *control element* using the phantom node/control element.

Collections of control elements can be grouped into Control Super Elements, which can be stored in CONTAM Library files and shared within and between projects. *Super elements* can reduce the amount of drawing of repetitive control logic and enable the creation of custom reusable sensors and control algorithms.

3.10.2 Drawing Controls

You draw controls using the links drawing tools previously described in the "Working with the SketchPad" section. When you finalize the drawing of a set of control links, the appropriate icons will be placed upon the SketchPad depending on where the link originates and where it terminates. For example, if you begin drawing a control link directly on a zone icon, a sensor icon will be placed adjacent to the zone icon, and if you terminate a control link on an airflow path, an actuator icon will be placed adjacent to the airflow path icon (See Figure below).

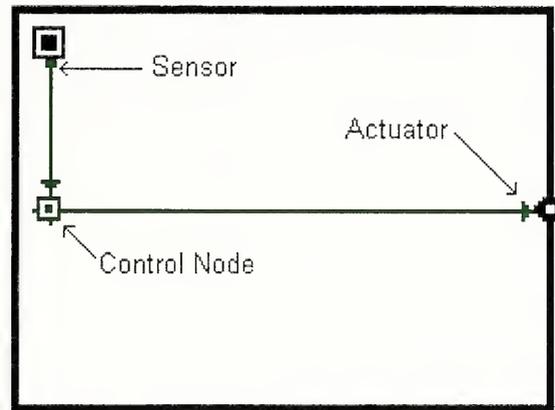


Figure – Control link icons

If you terminate a control link on a blank cell, a control node icon will be placed within that cell. You may not originate and terminate a link on a non-control node icon (e.g. zone, flow path or occupant) without an intermediate control node icon (See Figure below). This is due to the requirement that the input from an actuator to a non-control node icon must be in the form of a multiplier ranging from 0.0 to 1.0.

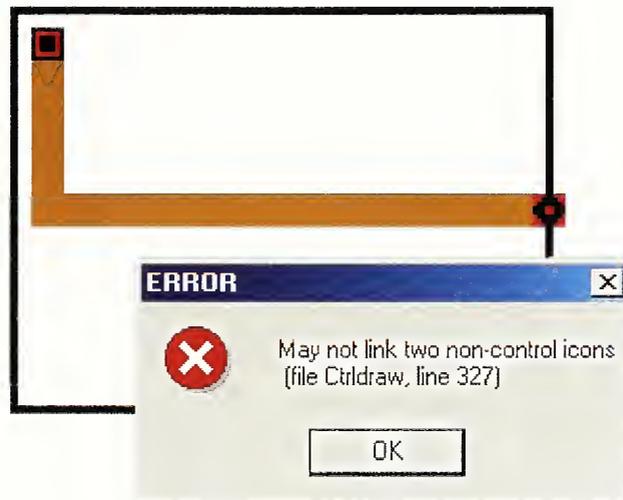


Figure – Invalid control link

Control node icons can have up to four links into or out of them depending on the type of control node you are defining. If a control node icon is not yet defined, ContamW will only allow you to draw at most two inputs (one vertical and one horizontal) into the node and at most four outlets from the node. However, if the control node icon is already defined, ContamW will only allow the correct set of links to be defined into the icon depending on the *control element type* (See Control Element Types) of the control node. This is due to the input signal convention of control nodes that distinguishes two different inputs signals for certain element types, e.g. limit controls, or to establish the order of operators for non-commutative mathematical operations, e.g. subtraction and division. As shown in the Figure below, *Input Signal 1* enters either from the left or right and *Input Signal 2* enters from either the top or bottom of the node. Details on input signals are provided for each element type where required in the description of the specific elements.

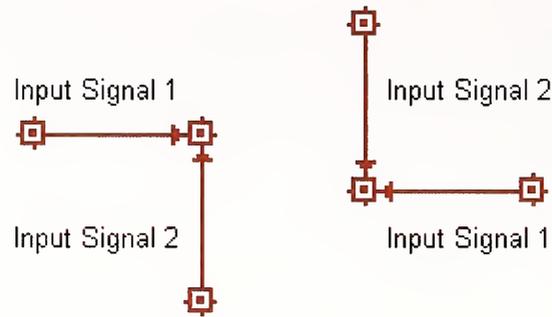


Figure – Control node input signal convention

3.10.3 Deleting Controls

Deleting currently defined controls is a two-step process and can only be performed when CONTAMW is in the *normal* mode, i.e., not viewing SketchPad results. That is, if you want to delete a control link, sensor, actuator or node, you must first *undefine* the item. To undefine a control you must highlight a control node or link and press the **Delete** key or select Delete from the Edit menu. This will highlight either a single node (if no output signals) or a link segment and prompt you to confirm the undefine operation by displaying a message box. Once you undefine a control node or link segment, it will again be displayed in red. To delete the control node or link you highlight any portion of the undefined node or link segment and perform the deletion. The entire portion to be deleted will be highlighted, and you will be asked to confirm the deletion. If you try to delete a node that is referenced by a phantom control element, you will be warned. Deleting such elements prior to deleting those phantom nodes that reference them can cause problems.

3.10.4 Hiding Controls

If there are controls in your project, you can hide them on the SketchPad. Use the **View → Control Links** menu command to toggle the display of the control network on and off.

3.10.5 Modifying Controls

To modify the parameters of a defined control node, you use the icon definition procedure (See Defining Building Component Icons) to display its properties and make the desired changes. You can not move and copy controls. To do this you must delete and redraw your control network, therefore careful planning is advised in laying out your control networks.

3.10.6 Defining Controls

You must define all *control nodes* and *sensor* icons in order to perform a simulation. Control links will be defined automatically upon definition of the node or sensor from which they emanate. Generally, you should define control nodes in the direction of information flow indicated by the link arrows. Define *control nodes* and *sensors* using the icon definition procedure (See Defining Building Component Icons) to display and edit their properties.

□ Sensors

When you double-click an undefined sensor icon, the Control Sensor definition dialog box will appear. The parameters will either be those for a *zone sensor* or *flow path sensor* depending on the building component with which the sensor is associated.

As noted in the Drawing Controls section above, sensor icons can be placed automatically when you begin drawing a control link on a sensible icon, e.g., zone or flow path. With CONTAM version 2.4, you can now define zone sensors within any blank cell inside of a zone. This increases the previous limit of four zone sensors per zone and is done by selecting the "Sensor" control element type when defining a control node as presented in the following sub-section.

□ **Control Nodes**

Double-clicking on an undefined control node icon will display the Control Element Types dialog box from which you select the type of control element you wish to define. Control elements can be categorized by the number of inputs that can be associated with them: unary, binary and multiple. Therefore, when the Control Element Types dialog is displayed, various types will be enabled/disabled for selection depending on the number of inputs that you have drawn into the currently selected node.

Once you select the type of element you want to create, you assign values to the parameters of the control elements. All control elements have an optional *Name* parameter with the exception of the phantom for which the *Name* parameter will be used to reference the name of another existing element. The name parameter is only required if you want to reference the element with a phantom node.

Each element also has an optional *Description* field, which is filled in with a default value when you create an element, but you may change the description. The description will be displayed in the status bar when you highlight the control node on the SketchPad. Many of the elements do not have any parameters other than the *Name* and *Description*, because they simply perform a predetermined operation on the input(s) to the node, e.g., the "+ Add" element simply takes two inputs, adds them and provides the results as an output signal. Other control elements have additional parameters required to define their behavior, e.g., the "Constant" element has a value parameter use to specify the constant value of the output signal from the control node.

3.10.6.1 Control Element Types

There are a number of control element types that you can associate with an undefined control node. Detailed descriptions and properties of these control elements are provided in the following sections. There are also New Control Elements for Version 2.1.

□ **Sensor**

There are two types of sensors that can be created depending on the type of icon with which the sensor is associated: *Zone or Junction Sensor* and *Path or Duct Sensor*. The Sensor definition dialog box will provide you with the properties available for the type of sensor being created.

Zone or Junction Sensor

Sensed Value: Select the property of the zone or junction that you would like the sensor to provide as output: *Temperature* or *Contaminant mass fraction*.

Path or Duct Sensor

Sensed Value: Select the property of the path or duct that you would like the sensor to provide as output: *Flow rate* through the path or *Pressure drop* across the path.

Sensor Output: Sensor output defaults to the base units of CONTAM, but you can modify this by changing the *Gain* and *Offset* values provided. However, the Report a Value control

element (described below) automatically provides values to convert from the base units to the default units of your project or of the particular contaminant which you are sensing.

$$\text{Output} = (\text{Sensed Value} - \text{Offset}) / \text{Gain}$$

Gain: a proportional value used to convert the *Sensed Value* to the *Output* value

Offset: an offset value used to convert the *Sensed Value* to the *Output* value

Location: These parameters only apply to *Mass fraction sensors* associated with 1D zones. Select whether you want the sensor to provide the **Zone Average** concentration of a 1D zone or the value within a single **1D Cell** of a 1D zone. The 1D cell option is only effective when performing a simulation using the *short time step* method otherwise the sensor will provide a value based on the zone average.

X, Y and Rel Elevation (Z) are the coordinates and units for the selected sensor. X and Y are absolute coordinates and Rel Elevation is relative to the level on which the sensor is located. These values are required for any sensor that is located within a cell of a *1D convection/diffusion zone* (see 1D Zone Data). If the sensor is to be located within a particular cell of a 1D zone, then these coordinates will be verified by ContamW against the axis of the 1D zone in which it is located. This verification will be performed automatically when you select either the **Run Simulation** or **Create a ContamX Input File** command from the **Simulation** menu.

Description: Use this to describe the sensor in detail. This description will be displayed in the status bar when you highlight the sensor icon. The description defaults to either "zone sensor" or "path sensor."

Phantom control

Use this control element to reference a named control element that already exists elsewhere within your project. Phantom nodes are useful when you need a control signal from a node that is located remotely upon the SketchPad or when you are implementing one of the multiple-input elements. This is useful in referencing a node on another level of a building, because CONTAMW does not allow you to draw links between building levels. This can also be used to "cascade" inputs into the multiple-input *sum* and *average* controls when you want to sum or average more than three inputs or remotely located, named inputs.

Schedule

Use this control element to apply a schedule via the control network. A schedule can be applied to airflow paths, simple air handling systems, inlets and outlets of simple air handling systems and source/sinks. Control schedules will override schedules that are defined as a parameter of an element. You can select an existing schedule or create a new one (See Creating Schedules).

Constant

Use this control element to define a simple constant. The constant is useful as input to other control elements such as the *mathematical*, *limit* and *switch* controls.

Modifier

Use this control element to modify an input signal by providing a Gain and Offset parameter. The input signal will be modified according to the following equation.

$$\text{output} = (\text{input} - \text{Offset}) / \text{Gain}$$

The parameters for the Modifier control are:

Gain – a proportional value used to convert the *input* signal

Offset – an offset value used to convert the *input* signal

□ **Hysteresis**

Use this control element to simulate hysteresis of a control actuator. Hysteresis is characterized by the Slack parameter where Slack is the fraction of the input signal’s range over which the output signal remains constant when the input signal changes direction. The relationship of the output signal to the input signal is shown in the following figure. This plot shows the signal increasing from 0 to 1 and back down to 0 again. The increasing and decreasing signals are characterized by the following equations respectively.

Increasing: $output = \text{maximum}(0.0, input * Slope + Int)$

Decreasing: $output = \text{minimum}(1.0, input * Slope)$

Where

$$Slope = 1 / (1.0 - Slack)$$

$$Int = output - Slope * input$$

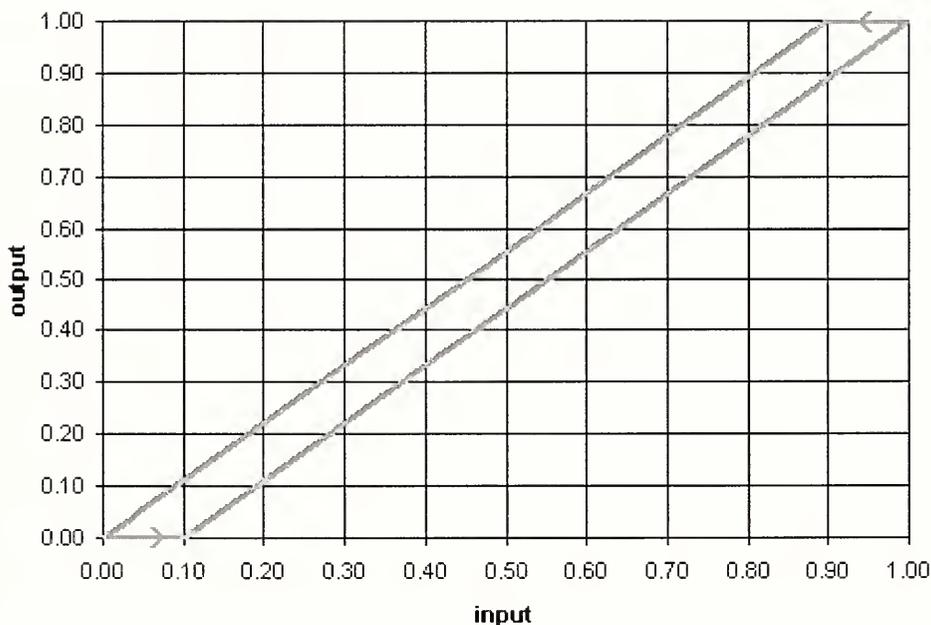


Figure – Plot showing hysteresis curve of Hysteresis control element

□ **Report a value**

Use this control element to report values to a file created by CONTAMW for the specific purpose of logging these reported values for each *listing time step* of a transient simulation (See Output Properties). If you perform a transient simulation and implement *report control elements*, CONTAMW will create a file in the same directory as your project file with the name of your project file and the .LOG extension appended. For example if your project is *MyProj.PRJ*, CONTAMW will create the file *MyProj.LOG*. You can also use the Report control element to provide automatic unit conversions of sensor data from the dimensionless

units of a sensor (e.g. K or kg/kg) to engineering units associated with the type of sensed data (e.g. °C or ppm(v)).

The value output from the report element will be determined according to the following equation:

$$output = (input - Offset) * Scale$$

The parameters for the Report control are:

Scale – a scale value used to convert the *input* signal

Offset – an offset value used to convert the *input* signal

Header – the header that will appear in the log file for this control element

Units – units that will appear below the header in the log file for this control element

The report control will also provide the default conversion values, Scale, Offset and Units if CONTAMW finds a sensor "down-link" in the control network that has not yet been converted from its dimensionless form.

❑ **Signal split**

Use this to split a signal, so the same signal can be directly linked to multiple nodes in the control network.

❑ **Logical**

Use these controls to provide logical operations including:

Binary – convert input signal to binary 0 or 1,

NOT – negate single input value and provide binary output 0 or 1,

AND – perform AND operation on two input values and provide binary output 0 or 1.

$$output = 1 \text{ IF } (input_1 > 0) \text{ AND } (input_2 > 0)$$

$$output = 0 \text{ otherwise}$$

OR – perform OR operation on two input values and provide binary output 0 or 1.

$$output = 1 \text{ IF } (input_1 > 0) \text{ OR } (input_2 > 0)$$

$$output = 0 \text{ otherwise}$$

XOR – perform XOR operation on two input values and provide binary output 0 or 1.

$$output = 1 \text{ IF } (input_1 > 0) \text{ AND } (input_2 \leq 0)$$

$$output = 1 \text{ IF } (input_2 > 0) \text{ AND } (input_1 \leq 0)$$

$$output = 0 \text{ otherwise}$$

❑ **Mathematical**

Use these controls to provide mathematical operations including:

Absolute value – provide absolute value of input signal,

Add, Subtract, Multiply, and Divide – perform operation on two input signals and provide result as output.

Note the order of operation is important for Subtract and Divide.

Subtract: $output = input_1 - input_2$

Divide: $output = input_1 / input_2$

Sum and Average – Sum or average multiple input signals and provide result as output signal. Three signals can be directly input to using links drawn into the control node icon, and more can be cascaded through the use of *phantom* nodes.

□ Limit switches

Limit switches provide the ability to set an output signal to either on or off (0 or 1) based upon the difference between the two input signals.

Lower limit switch

$output = 1$ IF ($input_1 < Limit$) [Limit = $input_2$]

$output = 0$ otherwise.

Upper limit switch

$output = 1$ IF ($input_1 > Limit$) [Limit = $input_2$]

$output = 0$ otherwise

The band switches incorporate the control parameter, **Band width**, to provide a *dead band* for the on/off control.

Lower band switch

$output = 1$ when $input_1$ rises above $input_2 + Band\ width$

$output = 0$ when $input_1 < input_2$

$input_2$ is the lower limit of the band

Upper band switch

$output = 1$ when $input_1$ falls below $input_2 - Band\ width$

$output = 0$ when $input_1 > input_2$

$input_2$ is the upper limit of the band

□ Limit controls

Limit controls provide the ability to set an output signal to a value ranging from 0.0 to 1.0 based upon the difference between the two input signals. These would typically be used to provide an *error signal* to a proportional controller.

Lower limit control

$output = input_2 - input_1$

Upper limit control

$output = input_1 - input_2$

The *upper and lower limit controls* can be used to prepare an error signal for the *proportional controls*. If the input signal to an *upper limit control* is greater than the limit value, a positive output results; e.g., if a contaminant concentration is too high, the positive signal could

activate an airflow device. If the input signal to a *lower limit control* is lower than the limit value, a positive output results; e.g., if the air flow through an opening is too low, a positive signal could activate a fan to provide the required flow.

❑ **Proportional and Proportional-Integral**

Proportional control – This is a model of a simple proportional controller where the input signal is an error signal that is typically a sensed value minus a set-point value which is obtained by using other control functions, e.g. the *constant control*. The input signal is multiplied by the proportionality constant parameter of the control, **Kp**. The output signal is limited to values between 0 and 1, so you should set the proportionality constant accordingly. The output signal could then be modified as required using other control elements.

$$\text{output} = \text{input} \cdot K_p$$

Proportional-Integral control – This is a model of a simple P-I controller where the input signal is an error signal. The output signal is limited to values between 0 and 1 and is modified by the control parameters **Kp** and **Ki** according to the following equation.

$$\text{output} = \text{output}^* + K_p \cdot (\text{input} - \text{input}^*) + K_i \cdot (\text{input} + \text{input}^*)$$

where *output** and *input** are the values of *output* and *input* at the previous time step. The Kp and Ki factors must be tuned for the specific problem and time step. This is similar to an "incremental" P-I algorithm described in equations (17-37) and (18-3) of [Stoecker and Stoecker 1989].

❑ **Scheduled Delay**

This control element provides the ability to simulate time delays associated with the ramping up/down of system components changing between states, e.g., the spin down of a fan or the opening/closing of a damper. The *scheduled delay* element allows you to define a schedule according to which the change of state occurs. The *output* will change according to this schedule when the *input* changes.

Node Name: This is an optional name you can provide for this control node. This name can be used to reference this node with a *Phantom control node*.

Schedule - Signal Increasing: Select/enter a day schedule that characterizes the delay in an increasing signal. The schedule must be *trapezoidal* beginning with a value of 0.0 at time 00:00:00 and increase to a value of 1.0 before 24:00:00. Only one increasing time period per schedule will be allowed.

Schedule - Signal Decreasing: Select/enter a day schedule that characterizes the delay in a decreasing signal. The schedule must be *trapezoidal* beginning with a value of 1.0 at time 00:00:00 and decrease to a value of 0.0 before 24:00:00. Only one decreasing time period per schedule will be allowed.

Description: Enter an optional description for this control element.

❑ **Exponential Delay**

This control element provides the ability to simulate time delays associated with the ramping up/down of system components changing between states, e.g., the spin down of a fan or the opening/closing of a damper. The *exponential delay* element allows you to define an exponential delay based on a time constant.

Node Name: This is an optional name you can provide for this control node. This name can be used to reference this node with a *Phantom control node*.

Time Constants:

Increase: Enter the amount of time it should take for the *output* signal to exponentially increase by $(1 - 1/e)$ % of the total change in the *input* signal when it rises from one state to the next. The format is hh:mm:ss.

Decrease: Enter the amount of time it should take for the *output* signal to exponentially decrease by $(1 - 1/e)$ % of the total change in the *input* signal when it falls from one state to the next. The format is hh:mm:ss.

NOTE: An amount of time of about 4.6 times the value entered above is required for the *output* signal to increase/decrease by 99 % of the total change in the *input* signal.

Description: Enter an optional description for this control element.

Maximum and Minimum

The *output* will be the *maximum* or *minimum* of all *input* signals to the control node each time step. Once a node is defined to be of this type, up to three input signals can be drawn directly into it. More signals can be cascaded through the use of *phantom* control elements.

Node Name: This is an optional name you can provide for this control node. This name can be used to reference this node with a *Phantom control node*.

Description: Enter an optional description for this control element.

Integrate over time

The *output* will be the integration over time of *input signal 1* (horizontal) controlled by *input signal 2* (vertical). Integration occurs when *input signal 2* > 0; there is no integration when *input signal 2* = 0; the integral is reinitialized to zero when *input signal 2* < 0. A simple trapezoidal integration method is used.

Running average

The *output* will be the average of the *input* signal integrated over the time span, Δt_{int} , set by

the user:
$$out = \int_{\Delta t_{int}} in dt / \Delta t_{int}$$
. At the start of simulation, when the simulated time is less than the time span, the output will be the average of the input up to that time.

Node Name: This is an optional name you can provide for this control node. This name can be used to reference this node with a *Phantom control node*.

Time Span: Enter the amount of time included in the running average, Δt_{int} . The format is hh:mm:ss.

Description: Enter an optional description for this control element.

Continuous Value File (CVF)

This control element allows you to implement general schedules by obtaining input values from a file. This allows you to create schedules that are not restricted by the 12-day schedule limit of CONTAM's *week schedules*. The file is referred to as a *continuous value file*. This file is an ASCII file that you create according to the format specified in the Continuous Values File (CVF) Format section in Part 2 of this document. You can only use one CVF file

per simulation, and the file may contain multiple lists of values. Value lists are referenced by *Value name* which are column headings in the file.

Value Name: Select the name of a set of values from the list of headings as they appear in the CVF file.

File Name: This field simply displays the CVF file that contains the data that will be used by the control node during simulation. You must use the **Data→Controls→Continuous Values File...** menu item to select the file you want to use prior to creating CVF control nodes.

Node Name: This is an optional name you can provide for this control node. This name can be used to reference this node with a *Phantom control node*.

□ **Discrete Value File (DVF)**

This control element allows you to implement scheduled discrete events by obtaining event times and values from a file. This allows you to create schedules that do not repeat according to CONTAM's *week schedules*. The file is referred to as a *discrete value file*. This file is an ASCII file that you create according to the format specified in the Discrete Values File (DVF) Format section. You can only use one DVF file per simulation, and the file may contain multiple lists of events. Event lists are referenced by *Value name* which are column headings in the file.

Value Name: Select the name of a value from the list of headings as they appear in the .DVF file.

File Name: This field simply displays the DVF file that contains the data you need. You must use the **Data→Controls→Discrete Values File...** menu item to select the file you want to use prior to creating a DVF control node.

Node Name: This is an optional name you can provide for this control node. This name can be used to reference this node with a *Phantom control node*.

3.10.6.2 Continuous Values File (.CVF) Format

Control node values that change linearly in time are made available to the ContamX 2.1 simulation program through the continuous values file which has a .CVF file name extension.

The first line of the CVF is used to identify the type of file. It is exactly:

```
ContinuousValuesFile ContamW 2.1
```

The second line is a description of the file entered by the user:

```
desc[] // file description (I1) {W}; may be blank
```

The next line defines the period covered by the file:

```
StartDate // first date for DEF data (mm/dd → IX)
```

```
EndDate // last date for DEF data (mm/dd → IX)
```

The next section specifies the control node names:

```
_nbvfn // number of CVF node names
```

followed by _nbvfn lines consisting of node names:

```
name[] // node name (I1)
```

The remainder of the file consists of data for all nodes for each date and time from StartDate to EndDate:

```
date // date (mm/dd → IX)
```

```
time          // time of day (hh:mm:ss → I4)
value[1]      // value [?] for name[1] (R4)
...
value[_nbvf] // value [?] for name[_nbvf] (R4)
```

The node data must start at time 00:00:00 on the StartDate and end at 24:00:00 on the EndDate. The times must be in consecutive order, but the difference between successive times need not be constant.

The file description may not begin with a '!'. The StartDate and EndDate are used to verify that the file data covers the entire period to be simulated. The StartDate may not be later than the EndDate. Data elements on a single line are separated by tabs. The data must be in time-sequential order. The file values must be in the units needed for the signal created by the control node.

Node names may not include imbedded blanks. Data for nodes that are not in the project file will be ignored. If a CVF node name in the project file is not included in the CVF, a fatal error will result. ContamW will assist the user by checking node names. ContamX will perform the name check before simulation begins.

3.10.6.3 Discrete Values File (.DVF) Format

Control nodes values that change discretely in time are made available to the ContamX 2.1 simulation program through the discrete values file which has a .DVF file name extension.

The first line of the DVF is used to identify the type of file. It is exactly:

```
DiscreteValuesFile ContamW 2.1
```

The second line is a description of the file entered by the user:

```
desc[]          // file description (I1) {W}; may be blank
```

The next line defines the period covered by the file:

```
StartDate       // first date for DEF data (mm/dd → IX)
```

```
EndDate         // last date for DEF data (mm/dd → IX)
```

The next section specifies the control node names:

```
_ndefn         // number of DVF node names
```

followed by _ndefn lines consisting of:

```
name[]         // node name (I1)
```

```
value         // initial value [?] (R4)
```

Succeeding lines present data whenever a node value changes:

```
date           // date (mm/dd → IX)
```

```
time          // time of day (hh:mm:ss → I4)
```

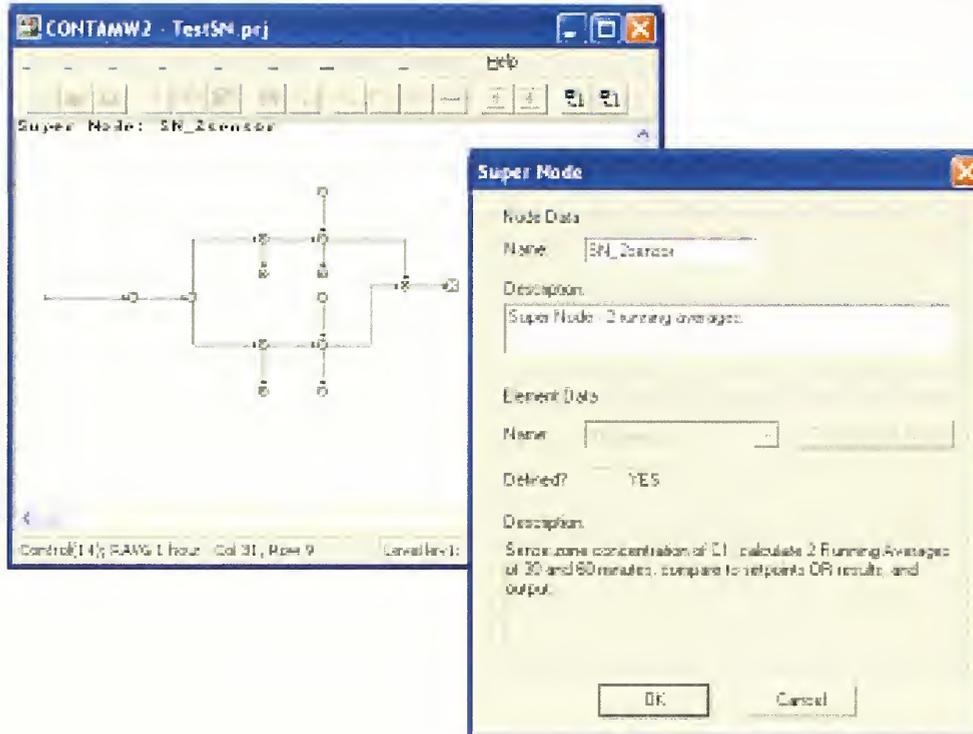
```
index         // index into name[]
```

```
value         // new value [?] (R4)
```

The file description may not begin with a '!'. The StartDate and EndDate are used to verify that the file data covers the entire period to be simulated. The StartDate may not be later than the EndDate. Data elements on a single line are separated by tabs. The data must be in time-sequential order. More than one node may change at the same time. The file values must be in the units needed for the signal created by the control node.

Node names may not include imbedded blanks. Data for nodes that are not in the project file will be ignored. If a DVF node name in the project file is not included in the DVF, a fatal error will result. ContamW will assist the user by checking node names. ContamX will perform the name check before simulation begins.

Super Node SketchPad is activated by *instantiating* an existing *control super element* or double-clicking on a Super Node icon. When the Super Node SketchPad is active, the upper left corner of the SketchPad will display "Super Node:" followed by the name of the super node currently displayed on the SketchPad. When working with super nodes the "Super Node" dialog box will also be displayed as shown below.



❑ **Defining Super Elements**

To create a super element you activate the Super Element SketchPad. Unlike other library element types, all super elements must be created and edited within a CONTAM project file. Initiating the super element editing mode activates both the Super Element SketchPad and the Control Super Elements dialog box shown below. Use the SketchPad to draw the *super element diagram* and define control sub-node types in much the same way as is done for controls on the main SketchPad. The dialog box is provided to name, save and set other super element properties described below.

❑ **Deleting Super Elements**

You can delete super elements via the CONTAMW Library Manager. Only those super elements that are currently not referenced by a Super Node can be deleted. You can determine whether or not a super element is referenced by the *super node reference count* provided on the Control Super Element dialog box when displaying a particular super element on the Super Element SketchPad.

❑ **Modifying Super Elements**

Modifying Super Elements consists of modifying the drawing (i.e., adding and deleting control links and sub-nodes), modifying the properties of individual sub-nodes and modifying the Super Element Properties.

Modifying the drawing – You can only modify the drawing if the *super node reference count* is 0. This limitation is in effect, because all instances of a *super element* (i.e., *super*

nodes) reference the sketch data and list of sub-nodes of the super element.

Modifying sub-node properties – You can modify the properties of sub-nodes as you desire. This will not affect existing instances of the super element; however, this will change the default properties of new instances of the super element.

Modifying Super Element properties – You can modify the super element properties as you so desire. This will not affect existing instances as super nodes simply reference the super element properties, but do not create new instances of them.

3.10.7.1 Control Super Element Properties

This section provides information on utilizing the Control Super Element dialog box that together with the Super Element SketchPad enables you to manage and define control super elements and is activated via the **Data→Controls→Super Elements...** menu item. There are two section of information presented: Element Manager and Element Properties editor.

□ Element Manager

This is used to select super elements for editing and creating new elements and activating the CONTAMW Library Manager to work with Super Element Library files.

Name: This combo box provides you with the name of the currently active super element for which the sketch data is also displayed on the Super Element SketchPad. The name should also appear on the upper left corner of the sketchpad. You can also select other existing super elements from the drop down list. This will change the super element displayed upon the sketchpad as well as the properties displayed on this dialog box.

Check Element: While working with a super element, you can perform a check on it to determine if is a fully defined super element. If the check passes then the super element is considered to be fully defined. If the check fails, then ContamW will provide feedback in the form of messages and highlighting of errors on the sketch. Even though a super element is undefined, it can still be saved in its incomplete form.

There are a few basic reasons why the check could fail. Super elements are very similar to control networks on the main SketchPad, so undefined super elements can result from problems similar to main SketchPad control network as well as problems specific to super elements.

Potential reasons for undefined Super Elements:

- undefined sub-nodes (i.e., undefined sketch icons)
- incorrect number of inputs to control nodes
- circular control logic
- no Output Node set
- pre-defined input to Input Node

New Element: This will create new super element having a default name of "SE_##" where ## will depend on the number of existing super elements. New super elements have no sketch data, so the SketchPad will be cleared. Once you create a super element, you do not have to save it. Any modification you make to the sketch and properties will be maintained automatically with the exception of the name which you must set via the "Rename Element" feature.

Library: Use this feature to activate the CONTAMW Library Manager to create control library files, copy super elements to and from super element library files (type LB5) and delete unused super elements from the current project file. (See Working with Data and Libraries)

Element defined? This feature indicates whether or not a super element is fully defined (as determined by ContamW) meaning it can be instantiated to create super nodes. Element definition is verified by "checking." Checking is performed using the "Check Element" feature described above, when you switch the currently displayed super element from the list of names, or when you close the Control Super Elements dialog box, i.e., exit the super element editing mode.

Super Node Reference Count: This is the number of instantiated super nodes that reference, the super element. The super element can not be deleted from the project file unless this count is equal to zero.

□ **Edit Element Properties**

These are the user-defined properties of the super element other than the sketch and control nodes. These properties include the Name, Description, Input Node name and Output Node name.

Rename Element: Use this feature to enter a new name for the super element and change it with the accompanying button. The new name will be compared to existing names before being reset.

Description: Provide a detailed description of the super element of up to 255 characters. This description will be displayed in the CONTAMW Library Manager. The description and the super element name are currently the only means by which to identify the super elements when viewing them in the library manager.

Input Node: The *input node* of the super element is the sub-node which will receive an input signal if one is drawn to an instantiated supernode on the main SketchPad. This is not required to be set. If needed, select a sub-node to serve as the main input for the super element from the list of *named sub-nodes*. Note that you must have provided a name for the sub-node which you wish to make the *input node*, and the sub-node must be of a type that accepts at least one input.

Multiple inputs

If you require a super element with multiple inputs, you can define a sub-node to be of type *phantom*. Once you instantiate a super node from the super element, you can select a named control element that you want the *phantom sub-node* to reference.

Output Node: The *output node* of the super element is the sub-node from which the signal will be provided to an output drawn from an instantiated *super node* on the main SketchPad. This is required to be set for a super element to be *defined*. Note that when instantiated, the output node will assume the name of the super node for the purposes of referencing with phantom nodes on the main SketchPad.

3.10.7.2 Control Super Nodes

Super nodes are specific instances of *super elements* having the same control logic and set of sub-nodes as the super element. When you instantiate a *super node* from a super element, it takes on the default values of the super element's sub-nodes; however, you can modify the values of the sub-node parameters as needed.

□ Creating Super Nodes

To create a super node, you select the *Super element...* from the Control Element Types dialog box just as you would define any other control node on the main SketchPad (See Defining Control Nodes). This will activate the Super Node SketchPad mode and the accompanying Super Node dialog box. You then work with both the dialog box and sketchpad to select a super element to instantiate and to modify the sub-nodes as you require for the particular instance (See Control Super Node Properties).

Once you have created the *super node* and exited the super node editing mode, the super node will appear as a single icon (*super node icon*) on the main SketchPad.

□ Deleting Super Nodes

You delete super nodes from the main SketchPad by simply highlighting the super node icon and pressing the delete key on the keyboard or selecting **Edit→Delete** from the main menu. If any of the sub-nodes of the deleted *super node* are referenced by phantom nodes, you will be informed and prompted as to whether you want to undefine the phantom nodes (and sub-nodes) that reference them.

□ Modifying Super Nodes

Once you have instantiated a *super node*, you can modify the sub-nodes of a *super node*. Activate the super node editing mode by double-clicking the super node icon on the main SketchPad. Simply double-click on the sub-nodes whose properties you wish to modify and edit them as you would nodes on the main SketchPad. Note that you can not modify the sketch in any way, i.e., you can not draw or delete sub-nodes.

3.10.7.3 Control Super Node Properties

This section provides information on utilizing the Super Node dialog box that together with the Super Node SketchPad enables you to manage and define control super nodes. This dialog box is activated when creating a new *super node* or when modifying an existing one. There are two sections of information presented: Node Data and Element Data.

□ Node Data

This displays *super node* specific data used to identify a specific super node.

Name: As with all control node types, you can provide a name for the *super node*. Named super nodes can be referenced by *phantom nodes*. When you name a super node, its output sub-node will also be given the same name, because the output sub-node will be the node actually referenced by any *phantom nodes* that refer to the *super node*.

Description: Enter a detailed description of the super node to help you identify the super node. This description will appear in the status bar when you highlight the super node icon on the SketchPad.

□ **Element Data**

This section enables you to select and set the super element from which to instantiate a new super node. Once instantiated this section simply provides information on the super element from which the super node was instantiated.

Name: Prior to instantiation of a super node, this combo box provides you with the name of the currently active super element for which the sketch data is also displayed on the Super Element SketchPad. The name should also appear in the upper left corner of the sketchpad. You can also select other existing super elements from the drop down list. This will change the super element displayed upon the sketchpad as well as the properties displayed on this dialog box.

Create Super Node: Click this button to instantiate a *super node* from the currently selected *super element*. Once instantiated, this button will be disabled along with the Name drop down combo box.

Note that when creating a new super node, until you *create the super node*, any changes you make to sub-nodes of the displayed super element will modify sub-nodes of the *super element* and not a *super node*.

Defined?: This indicates whether or not the currently selected *super element* is fully defined – if "YES" then a *super node* can be instantiated from the super element, otherwise not.

Description: This displays the detailed description of the currently displayed *super element*.

3.11 Working with Species and Contaminants

Airborne contaminants are dispersed throughout buildings due to several different transport mechanisms. These include air movements into, out of, and within a building system; heating, ventilating, and air-conditioning (HVAC) systems; removal by filtration, deposition, and sorption; generation; chemical reaction; and radio-chemical decay. CONTAM provides you with the ability to simulate these methods of dispersal of airborne contaminants within built structures.

Species is a general term used (by CONTAM) to identify substances that can be used as contaminants during a simulation, i.e., contaminants are those species you designate to be used in a simulation. You can include as many species/contaminants as you need within a project.

CONTAM handles both *trace and non-trace contaminants*. Trace contaminants are those that exist at concentrations that do not cause a "significant" change in the density of air. Non-trace contaminants are those that can affect the density of the air. The mass fractions of non-trace contaminants will always sum to 1 kg/kg, i.e., they are the constituents of the air.

In versions prior to CONTAM 2.4 an underlying assumption of the model was that each zone is considered to have a single concentration value at any given time. This assumption is sometimes referred to as the "well-mixed" zone assumption. With respect to contaminant analysis, this means that there are no spatial concentration gradients within a given zone. However, beginning with version 2.4, zones can be preconfigured by the user to be 1 dimensional convection/diffusion zones in which contaminants can be allowed to vary along a user-defined axis.

You can use CONTAM to perform steady state, transient, and cyclical contaminant analysis. Steady state analysis refers to the determination of contaminant concentrations under steady flow conditions. Transient analysis refers to the determination of the time history of contaminant concentrations within each zone for a set of discrete time-steps. With a transient analysis you can implement the effects of varying ambient contaminants using a contaminant file (See Contaminant Files). A cyclic analysis determines the time history of contaminant concentrations for a 24 h period such that the concentrations at the beginning and end of the 24 h period are the same for each zone.

With CONTAM you must define contaminants in order to utilize the features that CONTAM provides to simulate the generation of contaminants by sources, the removal by sinks (including filters and kinetic reactions), and to determine occupant exposures. Contaminants must be created/defined prior to implementing any of these contaminant-dependent features of CONTAM. Prior to implementing contaminants within a project, you can first define the building layout on the SketchPad, define all airflow paths (including ducts) and test your building model by performing a steady state airflow simulation.

3.11.1 Non-trace Contaminants and Water Vapor

CONTAM handles non-trace contaminants, which are those contaminants that you define to be the constituents of air. Typically, you might want to treat water vapor as a non-trace contaminant. CONTAM will treat a contaminant with the name "H2O" as water vapor. The concentration will be affected by the humidity ratio contained in weather files that you use during simulations (See Defining Transient Weather and Wind). ContamX will calculate the concentration of H2O within each zone, as the mass fraction of water to air in units of kg_H2O

/kg_{air}, where kg_{air} = kg_{dry air} + kg_{H2O}. Whereas the humidity ratio is typically given as the ratio of the mass of water vapor to that of dry air. You can convert between the two using the following relationship.

$$\text{Humidity ratio} = \text{mass fraction of H}_2\text{O} / (1 - \text{mass fraction of H}_2\text{O})$$

For example, to create air from its typical constituents you might define the following set of non-trace contaminants (See Creating Species and Contaminants):

Name	Molar Mass [kg/kmol]	Default Concentration [kg _{cont} /kg _{air}]	Trace Contaminant	Use in Simulation
N2	28	0.7808	Non-trace	Use
O2	32	0.2095	Non-trace	Use
Ar	40	0.0093	Non-trace	Use
CO2	44	0.0003	Non-trace	Use
H2O	18	0.0000	Non-trace	Use

Note in the above example, the Σ Default Concentration = 1.0 kg/kg.

As a simpler example, you might create a dry air contaminant, Air, and a water vapor contaminant, H2O, as follows.

Name	Molar Mass [kg/kmol]	Default Concentration [kg _{cont} /kg _{air}]	Trace Contaminant	Use in Simulation
DryAir	28.965	1.000	Non-trace	Use
H2O	18.015	0.000	Non-trace	Use

3.11.2 Creating Species and Contaminants

You can create species by either activating the Project Species dialog box via the **Data → Contaminants...** menu or using the CONTAMW Data and Library Manager: Species dialog box accessed via the **Data → Species...** menu. The Project Species dialog box will enable you to create and modify species for the current project. The Library Manager will enable you to import species from species-related libraries. A detailed description of each of the contaminant properties is given in the Species Properties section of this manual. Each species has a *default concentration* associated with it that you define (or leave zero). You can use this default concentration to set the initial concentrations within a building without having to utilize a source element.

3.11.3 Deleting Species and Contaminants

You can delete species using the Delete button of the Project Species dialog box or of the CONTAMW Data and Library Manager: Species dialog box. Species can not be deleted if they are associated with a source/sink, filter or kinetic reaction. For those species for which an

association exists, the Delete buttons will be deactivated. You can also disable the use of a contaminant within a simulation by deselecting the "Use in Simulation" box for a given species from within the Species Properties dialog box.

3.11.4 Modifying Species and Contaminants

Once you have created species for a project, you can modify any of the related parameters. This is done by using the Edit button from within the Species Properties dialog box that is accessed via the **Data → Contaminants...** menu item. ContamW will allow you to change the name of an existing contaminant without affecting any of the currently associated sources, sinks, filters or kinetic reactions as they reference contaminants by an internal index created by ContamW.

3.11.5 Project Species

Species must be defined before any related elements (e.g. source/sinks, filters and kinetic reactions) can be defined. From the Project Species page any previously defined species may be viewed, edited, or deleted and new species may be defined. The following describes the information that appears on the Project Species dialog box.

Species: This is a list of the species that are currently defined within the current project.

Species Properties: These are the properties of the currently highlighted species.

Contaminants: This is a list of the project species for which the "Use in Simulation" property is set to "Use," i.e., these are selected to be contaminants in the simulation.

Contaminant Summary: This is a summary of contaminant-related information including the number of non-trace contaminants and the summation of the default concentrations of the non-trace contaminants which must be 1.0.

New, Edit and Delete: Click the "New" button to create a new species. To modify the properties of an existing species, highlight the species in the species list and click the "Edit" button. Click the "Delete" button to delete the selected species.

NOTE: Changing the number of contaminants, i.e. those species used in the simulation, will reset the initial concentrations of all zones to the default contaminant concentrations. You can reset all of these concentrations via the Run Control Properties of the simulation parameters or via the Contaminant Data properties of the individual zones.

3.11.6 Species Properties

This section provides detailed descriptions of the specific species properties.

Name: You will use this name to refer to the species throughout the program. You must give each species a name that is unique to the current project.

Molar Mass: The molar mass in g/mol or kg/kmol. CONTAM uses this value to convert volumetric concentrations to mass fractions and to calculate the gas constant of the air mixture made up by non-trace contaminants. You must enter a value of at least 1.0.

Default Concentration: CONTAM will apply this value as the default initial concentration for each zone that you create. You can revise the initial concentration for individual zones as per your requirements. This is also the ambient contaminant concentration during simulations which do not utilize contaminant data from one of the external contaminant data files (CTM or WPC).

Diffusion Coefficient: This value will be used during simulations using the *1D Duct Model* of the *Short Time Step Method* (See Run Control Properties in the Working with Simulations section). The dispersal coefficient will be calculated from this value depending on the Reynolds Number in a given duct segment.

Mean Diameter: Enter the mean particle diameter if this species is to be a particulate type species. CONTAM utilizes this value when converting between particle count units and particle mass and volume units and to determine filter efficiency of Simple Particle Filter elements for this size particle (See Simple Particle Filter in the Working with Filters section).

NOTE: This is not necessarily meant to be, for example, the aerodynamic diameter as the current contaminant source models do not treat particles different from gaseous contaminants.

Effective Density: Enter a density that you want CONTAM to use as the effective density of a species you want to consider to be a particulate type species. Currently, CONTAM only utilizes this value when converting between particle count units and particle mass and volume units.

Specific Heat: The specific heat of a contaminant will be used during simulation only when the *Variable Junction Temperatures* method is selected.

Decay Rate: This is an exponential decay constant based upon the half-life of the radioactive species calculated using the following equation:

$$\text{Decay Rate} = \ln(2) / \tau_{1/2},$$

where $\tau_{1/2}$ is the half-life of the radioactive species in seconds.

ContamW will use this value to perform unit conversions for radioactive contaminants. Radioactive species are only distinguished from other species by the units associated with them, not by modeling the radioactive decay process. To model a radioactive decay, use a kinetic reaction (See Kinetic Reaction Data).

Non-Trace Contaminant: Non-trace contaminants are those that can affect the density of the air, i.e., are considered components of the air. See Non-trace Contaminants and Water Vapor.

Use in Simulation: This determines whether or not a species will be a contaminant. Checking "Use in Simulation" indicates that a species will be a contaminant and used in the simulation.

Description: Use this to provide a more detailed description of the species.

Guideline Value: Currently not implemented.

Guideline Description: Currently not implemented.

3.11.7 Contaminant Files

You can account for changes in the ambient contaminant concentrations when performing transient contaminant simulations using ambient contaminant (.CTM) files. If you associate a contaminant file with your project (See Run Control Properties) ContamX will compare those contaminants defined within your project file with those contained in the contaminant file. If they match, ContamX will use the ambient concentrations in the contaminant file within the simulation, otherwise they will be ignored.

You create the contaminant files external to ContamW. You can use a utility program provided by NIST, Weather 2.0, to convert existing CONTAM 1.0 weather (which previously contained ambient contaminant concentrations) files to CONTAM 2.0 weather and contaminant files.

Note that with this method the ambient concentration does not vary spatially, only temporally. If you require spatial variations of outdoor contaminant level, then refer to the section entitled Working with WPC Files.

□ Contaminant (.CTM) File Format

The CONTAM contaminant (species) file is a tab-delimited ASCII text file. These files are easily created, imported and modified using common spreadsheet applications. They can be saved as tab-delimited text files for use with CONTAM. In the listing below I1 indicates a character string, I2 indicates a short integer and R4 indicates a four-byte real number. Individual lines of data are separated by dashes (these dashes are only shown here for format presentation purposes and should not be included in the actual files). Each day must begin with time 00:00:00 and end with time 24:00:00. The time interval between 00:00:00 and 24:00:00 can be either regular or irregular – ContamX will interpolate as necessary. Comments are allowed and are indicated by an "!" (exclamation point). Anything that appears on a line following a comment indicator is ignored by the contaminant file reader.

```
SpeciesFile ContamW 2.0 ! file and version identification
-----
I1 Descr[]           ! species file description
-----
I1 StartDate        ! day of year (1/1 - 12/31)
I1 EndDate          ! day of year (1/1 - 12/31)
I2 NumCont          ! number of species
-----
I1 Name[1]          ! name of species 1 (16 characters max)
I1 Name[2]          ! name of species 2
...
I2 Name[NumCont]    ! name of species NumCont
-----
!Date  Time Name[1] Name[2] ... Name[NumCont]
-----
! for each day:  start_date - end_date
! for each time: 00:00:00 - 24:00:00 increment may be variable
  I1 Date          ! date (1/1 - 12/31)
  I1 Time          ! time of day 00:00:00 - 24:00:00
```

```

R4 Conc[1]          ! concentration of species 1
R4 Conc[2]          ! concentration of species 2
...
R4 Conc[NumCont]   ! concentration of species nc
                   ! [kg_species/kg_air]
    
```

□ Editing Contaminant Files

You can import ambient files into a spreadsheet program for editing. However you must make sure that the date format is compatible with CONTAM when you save the file. One well-known spreadsheet program will convert the CONTAM date format of mm/dd into the mmm-dd form. For example the CONTAM date format for January 1 is "1/1", but the spreadsheet program will convert it to "Jan-01" unless you "force" it to import the first column of data – the one that contains date information – as text. Another option is to accept the default importing option then change the cell format for each date to the CONTAM format mm/dd prior to saving the file as tab-delimited.

□ Contaminant Unit Conversions

Gaseous contaminants – to convert from ppm to:

$$kg/kg = (ppm \times MW) / (1\,000\,000 \times V_s \times \rho_{air})$$

$$mg/L = (ppm \times MW) / (1\,000 \times V_s)$$

$$mg/m^3 = (ppm \times MW) / V_s$$

$$mg/ft^3 = (ppm \times MW) / (35.3147 \times V_s)$$

Particle Contaminants – to convert from #/m³ to kg/kg:

$$\frac{kg_{cont}}{kg_{air}} = \frac{\#}{m^3} \times \frac{\rho_{eff}}{\rho_{air}} \times \frac{1}{6} \pi D_{mean}^3$$

where:

MW = molar mass of contaminant

V_s = 24.05 m³, the volume of one mole of air at T = 20 °C and P = 101 325 Pa (1 atm)

ρ_{air} = 1.204 kg/m³ at T = 20 °C

ρ_{eff} = effective particle density [kg/m³]

D_{mean} = mean particle diameter [m³]

3.12 Working with Sources and Sinks

Sources and sinks are used to generate contaminants within and remove contaminants from zones. Sources can be used to simulate building materials such as carpet, pieces of furniture, and episodic events such as the use of cleaning products. You can define an almost unlimited number of sources within a project.

To implement a source/sink, you must first define the species with which you want the source/sink to be associated. You then place a source/sink icon within a zone on the SketchPad. Once you place source/sinks on the SketchPad, you can define their characteristics, move, copy, and delete them. You must associate each source/sink with a predefined species and provide specific information that describes its emission and/or removal characteristics.

NOTE: It is important to note that even though you associate a species with each source/sink, only those species that are designated to be contaminants will actually be used when performing simulations (See Species Properties).

Much of the information that describes the characteristics of a source/sink is contained in *source/sink elements* that you must create when defining source/sinks. The details of drawing, defining and modifying sources and sinks are described in the following sections.

3.12.1 Source/Sink Elements

CONTAM uses mathematical relationships referred to as *source/sink elements* or models to implement sources and sinks when performing contaminant simulations. CONTAM can implement several source/sink models to generate contaminants within or remove contaminants from a zone. Depending on the source/sink model, it could be used to emulate either a source, a sink or both. Each of these source/sink element types is described in detail in the Source/Sink Element Types section. Every source/sink must refer to a single *source/sink element*; however, multiple source/sinks can refer to the same *source/sink element*.

Source/sink elements can also be stored within a CONTAM library file and shared between different CONTAM project files (See Working with Data and Libraries).

3.12.2 Creating Sources and Sinks

You create source/sinks by placing a source/sink icon on the SketchPad (See Drawing Building Component Icons). You must first define at least one species before ContamW will allow you to place a source/sink icon upon the SketchPad. Each source/sink must be placed within a building zone. You cannot place them in the ambient zone. Ambient sources are handled using either the initial concentration property of a contaminant (for steady state simulations) or the concentration contained within a weather file (for transient simulations). The "Source/sink" menu selection will be disabled (grayed out) if you pop-up on a location where ContamW does not allow the icon to be placed.

3.12.3 Deleting Sources and Sinks

You delete source/sink icons using the icon deletion procedure (See Deleting Building Component Icons). Deleting source/sink icons from the SketchPad does not delete any source/sink elements that you may have defined. These elements can only be deleted using the CONTAMW Library Manager, accessible through the **Data → Source/Sinks...** menu.

3.12.4 Modifying Sources and Sinks

To modify the parameters of a source/sink, you use the icon definition procedure (See Defining Building Component Icons) to display its properties and make the desired changes. You can also move and copy source/sinks.

❑ Moving Source/Sink Icons

Once you define a source/sink, you can move the icon using the **Cu**t and **P**aste functions of the **E**dit menu. You can move the source/sink icon anywhere within the zone in which it is currently located.

❑ Copying Source/Sink Icons

Once you define a source/sink, you can copy the icon using the **C**opy and **P**aste functions of the **E**dit menu. You can copy a source/sink to any blank cell within a zone on any level of a project.

3.12.5 Defining Sources and Sinks

After you draw a *Source/Sink icon* on the SketchPad you must define it using the icon definition procedure (See Defining Building Component Icons) to display and edit the properties. This will display the "Source/Sink Properties" property sheet. Detailed descriptions of all source/sink properties are given in the Source and Sink Properties section of this manual. Once you have defined the properties, the icon will be displayed in black. From now on, you use this icon to access the properties of the source/sink.

Among the properties associated with a source/sink are optional *multiplier* and *schedule* information. You would use the multiplier to proportionally increase the generation or removal of a contaminant for a particular source/sink. For example, if you have created a *source/sink element* with a certain emission rate per square meter of surface area (such as a floor covering), you could use this multiplier to account for the surface area of the source/sink for each zone in which you place a source/sink implementing this *source/sink element*. You could use a *schedule* to vary the emission/removal characteristics of a source/sink with time. For example, you could provide a *schedule* for an episodic source (burst source) that would periodically cause the release of a contaminant when performing a transient simulation.

3.12.5.1 Source/Sink Properties

This section provides detailed descriptions of the specific source/sink properties. Each of the following subsections are the context-sensitive help topics that you can access by pressing **F1** when working with property pages of the "Source/Sink Properties" property sheet.

Source/Sink – Source/Sink Element Properties

Source/Sink Number: This is the number that appears in the status bar when you highlight an icon. ContamW automatically assigns this number to each *source/sink* once they are defined. This number could change as you add and remove source/sinks to or from a project. Each time you save a project, the source/sink numbers are reassigned. Numbers are assigned beginning on the top level, starting from the upper-left corner of the SketchPad, moving left-to-right and top-to-bottom of the SketchPad.

Source/Sink Element:

Description: Field that shows a more detailed description of the specific sink/source element if entered by the user. This description can be modified by pressing the "Edit Element" button next to the description field.

Name: <required> You may select a previously defined source/sink element from those contained in the Name list. Click the "New Element" button to define a new element. To view or modify existing element data click the "Edit Element" button. Every airflow path must have an airflow element associated with it. This field may not be left blank.

You can also access the CONTAMW Library Manager to import source/sink elements from library files by clicking the "Library" button (See Working with Data and Libraries).

Model Summary: This summarizes the information associated with the source/sink element whose name is currently displayed in the Name field.

Source/Sink – Multiplier, Schedule & Location Properties

Species: This is the species associated with the source/sink.

Multiplier: A constant value by which the source strength will be multiplied during simulation. With this feature you could define a *source/sink element* having a source strength per unit area then use the multiplier as the area of the zone for each source/sink that uses the per unit area source/sink element.

Schedule: You can use a schedule to modify the source strength as a function of time. Two source/sink element types, *Burst Source* and *Decaying Source*, require the use of a schedule. If there are previously defined schedules within the current project, you may select one from the Name list. Click the "New Schedule" button to define a new schedule. To view or modify an existing schedule click the "Edit Schedule" button. Setting this field to <none> indicates no schedule (See Working with Schedules).

Location: These are the X, Y and Relative Elevation (Z) coordinates and units for the selected source/sink. X and Y are absolute coordinates and Rel Elevation is relative to the level on which the source/sink is located. To create a point source, set the Minimum and Maximum values to be the same. To create a source which distributes emissions along the 1D axis of the zone, enter Minimum and Maximum values that span the desired region of the zone in which the source is located.

These values are required for any source/sink that is located within a *1D convection/diffusion zone* (see 1D Zone Data). These coordinates will be verified by ContamW against the axis of the 1D zone in which it is located. This verification will be performed automatically when you select either the **Run Simulation** or **Create a ContamX Input File** command from the **Simulation** menu.

Source/Sink Element Types

These are the source/sink element types or models that you can implement with CONTAM. Refer to the related information for each specific model to obtain detailed descriptions of them.

CONTAM Elements:

Constant Coefficient Model: The general source/sink model allows constant contaminant generation and deposition rates.

Pressure Driven Model: This model is for contaminant sources that are governed by the pressure differences between zones.

Cutoff Concentration Model: This model will reduce emissions as the concentration within the zone in which it is implemented approaches a specified cutoff concentration. This model may be appropriate for some sources of Volatile Organic Compound (VOC).

Decaying Source Model: This model provides a source that will exponentially decay with time according to a user-defined time constant. This model may also be appropriate for some VOC sources.

Boundary Layer Diffusion Model: Used to model reversible diffusion of a contaminant through the air-phase boundary layer of a material surface and the sorption of the contaminant into a sink with a linear sorption isotherm.

Burst Source Model: Used to model an instantaneous release of a mass of contaminant within a zone.

Deposition Velocity Sink Model: Used to model contaminant deposition as specified by a *deposition velocity*.

Deposition Rate Sink Model: Used to model contaminant deposition as specified by a *deposition rate*.

Source/Sink Element: Constant Coefficient Model

$$S = G - D \cdot C$$

G = Generation rate [mass of contaminant / time]

D = Effective removal rate [mass of air / time]

Determine by multiplying the first-order removal rate of the contaminant [1 / time] by air density and volume of the zone in which the source is to be located.

C = Current concentration [mass of contaminant / mass of air]

Name: Enter a unique name you want to use to identify the source/sink element. The element will be saved within the current project and can be associated with multiple source/sinks.

Generation Rate: The rate at which the contaminant is introduced into the zone.

Removal Rate: The rate at which the contaminant is removed from the zone.

Species: Select a previously defined species from the drop down box.

Description: Field for entering a more detailed description of the specific sink/source element.

Source/Sink Element: Pressure Driven Model

The pressure source/sink model is intended to model contaminant sources that are controlled by the pressure differences between interior and exterior zones, e.g., radon or soil gas entry. The governing equation is as follows:

$$S = G \cdot \Delta P^n$$

S = contaminant source strength

G = generation rate coefficient

ΔP = pressure difference

n = pressure exponent

Name: Enter a unique name you want to use to identify the source/sink element. The element will be saved within the current project and can be associated with multiple source/sinks.

Generation Rate: This coefficient describes the rate at which the contaminant is introduced into the zone as a function of pressure difference.

Pressure Exponent: Used to describe the dependence on pressure of the contaminant entry.

Species: Select a previously defined species from the drop down box.

Description: Field for entering a more detailed description of the specific sink/source element.

Source/Sink Element: Cutoff Concentration Model

For volatile organic compounds the source model is sometimes expressed as:

$$S = G \left(1 - \frac{C}{C_{cut}} \right)$$

S = source strength

G = generation rate coefficient

C = current concentration

C_{cut} = cutoff concentration at which emission ceases

Name: Enter a unique name you want to use to identify the source/sink element. The element will be saved within the current project and can be associated with multiple source/sinks.

Generation Rate: This coefficient describes the rate at which the contaminant is introduced into the zone as a function of concentration.

Cutoff Concentration: The concentration level where the source ceases its emission.

Species : Select a previously defined species from the drop down box.

Description: Field for entering a more detailed description of the specific sink/source element.

Source/Sink Element: Decaying Source Model

$$S(t) = G_o \cdot e^{-t/t_c}$$

G_o = initial generation rate

t = time since the start of emission

t_c = time constant

Name: Enter a unique name you want to use to identify the source/sink element. The element will be saved within the current project and can be associated with multiple source/sinks.

Initial Generation Rate: Contaminant generation is controlled by a schedule. Contaminant generation begins when the schedule changes from a zero to a non-zero value (between zero and one). The initial generation rate is equal to the schedule value times the initial generation rate. A single schedule may be used to initiate several emissions at different times.

Time Constant: The time at which the generation rate reaches 0.37 of the original rate.

Species: Select a previously defined species from the drop down box.

Description: Field for entering a more detailed description of the specific sink/source element.

NOTE: You must associate a schedule with this type of source.

Source/Sink Element: Boundary Layer Diffusion Model

The boundary layer diffusion controlled reversible sink/source model with a linear sorption isotherm follows the descriptions presented in [Axley 1991]. The boundary layer refers to the region above the surface of a material through which a concentration gradient exists between the near-surface concentration and the air-phase concentration. The rate at which a contaminant is transferred onto a surface (sink) is defined as:

$$S = h \cdot d \cdot A \left(C_i - \frac{C_s}{k} \right)$$

h = film mass transfer coefficient over the sink

d = film density of air

A = surface area of the adsorbent

C_i = concentration in air

C_s = concentration in the adsorbent

k = Henry adsorption constant or the partition coefficient

Name: Enter a unique name you want to use to identify the source/sink element. The element will be saved within the current project and can be associated with multiple source/sinks.

Film Transfer Coefficient: The average mass transfer coefficient of the contaminant within the boundary layer (or film) above the surface of the adsorbent.

Film Density of Air: This is the average of the bulk air density and the near-surface density.

Surface Mass: Enter the mass for a unit surface area and use the source/sink multiplier to enter the total area of the adsorbent.

Partition Coefficient: Coefficient that relates the concentration of the contaminant in the bulk-air to that at the surface of the adsorption material.

Species: Select a previously defined species from the drop down box.

Description: Field for entering a more detailed description of the specific sink/source element.

Source/Sink Element: Burst Source Model

The burst model simulates an instantaneous addition of contaminant mass to a zone.

Name: Enter a unique name you want to use to identify the source/sink element. The element will be saved within the current project and can be associated with multiple source/sinks.

Mass added to zone: Enter a value for the instantaneous mass release into the zone. The addition of mass is controlled by a schedule; when the schedule changes from zero to a non-zero value, mass is added to the zone in one simulation time step. The amount of mass added is equal to the mass added to zone value times the schedule value. A single schedule can initiate several events at different times.

Species: Select a previously defined species from the drop down box.

Description: Field for entering a more detailed description of the specific sink/source element.

NOTE: You must associate a schedule with this type of source.

Deposition Velocity Sink Model

This sink model is provided as a convenient means to create a sink using deposition velocities.

$$R_{\alpha}(t) = v_d A_s \rho_{air}(t) C_{\alpha}(t)$$

where:

$R_{\alpha}(t)$ = Removal rate at time t [M_{α}/T]

v_d = Deposition velocity [L/T]

A_s = Deposition surface area [L^2]

$\rho_{air}(t)$ = Density of air in the source zone at time t [M_{air}/L^3]

$C_{\alpha}(t)$ = Concentration of contaminant α at time t [M_{α} / M_{air}]

This model differs from the *Deposition Rate Sink Model* in that it is not dependent on the mass of air in the zone in which it is located.

Name: Enter a unique name you want to use to identify the source/sink element. The element will be saved within the current project and can be associated with multiple source/sinks.

Deposition Velocity: Enter deposition velocity and units. The air density will be obtained at the time of simulation from the zone in which the source/sink is located.

Surface Area: Enter desired surface area and units. You can enter a value of 1 and use the source/sink multiplier to specify the total deposition surface area.

Species: Select a previously defined species from the drop down box.

Description: Field for entering a more detailed description of the specific sink/source element.

Deposition Rate Sink Model

This sink model is provided as a convenient means to create a sink using deposition rates. This model is dependent on the mass of air in the zone in which the source/sink is located and will behave similar to that of a kinetic reaction having the same reaction rate.

$$R_{\alpha}(t) = k_d V_z \rho_{air}(t) C_{\alpha}(t)$$

where:

$R_{\alpha}(t)$ = Removal rate at time t [M_{α}/T]

V_z = Zone volume [L^3]

k_d = Deposition rate [$1/T$]

$\rho_{air}(t)$ = Density of air in the source zone at time t [M_{air}/L^3]

$C_{\alpha}(t)$ = Concentration of contaminant α at time t [M_{α} / M_{air}]

Name: Enter a unique name you want to use to identify the source/sink element. The element will be saved within the current project and can be associated with multiple source/sinks.

Deposition Rate: Enter deposition rate and units. The volume and density will be obtained at the time of simulation from the zone in which the source/sink icon is located.

Species: Select a previously defined species from the drop down box.

Description: Field for entering a more detailed description of the specific sink/source element.

3.13 Working with Filters

CONTAM provides some building components with the ability to incorporate filters that act as contaminant removal devices. Filters consist of two parts: the *filter* itself which is associated with a specific building component (path or duct segment) and *filter elements* that you can associate with one or more filters. Filter elements provide the physical characteristics of the filter and the mathematical relationship that describes the behavior of the filter during a simulation. This is similar in nature to the relationship between airflow paths and airflow elements and between duct segments and duct flow elements. In another sense, the filter is a specific instance of a filter element.

The following building components can incorporate filters and are described as *filter-ready*:

1. Airflow paths
2. Implicit outdoor air and recirculation air paths of simple air-handling systems
3. Supplies and returns of simple air-handling systems
4. Duct segments
5. Duct terminals

3.13.1 Filter Elements

CONTAM uses mathematical relationships referred to as *filter elements* or models to implement filters when performing contaminant simulations. CONTAM can implement several different filter models to remove contaminants as they are transported through building components that have filtering capabilities. These element types are described in detail in the Filter Element Types section. Every *filter* must refer to a single *filter element*; however, multiple *filters* can refer to the same *filter element*.

Filter elements can also be stored within a CONTAM library file and shared between different CONTAM project files (See Working with Data and Libraries).

3.13.2 Creating Filters

You create filters for each building component for which you require a filter. When you are defining one of the *filter-ready* building components presented above, a filter property page will be provided that will enable you to establish whether or not to associate a filter, or to modify or delete a previously defined filter for that component.

3.13.3 Deleting Filters

You can delete filters by accessing the properties of the individual component whose filter you want to delete or by deleting the building component itself. Note that deleting a *filter* does not delete the associated *filter element* (you can not delete a *filter element* that is currently associated with a *filter*).

3.13.4 Modifying Filters

You modify a filter by accessing the Filter property page of the individual building component icon and selecting the "Edit Filter" button to display the "Filter" dialog box.

3.13.5 Filter Properties

Filter Number: This number is automatically assigned by ContamW and can change as you add and remove filters to or from a project. Each time you save a project, the filter numbers are reassigned.

Even though this is the actual filter number used by CONTAM, a different number will be used when presenting filter accumulation results in the *contaminant summary file* (.csm file). This file will present filters according to the building component with which they are associated. For example, filters associated with airflow paths and air handling system supply and return points will be referenced by the path number and the letter "p" to indicate it is a path filter. Filters associated with ducts will be referenced by the duct segment number and the letter "d" or the terminal number and the letter "t" for terminal.

Filter Element

Name: <required> You may select a previously defined filter element from those contained in the Name list. Click the "New Element" button to define a new filter element. To view or modify filter element data click the "Edit Element" button. You can also access the CONTAM Library Manager to import filter elements from library files by clicking the "Library" button (See Working with Data and Libraries). Every filter must have a filter element associated with it.

Description: Displays the description of the filter element currently displayed in the Name field.

Type: Displays the type of filter element for the filter currently displayed in the Name field.

Initial Filter Loading

Total Relative Load: Set the initial loading of the filter relative to the mass of the filter. This can be set to zero to represent a "clean" filter at the beginning of a simulation; otherwise set the value to establish the total mass of contaminants (as a percent of the mass of the filter) accumulated on the filter at the start of a simulation. Relative load is only used for the Gaseous Filter element which is currently the only element for which efficiency is described as a function of filter loading.

3.13.6 Filter Element Types

These are the filter element types or models that you can implement with CONTAM. Refer to the related information for each specific model for detailed descriptions.

Constant Efficiency Filter: This model provides for the definition of filters having constant efficiencies for each user-selected species.

Simple Gaseous Filter: This model provides for the definition of filter efficiency as a function of filter loading for a user selectable set of species.

Simple Particle Filter: This model provides for the definition of filter efficiency as a function of particle size, for example a MERV curve.

3.13.6.1 Constant Efficiency Filter

The *constant efficiency filter* element is made up of a set of species and associated efficiencies. Each filter element can be associated with anywhere between one and the number of species that exist within the current project. Only those species that you have selected to be contaminants will be accounted for when you perform simulations.

Name: A unique name you want to use to identify the filter element.

Area: The face area and units of the filter element.

Depth: The depth of the filter element along the axis of airflow.

Density: The density of the filter media.

Area, Depth and Density are currently used to calculate the mass of the filter media which is only relevant for the Simple Gaseous Filter model.

Description: Use this field to provide a more detailed description of the filter element.

Edit Element Data: Select to define the specific properties of the filter element.

Filtered Species: This is a list of the species and filter efficiencies for the filter element.

Edit Filter Data: Use the *Species* and *Efficiency* fields to set and modify the *Filtered Species* list along with the *Add*, *Replace* and *Delete* command buttons.

Efficiency: Use this to enter the *filter efficiency* for the currently highlighted species. Filter efficiencies must be between 0.0 and 1.0.

Species: This is a list of all the *species* in either the current project or library file depending on whether you are editing project data or library data. Select the *species* for which you want to set a filter efficiency.

3.13.6.2 Simple Gaseous Filter

The *simple gaseous filter* element is made up of a set of species and associated efficiency vs. loading curves. Therefore, the efficiency is a function of the amount of contaminant "absorbed" by the filter. Each filter element can be associated with anywhere between one and the number of species that exist within the current project. Only those species that you have selected to be contaminants will be accounted for when you perform simulations.

Name: A unique name you want to use to identify the filter element.

Area: The face area and units of the filter element.

Depth: The depth of the filter element along the axis of airflow.

Density: The density of the filter media.

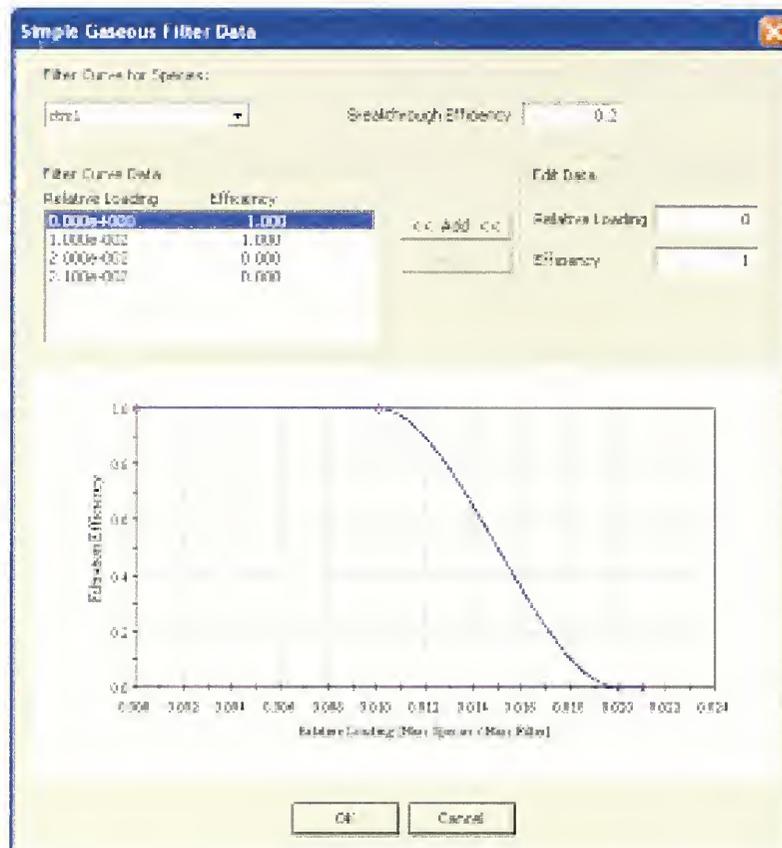
Area, Depth and Density are currently used to calculate the mass of the filter media which is in turn used to calculate filter loading.

Description: Use this field to provide a more detailed description of the filter element.

Edit Element Data: Select to define the specific properties of the filter element.

Filter Curve Data: This is a list of relative loading and filter efficiency data pairs used to define the filter curve for the currently selected species (see figure below). Once you have

entered at least three data points, ContamW will attempt to calculate a cubic spline fit to the points. If there are any errors in the spline fit procedure, you will be notified and the offending portion of the curve will be highlighted.



Species: This is a list of all the *species* in either the current project or library file depending on whether you are editing project data or library data. Select the *species* for which you want to display/edit filter curve data.

Breakthrough Efficiency: Set the breakthrough efficiency for each species. This value will be used during simulation to report if and when the filter efficiency drops below this value. Breakthrough is reported to the *Contaminant Summary* simulation result file (See Results Files in the Working with Results section).

Filter Curve Data: This is the list of data points you create from which a *filter curve* is generated using the cubic spline fit method. A minimum of four data points is required before a curve will be generated. The space below will provide a plot of *Relative Loading* vs. *Filter Efficiency*. *Relative loading* is the ratio of Mass of contaminant accumulated by the filter to the Mass of the filter.

Use the **Relative Loading** and **Efficiency** data entry fields to set and modify the *Filter Curve Data* list along with the **Add** and **Delete** buttons.

3.13.6.3 Simple Particle Filter

The *simple particle filter* element consists of a single efficiency vs. particle size curve. In this manner, a single filter element can be used to filter multiple particle contaminants based on particle size alone. This can be used to define filters based on MERV test data [ASHRAE 1999a]. Only those species considered particle types and which you have selected to be contaminants will be filtered when you perform simulations.

Name: A unique name you want to use to identify the filter element.

Area: The face area and units of the filter element.

Depth: The depth of the filter element along the axis of airflow.

Density: The density of the filter media.

Area, Depth and Density are currently used to calculate the mass of the filter media which is only relevant for the Simple Gaseous Filter model.

Description: Use this field to provide a more detailed description of the filter element.

Edit Element Data: Select to define the specific properties of the filter element.

Filter Curve Data: This is a list of *particle size* and *filter efficiency* data pairs used to define the filter curve. Once you have entered at least three data points, ContamW will attempt to calculate a cubic spline fit to the points. If there are any errors in the spline fit procedure, you will be notified and the offending portion of the curve will be highlighted.

Use the **Particle Size** and **Efficiency** fields to set and modify the *Filter Curve Data* list along with the **Add** and **Delete** command buttons.

Particle Size: Use this to enter the particle size and units for which to define a *filter curve* data point. This corresponds to the *Mean Diameter* property of a species.

Efficiency: Use this to enter the *filter efficiency* for the associated particle size. Filter efficiencies must be between 0.0 and 1.0.

At the beginning of a simulation, ContamX will use the curve to establish a filter efficiency for each particle contaminant having a mean diameter in the range of the filter curve.

3.13.6.4 Super Filter

Use this element to combine multiple filter elements, in series, into a single filter.

Name: A unique name you want to use to identify the super filter element.

Description: Use this field to provide a more detailed description of the filter element.

Sub-Elements: You build this list of sub-elements from the list of *Available Elements*. The order in which these elements appear in the list is important. They should be placed in the order in which air is to flow through the filter from top to bottom in the list. Use the Add, Delete and Move buttons to modify this list.

Available Elements: This is the list of fundamental filter elements, e.g., *constant efficiency*, *simple gaseous*, and *simple particle*, contained in the current project file. Use the *Edit Element* and *New Element* buttons to modify this list.

3.14 Working with Kinetic Reactions

CONTAM provides the ability to model chemical reactions among project contaminants within zones of a building including the implicit supply and return nodes of a simple air-handling system (See Air Handling System). Kinetic reactions in CONTAM are modeled as first-order exponential functions between species that contain the paired first-order reaction rate coefficients between the reactant and product contaminants. See Axley [1988] and Axley [1995] for a detailed presentations of the kinetic reaction simulation method employed by ContamX. The production or destruction of a contaminant α is given by

$$R_{\alpha} = \sum_{\beta} (K_{\alpha,\beta} \cdot C_{\beta})$$

where

R_{α}	rate of production or destruction of contaminant α
$K_{\alpha,\beta}$	reaction rate coefficients between contaminants α and β , in units of 1/s
C_{β}	concentration of a reactant

CONTAM uses the convention that positive reaction coefficients produce an increase in species concentration and negative coefficients decrease species concentrations. The units of concentration are kg_species / kg_air (See Concentration Conversions).

Kinetic reactions can occur within zones of a building. To use kinetic reactions, you must have at least one species for the project. Kinetic reactions can be associated with one or more zones within a project. You can access the kinetic reaction data dialog box via the property sheets of the zones that provide a kinetic reaction definition section (i.e. zones and simple air-handling systems).

You can also access the Kinetic Reaction Data dialog box using the CONTAMW Data and Library Manager: Kinetic Reactions to create and edit local project data or to share kinetic reactions between CONTAM projects using contaminant libraries.

When you are defining building zones, you will be able to select a currently defined kinetic reaction to associate with it, define new, modify existing or import kinetic reactions from species-related CONTAM library files (i.e. LB0 library files). All kinetic reactions are defined and modified using the Kinetic Reaction Data dialog box as described in the following section.

3.14.1 Kinetic Reaction Data

Kinetic reaction elements are made up of source/product species pairs. You can define any number of pairs per element you desire. Only those species that you have selected to be contaminants will be accounted for when you perform simulations. These are the properties that describe a kinetic reaction.

Name: Enter a unique name you want to use to identify the reaction matrix. You will use this name to associate this element with zones within the project.

Description: Use this to provide a more detailed description of the kinetic reaction.

Reactant→Product Pairs: This is a list of the reactant/product pairs and their associated reaction coefficients that are currently defined for the current kinetic reaction element.

Edit Reactant→Product Pair: Use this section to edit the highlighted Reactant→Product pair or create new ones. Use the Add, Replace and Delete buttons accordingly.

Reactant Species: This is a list of all the species in either the current project or library file depending on how this dialog box was accessed. Select a species to be the reactant in the R→P Pair.

Product Species: This is a list of all the species in either the current project or library file depending on how this dialog box was accessed. Select a species to be the product in the R→P Pair.

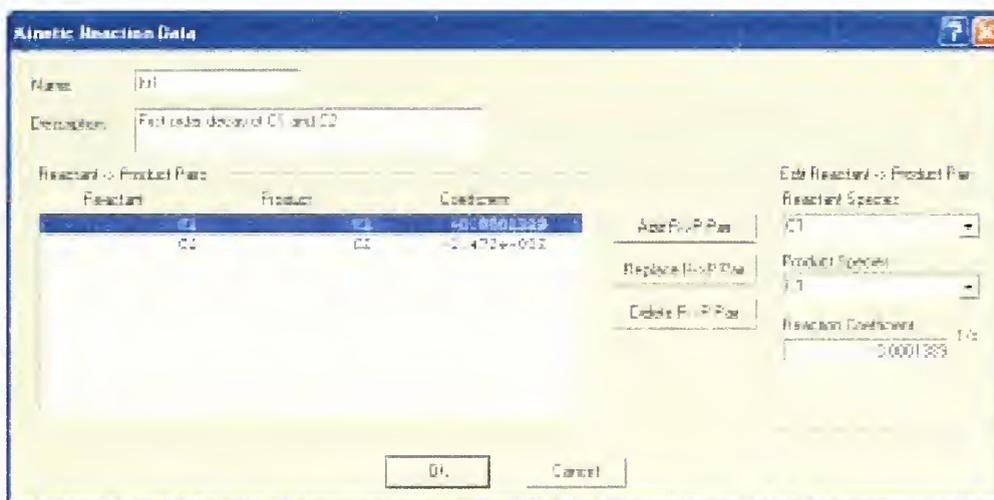
Reaction Coefficient: Use this edit field to set the reaction coefficient for the currently highlighted R→P Pair. The units of the coefficient are s^{-1} .

Examples

The following are examples of kinetic reactions as implemented within CONTAMW.

□ Kinetic Reaction Example 1

To simulate the first-order decay of a species, you would set the coefficient of the Reactant→Product Pair to the negative of the reaction rate in units of 1/s.



This figure shows kinetic reaction coefficients for two species reactions that will yield exponential decays of each species according to the following equation:

$$C_j(t) = C_i(0)e^{K_{ji}t}$$

Where,

$C_j(t)$ = concentration of product contaminant j at time t

$C_i(0)$ = initial concentration of reactant contaminant i at time 0

K_{ji} = reaction rate coefficient between product j and reactant i (time constant)

In the above example, the reaction rate coefficients are as follows:

$$K_{11} = -0.0001388/s$$

$$K_{22} = -0.00003472/s$$

If the reaction is the only means of species removal, i.e., there is no dilution due to airflow, then species C1 would decay at a rate of

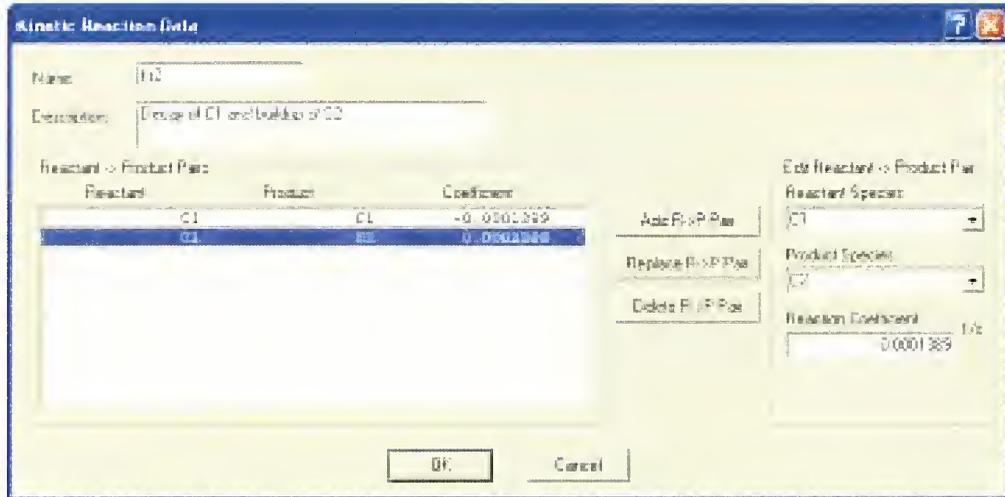
$$0.0001389 * 3600 = 0.5/\text{h},$$

and C2 would decay at a rate of

$$0.00003472 * 3600 = 0.125/\text{h}.$$

□ **Kinetic Reaction Example 2**

This example illustrates the first-order decay of species C1 (reactant and product) and the first order build-up of another species C2 (product) based on C1(reactant).



The figure above shows the parameters for the "kr1" kinetic reaction element. This element provides for two reactions that will yield an exponential decay of contaminant C1 according to the equation

$$C_1(t) = C_1(0)e^{K_{11}t},$$

and the build-up of contaminant C2 according to

$$C_2(t) = C_1(0)(1 - e^{K_{21}t})$$

In the above example, the reaction rate coefficients are as follows:

$$K_{11} = -0.0001389/\text{s}$$

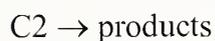
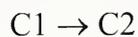
$$K_{21} = 0.0001389/\text{s}$$

Note that both the decay and build-up are dependent on the species of C1.

□ **Kinetic Reaction Example 3**

You can also use the kinetic reaction to simulate a consecutive first-order reaction series. (The radioactive decay chain of Radon gas is an example of such a reaction.)

A two-step consecutive reaction series involving a single reactant at each step is indicated as



You would model this reaction chain by inputting coefficients based on the half-life of the radioactive reactants. Convert half-life $\tau_{1/2}$ to a first-order reaction coefficient, K_{21} , using the following equation:

$$K = \ln(0.5) / \tau_{1/2}$$

In CONTAM, you would require two species C1 and C2 (if interested in contaminant C2) and three reactant/product pairs have the following reaction coefficients characteristics:

K_{11} would have negative non-zero values (decay of C1)

K_{21} would have a non-zero positive value (build-up of C2 from C1)

K_{22} would have negative non-zero values (decay of C2)

3.15 Working with Occupant Exposure

CONTAM provides you with the ability to simulate the existence of *occupants* within a building. You can simulate the movement of occupants throughout the building, allow the occupant to leave the building, and determine the contaminant exposure of each occupant. Occupants can also generate contaminants within a building, so you can represent a contaminant source that moves through the building using an occupant.

Occupant exposure is determined by integrating the contaminant concentration to which the occupant is exposed over a period of time.

$$E = \int_{t_1}^{t_2} C(t) dt$$

CONTAM presents results in terms of $E/\Delta t$. Where Δt is the user-defined *output time step* established for a transient simulation (See Results Files in the Working with Results section). From these results you can calculate a dosage as desired.

To implement an occupant, you must first define at least one species. You then place an *exposure icon* (shown below) on the SketchPad. Once you place the exposure icon on the SketchPad, you can define its characteristics, move, copy, and delete it. You must then define an occupant schedule for the exposure icon. The details of drawing, defining and modifying occupants are described in the following sections.

Occupant exposure icon 

3.15.1 Creating Occupants

You create an occupant by placing an exposure icon on the SketchPad (See Drawing Building Component Icons). You can place them in any blank cell of the SketchPad. The "Exposure" menu selection will be disabled (grayed out) if you pop-up on a location where ContamW does not allow the icon to be placed.

3.15.2 Deleting Occupants

You delete exposure icons using the icon deletion procedure (See Deleting Building Component Icons). Deleting occupant exposure icons from the SketchPad deletes the occupant data associated with the icon, but it does not delete any occupant-related schedules that you might have created. These schedules can only be deleted using the CONTAMW Library Manager, accessible through the **D**ata menu for each type of data element.

3.15.3 Modifying Occupants

To modify the parameters of an occupant, you use the icon definition procedure (See Defining Building Component Icons) to display its properties and make the desired changes. You can also move and copy occupants.

Moving Occupant Exposure Icons

Once you define an occupant, you can move the icon using the **C**ut and **P**aste functions of the **E**dit menu. You can move the occupant exposure icon anywhere within the level upon which it is currently located.

□ **Copying Occupant Exposure Icons**

Once you define an occupant, you can copy the icon using the **C**opy and **P**aste functions of the **E**dit menu. You can copy an occupant to any blank cell on any level of a project.

3.15.4 Defining Occupants

After you draw an exposure icon on the SketchPad you must define it using the icon definition procedure (See Defining Building Component Icons) to display and edit the properties. This will display the "Occupant Exposure Properties" property sheet. Detailed descriptions of all occupant exposure properties are given in the Occupant Properties section of this manual. Once you have defined the properties, the icon will be displayed in black. From now on, you use this icon to access the properties of the occupant.

Among the optional properties associated with an occupant are *multiplier*, *inhalation*, and *contaminant generation* information. You would use the multiplier to proportionally increase the contaminant generation rate associated with an individual exposure icon to represent the contaminant generation of multiple occupants. You would input contaminant generation information if you want to account for the contaminants generated by an occupant while they are in the building. Inhalation information is useful for determining the dose received by an occupant based upon their exposure to a contaminant. CONTAM does not currently implement the calculation of contaminant dose, however, you can use the exposure information to determine the dose of an occupant.

3.15.4.1 Occupant Properties

This section provides detailed descriptions of the specific occupant properties. Each of the following subsections are the context-sensitive help topics that you can access by pressing **F1** when working with property pages of the "Occupant Exposure Properties" property sheet.

Occupant – Occupant Data Properties

Occupant Number: This is the number that appears in the status bar when you highlight an icon. ContamW automatically assigns this number to each exposure icon once it is defined. This number could change as you add and remove occupants to or from a project. Each time you save a project, the occupant numbers are reassigned. Numbers are assigned beginning on the top level, starting from the upper-left corner of the SketchPad, moving left-to-right and top-to-bottom of the SketchPad.

Body weight/Peak inhalation rate: CONTAM uses these parameters to compute a contaminant dose. Inhalation is defined in terms of a peak rate and a schedule. Body weights for typical persons and inhalation values for different activity levels are available in the [EPA 1989].

NOTE: This feature and the associated Inhalation Schedule are not implemented in CONTAM version 2.0 and beyond, because contaminant dose calculation is no longer performed by CONTAM. However, occupant exposure is still determined from which you can obtain your own dose calculations.

Description: Field for entering a more detailed description of the specific exposure.

Inhalation Schedule: Use this when you want to modify the peak inhalation rate according to a schedule. If there are previously defined schedules within the current project, you may select one from the Name list. Click the "New Schedule" button to define a new schedule. To view or

modify an existing schedule click the "Edit Schedule" button. Setting this field to <none> indicates no schedule. (See Working with Schedules)

Edit Occupancy Schedule: Press the "Edit Occupancy Schedule" button to display the Occupancy Week Schedule dialog box and define the movement of this occupant within the building.

Occupant – Contaminant Generation Properties

Multiplier: Use the multiplier to modify the generation rate of an occupant. You could define the contaminant generation rate(s) for a single occupant and use the multiplier to represent a generation rate of multiple occupants.

Generate Contaminants: This checkbox is used to turn on and off the contaminant generation of the occupant. If the box is checked, contaminants will be generated. This is useful for contaminant generation such as that of carbon dioxide due to human metabolism.

Generation Data: A list of available contaminants is displayed in the textbox provided. You can highlight these contaminants and select the generation rate for each in the edit box below the contaminant list.

Generation Schedule: A new inhalation schedule is defined by pressing the "New Schedule" button and following the directions for editing a Week Schedule.

Description: Field that shows a more detailed description of the specific schedule if entered by the user. This description can be modified by pressing the edit schedule button next to the description field and then changing the description field for the schedule.

Name: This drop down box contains the names of all previously created schedules. The schedule selected from this box will then define the generation schedule for the exposure element being viewed.

3.16 Working with Data and Libraries

CONTAM enables you to share certain types of data between building components contained within a project as well as between projects. The sharing of data within a project (locally) is accomplished by allowing you to associate multiple building components with *data elements* defined within a project. Sharing data between projects is accomplished by exporting data elements from one project into CONTAM library files and then retrieving the data elements into another project from the library files. The following table lists the different types of data elements implemented by CONTAM. All of the data elements can be shared within a project, and all but occupancy schedules can be shared between projects.

Library/Local Data Elements	Local-Only Data Elements
Contaminant-related (LB0)	Occupancy Schedules
Species	
Source/Sink Elements	
Contaminant Filters	
Kinetic Reactions	
Schedules (LB1)	
Wind Pressure Profiles (LB2)	
Airflow Elements (LB3)	
Ductflow Elements (LB4)	
Control Super Elements (LB5)	

□ Data Elements

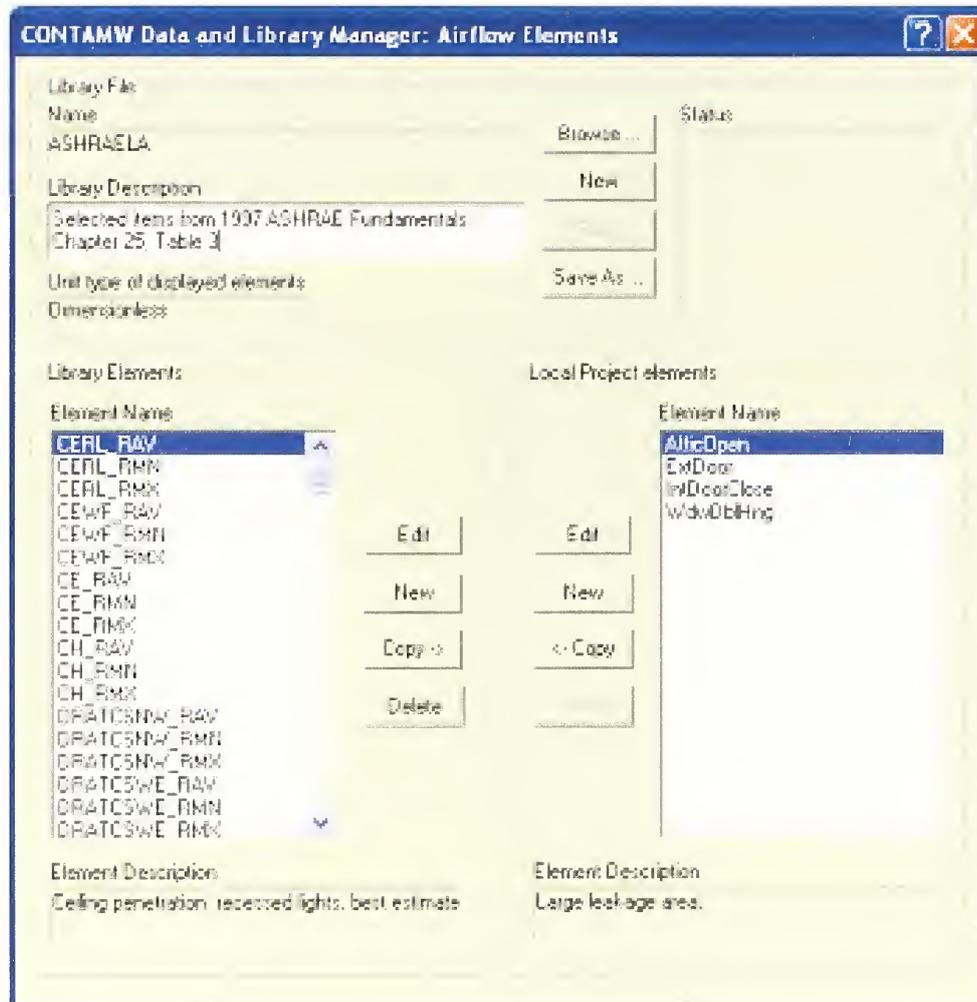
You create and modify data elements according to the procedures given throughout the "Using CONTAM" section of this manual for the specific data elements. Once you have created a data element, you can associate it with the relevant building components as you create them. You do this by selecting the desired element from the local list that ContamW presents to you when you are editing the properties of a particular building component. To share data elements between project files, you must utilize the CONTAMW Library Manager. You must also use the CONTAMW Library Manager to delete data elements from either the local project or the library. To delete local data elements, they must not be referenced by any building components.

□ Library Files

CONTAM Libraries are files that contain the descriptions of sharable data elements. Each type of sharable data element is stored in a different library file. You can create as many of each type of library file that you want, and store them anywhere within the file structure of your computer. Each type of library file is associated with a different three-character extension as shown in the table above.

❑ **CONTAMW Library Manager**

The CONTAMW Library Manager, shown in the following figure, is a dialog box that is displayed when you access data elements from the **Data** menu or by clicking on a "Library..." button when editing the properties of one of the building components that utilizes library data elements. This dialog box is divided into three sections: Library File, Library Elements and Local Project Elements sections. The Library File section displays the name of the currently displayed library file and allows you to save, rename and open new and existing library files. The Library Elements section displays the data elements contained in the library file that is currently listed in the Library File section. The Local Project Elements section displays the data elements that are contained in the current project file.



3.16.1 Opening Libraries

If a library file is currently opened, the name of the file will be displayed in the "Name" field of the "Library File" section of the Library Manager. You can open a CONTAM library file using the "Browse..." button of the Library Manager. This will activate the File Open dialog box typical of Windows applications. This dialog box is set to display only those files having the three-character file extension associated with the type of data elements that you are currently working with. Follow the typical procedure for opening a file using the Windows operating system.

CONTAM 2.0 library files are of a different format than the 1.0 version. ContamW will allow you to open a 1.0 version file and you will then be required to give the file a name when saving the converted file. It is suggested that you save the file under a different name from that of the 1.0 formatted library file. The Open dialog box will display both the file *Version* information and the Library file *description*.

3.16.2 Creating Libraries

Using the CONTAMW Library Manager, you create new libraries one of two ways. You can either save an existing library under a new name using the "Save As..." button, or use the "New" button of the Library File section of the CONTAMW Library Manager dialog box. The "Save As..." method makes a copy of the currently displayed library file under a new name, but maintains the original file. The "New" method clears the data currently displayed and allows you to start with an empty library file.

3.16.3 Deleting Libraries

You must use the file deletion commands of your operating system to delete CONTAM library files.

3.16.4 Modifying Libraries

You modify CONTAM library files using the CONTAMW Library Manager. The CONTAMW Library Manager allows you to create new data elements within the library, copy data elements from the local project into the library, modify library data elements, and delete data elements from the library.

□ Creating Library Data Elements

You use the "New" button contained within the "Library Elements" section of the CONTAMW Library Manager to create a new data element. You can only create the type of data element that is currently being displayed by the Library Manager (e.g. airflow elements, ductflow elements, schedules, etc.). The "New" button will activate the ContamW dialog box associated with creating the type of data element that you are currently working with.

Species must be available in order to create new contaminant-related elements such as Sources/Sinks, Filters and Kinetic reactions.

□ Copying Library Data Elements (between the Project and Library)

You can either copy data element from a library to the local project or from the local project to the library. All copying is done between the local project and the library file that is currently indicated in the "Name" box of the Library File section of the CONTAMW Library Manager. If the name box is blank, then you are dealing with a new/unnamed library file. To copy data elements, you use the "Copy ->" and "<- Copy" buttons. These copy buttons indicate the direction of the copy operation. If an element exists in both the library file and the local project, then the copy buttons will be disabled to prevent you from overwriting elements with the same name. You may rename local data elements without affecting their associations with building components that refer to them in the local project.

Some elements rely on other types of elements for proper implementation, e.g., source/sinks require an associated species and week schedules require day schedules. If you copy one of these elements that requires a sub element, ContamW will prompt you to replace existing elements of the same name if they already exist within the destination file.

❑ Modifying Library Data Elements

To modify a library data element, you must first highlight the Element Name within the list of elements in the Library Elements section of the CONTAMW Library Manager. You then click on the "Edit" button next to the list. This will activate the ContamW dialog box associated with displaying the properties of the type of data element with which you are currently working. You then make the desired changes and click the "OK" button to make the changes take affect.

❑ Deleting Library Data Elements

To delete a library data element, you highlight the Element Name in the Library Elements section of the CONTAMW Library Manager then click on the "Delete" button next to the list. You will be prompted to confirm the deletion, and the data element will no longer be displayed in the list of Library Elements. However, the deletion will not be affected until you save the library file. You can retrieve the previously deleted data elements by reopening the library file as long as you don't save the file after deleting the elements.

Deleting data elements that have associated sub-elements will not delete the sub-elements. You must delete the sub-elements using a separate Library Manager operation.

3.16.5 Modifying Local Data

You modify local lists of data elements using the CONTAMW Library Manager. The Library Manager allows you to create new local data elements, copy data elements from the library into the local project, modify local data elements, and delete local data elements.

❑ Creating Local Data Elements

You use the "New" button contained within the "Project Elements" section of the Library Manager to create a new data element. You can only create the type of data element that is currently being displayed by the Library Manager (e.g. airflow elements, ductflow elements, schedules, etc.). The "New" button will activate the ContamW dialog box associated with creating the type of data element that you are currently working with.

Species must be available in order to create new contaminant-related elements such as Sources/Sinks, Filters and Kinetic reactions.

❑ Copying Local Data Elements (between the Project and Library)

See Copying Library Data Elements.

❑ Modifying Local Data Elements

The Library Manager provides you with another means of accessing local project elements. This is in addition to the method of accessing building component properties presented throughout the Using CONTAM section of this manual. You must first highlight the Element Name within the list of elements in the Project Elements section of the Library Manager. You then click on the "Edit" button next to the list. This will activate the ContamW dialog box associated with displaying the properties of the type of data element that you are currently working with. You then make the desired changes and click the "OK" button to make the changes take affect.

❑ Copying Data Elements (within a Project)

You can use the Library Manager to create elements that are similar to existing elements but have a different name. To do this you copy a project element to the library (you don't have to have an existing library file open), give the library element a new name, make the modifications you desire and copy the element back into your project.

□ **Deleting Local Data Elements**

You can only delete local data elements if they are not referenced by any building components within the project. To delete a project data element, you highlight the Element Name in the Project Elements section of the CONTAMW Library Manager then click on the "Delete" button next to the list. The "Delete" button will only be enabled if the currently highlighted element is not referenced within the project. You will be prompted to confirm the deletion, and the data element will no longer be displayed in the list of Project Elements. However, the deletion will not be affected until you save the project file. You can retrieve the previously deleted data elements by reopening the project file as long as you don't save the file after deleting the elements.

Deleting data elements that have associated sub-elements will not delete the sub-elements. You must delete the sub-elements using a separate Library Manager operation.

3.17 Working with Weather and Wind

CONTAM enables you to incorporate the effects of weather on a building. Weather parameters include ambient temperature, barometric pressure, wind speed, wind direction and also outdoor contaminant levels. The type of weather data you require for a simulation depends on the type of simulation you are performing, whether you want to account for the effects of wind, and whether or not you are simulating contaminants. If you are performing a steady-state airflow simulation, you will only need steady state weather and wind data. If you are performing a transient airflow simulation, you can use either steady state or transient weather data. Transient weather data is implemented through the use of weather files or WTH files. If you are performing transient contaminant simulations, you can use either steady-state or transient ambient contaminant data as well. Transient ambient contaminant data is implemented via the use of ambient contaminant files or CTM files (See Contaminant Files).

NOTE: Both the WTH and CTM files assume a single set of values for the ambient zone at any given point in time, i.e., there is no spatial variation in either the pressure or contaminant fields. However, CONTAM also allows the use of transient wind pressure and contaminant files (WPC file) with which you can account for spatial variation in both the pressures and contaminant concentrations over the surface of a building envelope (See Working with WPC Files).

□ **Steady State Weather**

When you use steady state weather data, ContamX keeps the ambient temperature, barometric pressure, wind speed and direction and relative humidity constant during a simulation. Detailed descriptions of these parameters are given in the following sections.

□ **Transient Weather**

You can use transient weather data to simulate the changing outdoor weather and wind conditions when performing a transient simulation. Transient weather data is stored in a weather file that you must create according to a special format as specified under the Weather File Format section that follows. Weather files contain ambient temperature, pressure, wind speed and direction and humidity ratio for discrete intervals of each day for at least one day and up to one year. If the intervals between data in the weather file don't match the simulation time step, ContamX will linearly interpolate the weather data to obtain a value at the required time step.

□ **Wind**

Wind pressure can be a significant driving force for air infiltration through a building envelope. It is a function of wind speed, wind direction, building configuration, and local terrain effects. CONTAM enables you to account for the effects of wind pressure on flow paths through the building envelope (external airflow paths). You define how you want to account for wind effect for each external flow path on a case-by-case basis. In ContamW, you can either choose to ignore the effects of wind on an envelope penetration, define wind pressure to be constant for each envelope penetration, or implement variable wind pressures for envelope penetrations.

CONTAM provides a general approach to handling the variable effects of wind on the building envelope. This approach requires you to provide CONTAM with information related to the determination of a local wind pressure coefficient for the building surface.

For a general introduction to the effects of wind pressure on buildings, see Chapter 16, Air flow Around Buildings, in the 2005 ASHRAE Fundamentals Handbook [ASHRAE 2005]. The following is an overview of how CONTAM handles these effects.

The equation for wind pressure on the building surface is

$$P_w = \frac{\rho V_H^2}{2} C_p$$

where

- V_H Approach wind speed at the upwind wall height (usually the height of the building)
- C_p Wind pressure coefficient

The wind pressure coefficient can be further generalized in terms of a local terrain effects coefficient and the direction of the wind relative to the wall under consideration. The following equation is that used by CONTAM when calculating wind pressures on the building.

$$P_w = \frac{\rho V_{met}^2}{2} C_h f(\theta)$$

where

- ρ ambient air density
- V_{met} wind speed measured at meteorological station. This value is obtained from the Weather file or Steady state weather data during simulation.
- C_h wind speed modifier coefficient accounting for terrain and elevation effects
- $f(\theta)$ coefficient that is a function of the relative wind direction. CONTAM refers to this function as the *wind pressure profile*.

The relative wind direction is given by

$$\theta \equiv \theta_w - \theta_s$$

where

- θ_w wind azimuth angle (N = 0°, E = 90°, etc.), and
- θ_s surface azimuth angle.

$\rho V_{met}^2 / 2$ in the above equation is computed by ContamX from the steady state or weather file temperature, pressure and wind speed data.

C_p , the pressure coefficient used by ASHRAE, is equivalent to $f(\theta)$ in the CONTAM formulation. In ContamW, you must define the function $f(\theta)$ in the form of *wind pressure profiles* as explained later. See page 27.5 of the 2005 ASHRAE Fundamentals Handbook [ASHRAE 2005] for details relating to pressure coefficient data. There are some examples of wind pressure profiles available in the form of CONTAM library files on the NIST website (<http://www.bfrl.nist.gov/IAQanalysis/CONTAMWdatalib.htm>). Cp Generator is another utility available from TNO Building and Construction Research at <http://cpgen.bouw.tno.nl>.

The wind pressure modifier, C_h , accounts for the difference between V_{met} and V_H . The value for C_h will typically be constant for all openings on a given building. However, ContamW allows you to define a default value that will be used for each flow path connected to the ambient (See Wind Properties) or to set this value individually for each of these paths. ContamW uses the following formula to compute the value of C_h :

$$C_h = \frac{V_H^2}{V_{met}^2} = A_o^2 \left(\frac{H}{H_{ref}} \right)^{2a}$$

where H is the wall height and A_o and a depend on the terrain around the building [ASHRAE 1993, p 14.3]:

Terrain Type	Coefficient (A_o)	Exponent (a)
Urban	0.35	0.40
Suburban	0.60	0.28
Airport	1.00	0.15

The wind speed, V_{met} , is usually measured at an airport (open terrain) at an elevation, $H_{met} = 10$ m, above the ground. The wind speed, V_o , at that same elevation at the building site is given by

$$V_o = A_o V$$

The wind speed, V_H , at the top of the wall, elevation H , is then given by

$$V_H = V_o \left(\frac{H}{H_{met}} \right)^a$$

An updated method, from that presented in ASHRAE 1993, of accounting for local terrain effects is presented Chapter 16 of ASHRAE 2005. Using this method, V_H is calculated according to the following equation:

$$V_H = V_{met} \left(\frac{\delta_{met}}{H_{met}} \right)^{a_{met}} \left(\frac{H}{\delta} \right)^a$$

where

- δ_{met} Wind boundary layer thickness for the meteorological station
- δ Wind boundary layer thickness for the local building terrain
- a_{met} Wind boundary layer exponent for the meteorological station
- a Wind boundary layer exponent for the local building terrain

Table 1 in Chapter 16 of ASHRAE 2005 contains values for the above parameters. ContamW calculates C_h according to equation 1 above and provides for the input of the *local terrain constant*, A_o , but not δ . Therefore, in order to implement the updated method, you must adjust the value of A_o according to the following equation:

$$A_o = \left(\frac{\delta_{met}}{H_{met}} \right)^{a_{met}} \left(\frac{H_{met}}{\delta} \right)^a$$

3.17.1 Defining Weather and Wind

You define weather that you want CONTAM to use based on the type of simulation you want to perform. You will either define steady state weather or transient weather. It is best to perform a set of steady state simulations prior to performing transient simulations. This may enable you to better identify any problems that may arise from working with transient weather. ContamW also provides a means to verify that you have your wind related data input correctly (See Checking Wind).

3.17.1.1 Defining Steady State Weather and Wind

To define the steady state weather and wind data you use the **Edit Weather Data...** selection of the **Weather** menu or the **F7** keyboard shortcut. This will display the "Weather and Wind Parameters" property sheet. The property sheet contains four different pages of parameters including: Weather, Wind, Location and Wind Pressure Display. Detailed descriptions of all wind and weather properties are given in the Weather and Wind Properties section of this manual.

You use the Weather page to define steady state weather and wind data. The Location page allows you to input the altitude of the building site which ContamW will use to determine a default barometric pressure if you do not know what it should be for a given building site. Default values of these two pages will provide conditions of no wind, 20 °C and a barometric pressure of 1 atmosphere (approximately 101 kPa).

The Wind page parameters allow you to enter the prevailing wind direction and other values used by ContamW to determine a wind speed modifier. The wind speed modifier is determined from the local terrain constant and the velocity profile exponent. Default values are provided for suburban terrain.

Checking Wind Pressure Data

ContamW provides a feature that you can use to verify the wind information visually on the SketchPad. To verify wind pressure information on the SketchPad, use the **Wind Pressure** selection of the **View** menu. The wind pressures that are displayed in this view are not necessarily the same as will be displayed when viewing simulation results on the SketchPad. There are two modes of this display: one based on the information set via the Wind Pressure Display property page and one when using a WPC file. If not using a WPC file then wind pressure display is calculated based on wind speed and direction established by the Wind Pressure Display Properties. If using a WPC file then wind pressure display is calculated based on the first set of path pressure data contained in the WPC file (see Working with WPC Files).

When you choose to display wind pressure results, colored lines (Pres Results color) will appear on the SketchPad indicating the wind pressure at certain airflow paths. Wind pressure will only be displayed for those airflow paths and duct terminals connected to the ambient zone, having the Wind Pressure Option set to Variable or a pressure specified in the WPC file if using a WPC file. The wind pressure display will reveal the relative magnitude of the pressure drop and the direction of airflow across the openings in the building envelope. You can move the caret to each of the airflow paths for which a wind pressure is being displayed to show the wind pressure, azimuth angle and elevation of the airflow path in the ContamW status bar. When reviewing this display, you should check if the direction of flow makes sense based on the information that you entered within the Wind Pressure Display property page or contained in the first record of the WPC file.

3.17.1.2 Defining Transient Weather and Wind

In order to implement transient weather and wind, you must create a CONTAM-compatible weather file. You must create the weather file external to CONTAM. NIST provides a utility program, WEATHER 2.0, to create CONTAM-compatible weather files.

WEATHER 2.0 enables you to convert existing CONTAM 1.0 weather files to the 2.0 format weather and ambient contaminant files. You can also convert TMY2 and EnergyPlus (.EPW) weather files to the CONTAM 2.0 weather file format and create weather file templates into which you can place your own data (e.g., experimental data). Both weather and ambient contaminant files are ASCII text files that can be created and edited using spreadsheet software according to the formats provided within this manual.

□ Weather File Format

The CONTAM weather file is a tab-delimited ASCII text file. These files are easily created, imported and modified using typical spreadsheet applications. They can be saved as tab-delimited text files for use with CONTAM. It is described below because it may be necessary for the you to create a weather file from sources not already provided. In the listing below I1 indicates a character string, I2 indicates a short integer and R4 indicates a four-byte real number. Individual lines of data are separated by dashes (these dashes are only shown here for format presentation purposes and should not be included in the actual files). Each day must begin with time 00:00:00 and end with time 24:00:00. The time interval between 00:00:00 and 24:00:00 can be either regular or irregular – ContamX will interpolate as necessary. Comments are allowed and are indicated by an "!" (exclamation point). Anything that appears on a line following a comment indicator is ignored by the weather file reader. Comments are provided by the weather file creation software (Weather 2.0) that show header information for the data in the file.

```
WeatherFile ContamW 2.0  ! file and version identification
-----
I1 descr[]           ! weather file description
-----
I1 start_date       ! day of year (1/1 - 12/31)
I1 end_date         ! day of year (1/1 - 12/31)
-----
!Date  DofW  Dtype  DST  Tgrnd
-----
! for each day: start_date - end_date
  I1 date          ! date (1/1 - 12/31)
  I2 dayofwk       ! day of week Sun - Sat (1 - 7)
  I2 daytype       ! type of day for schedule reference (1 - 12)
  I2 DST           ! daylight savings time indicator (0 or 1)
  R4 Tground       ! ground temperature [K]
-----
!Date Time Ta  Pb  Ws  Wd  Hr  Ith  Idn  Ts  Rn  Sn
-----
! for each day: start_date - end_date
! for each time: 00:00:00 - 24:00:00 increment may be variable
  I1 date          ! date (1/1 - 12/31)
  I1 time          ! time- of- day (00:00:00 - 24:00:00)
```

```

R4 tmpambt ! ambient temperature [K]
R4 barpres ! barom. pressure(Pa) NOT corrected to sea level
R4 windspd ! wind speed [m/s]
R4 winddir ! wind direction [deg]: 0=N, 90=E, 180=S, ...
R4 humratio ! humidity ratio [g H2O/kg dry-air]
R4 solhtot ! total solar flux on horizontal surface [W/m^2]
R4 solhdif ! diffuse solar flux on horizontal surface [W/m^2]
R4 tskyeff ! effective sky temperature [K]
I2 rain ! rain indicator 0 or 1
I2 snow ! snow indicator 0 or 1

```

The Tground, solar, sky, rain and snow values are not used in CONTAM. They are reserved for thermal analysis, if it is added to the program.

□ Editing Weather Files

You can import weather files into a spreadsheet program for editing. However you must make sure that the date format is compatible with CONTAM when you save the file. One well-known spreadsheet program will convert the CONTAM date format of mm/dd into the mmm-dd form. For example the CONTAM date format for January 1 is "1/1", but the spreadsheet program will convert it to "Jan-01" unless you "force" it to import the first column of data – the one that contains date information – as text. Another option is to accept the default importing option then change the cell format for each date to the CONTAM format mm/dd prior to saving the file as tab-delimited.

3.17.1.3 Weather and Wind Properties

This section provides detailed descriptions of the specific steady state weather properties. Each of the following subsections are the context-sensitive help topics that you can access by pressing **F1** when working with property pages of the "Weather and Wind Properties" property sheet.

Weather and Wind – Wind Properties

The wall height, terrain and exponent are used to calculate the wind speed modifier.

Relative North: The direction to relative north will be used relating wall angles on the SketchPad to true north for determining wind direction effects. Enter the clockwise angle from vertical on the ContamW SketchPad to true north.

Roof or Wall Height: Enter the height of the upwind wall of the building. This is the value of H in the equations presented in the section Working with Weather and Wind. Entering a value of zero will reduce the wind speed modifier to zero. This will cause computed wind pressures to be zero no matter what the wind speed, unless you override the wind pressure coefficients of exterior airflow paths. Figure 4 in Chapter 16 of the 2005 ASHRAE Fundamentals Handbook [ASHRAE 2005] shows relative wind pressures on the face of a building based on the local wind pressure at the top of the wall.

Terrain & Exponent: Enter the values that describe the wind speed profile for the type of region in which your building is located. Figure 4 in Chapter 14 of the 1993 ASHRAE Fundamentals Handbook [ASHRAE 1993] shows terrain constants and velocity profile exponents for representative areas.

An updated method, from that presented in ASHRAE 1993, of accounting for local terrain

effects is presented Chapter 16 of ASHRAE 2005. Refer to Working with Weather and Wind for details on how to implement this method within CONTAM.

Modifier: This is the wind speed modifier, Ch , calculated by ContamW based on the Wall Height, Terrain and Exponent values above. This value will be the default provided for each envelope airflow path that you create on the SketchPad and for which you select the variable wind pressure option. See section Working with Weather and Wind for a more detailed explanation of this term.

Weather and Wind – Weather Properties

The following constant values are used by CONTAM when you specify the use of steady state weather conditions when performing simulations (See Working with Simulations).

Temperature: ContamX uses the ambient temperature when calculating outside air density and building stack effects.

Absolute Pressure: Absolute barometric pressure (not corrected to sea level). If you don't know this value, you can enter the altitude of the building site on the Location Properties page to have ContamW calculate the barometric pressure for you.

Relative Humidity: Relative humidity as a fraction, i.e., between 0.0 and 1.0. This value will be used along with Ambient Temperature and Pressure to calculate the Humidity Ratio and Mass Fraction of H₂O.

Humidity Ratio: The mass of water vapor per mass of air (excluding water vapor) calculated based on the Temperature, Pressure and Relative Humidity.

Humidity Ratio = Mass Fraction / (1 – Mass Fraction)

Mass Fraction (H₂O): The mass of water vapor per mass of air (including water vapor). These are the default units of contaminants in CONTAM. This value is provided so you can use it as the default contaminant concentration of H₂O.

Wind Speed: Magnitude of the wind velocity.

Wind Direction: Direction from which the wind blows as measured in degrees clockwise from true north.

Day Type: This is the type of day you want to use when performing a simulation with steady airflows and no weather file. The Day Type corresponds to one of the twelve day schedules that you can create.

Weather and Wind – Location Properties

Longitude, latitude, and time zone are currently not used by CONTAM. They have been included for potential use in the future if heat transfer analysis is implemented within the program.

Altitude: ContamW uses the site altitude to determine a default absolute barometric pressure for the steady state weather data.

Weather and Wind – Wind Pressure Display Properties

The wind pressure display values are used to calculate the wind pressures displayed on the screen in wind pressure display mode. You activate the wind pressure display mode by selecting **Wind Pressure** from the **View** menu.

NOTE: These values are not used by ContamX when performing a simulation. To display the wind pressure based on these values use the **Wind Pressure** selection of the **View** menu. This feature is provided so that you can verify the wind speed and direction information visually on the SketchPad. (See Checking Wind Pressure Data in the Defining Steady State Weather and Wind section)

Ambient Temperature: The ambient temperature is used to calculate outside air density and building stack effects.

Absolute Pressure: Be sure to enter the absolute barometric pressure instead of pressure corrected to sea level.

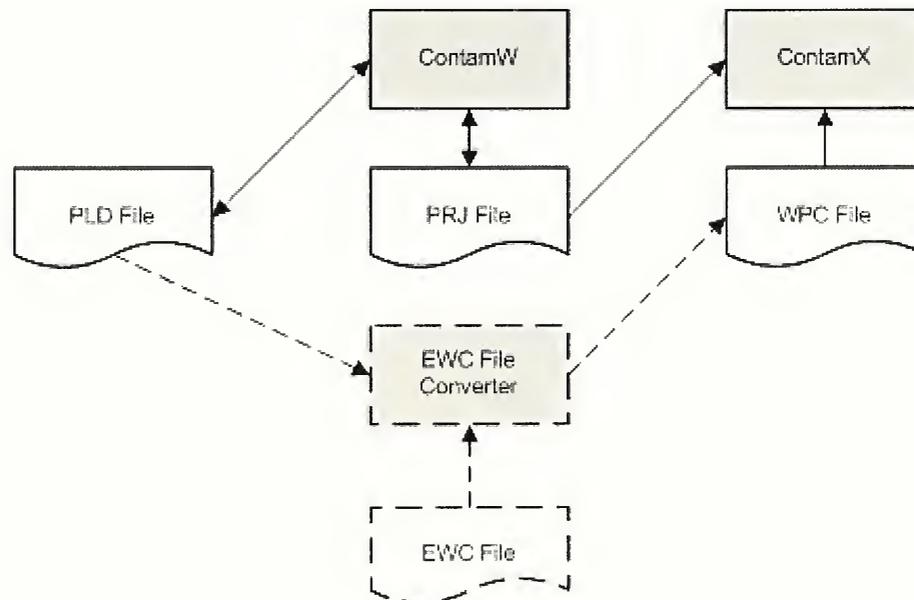
Wind Speed/Direction: Enter the wind speed for the site. The wind direction is calculated in degrees clockwise from true north.

Day Type: This is the day type as defined in the Week Schedule .

3.18 Working with WPC Files

A new method to account for the variation of external wind pressures and outdoor contaminant concentrations over the building envelope has been implemented in CONTAM. This method addresses the need to allow for the use of general, spatially varying wind pressure and ambient contaminant concentrations such as those from wind tunnel experiments or atmospheric models, e.g., plume or puff dispersion simulation tools. This method involves the implementation of a Wind Pressure and Contaminants file (WPC file). This file provides exterior pressure and/or contaminant concentrations time histories for every flow path that connects to the ambient zone including duct terminals and outdoor air intakes of CONTAM’s simple air handling systems.

The WPC files are created externally to CONTAM, however their creation can be assisted by ContamW that creates a Path Location Data (PLD) file listing all the airflow path locations that are connected to the ambient zone. The following figure illustrates the interaction between CONTAM and the WPC file. The dashed lines in the figure represent optional components. The WPC file is an ASCII file that could be created using conventional means, e.g., spreadsheet converted to text. One could also develop a WPC File Converter program to create the files from External Wind Pressure and Contaminant data files created by a separate tool, e.g., exterior CFD program. A converter could then work with the PLD file to create a WPC file specific to the building in question, i.e., the PRJ file. ContamW provides a means to activate a user-selectable converter. The details of the WPC and PLD file formats are presented in the sections titled WPC File Format, and PLD File Format respectively. CONTAM does not include either an EWC file creation tool or an EWC-to-WPC converter tool.



WPC File Implementation Schematic

Steps to implement WPC files:

- Specify WPC file usage parameters (Weather→Use WPC File...)
 - o Select type of data the WPC file includes – wind pressures and/or contaminants
 - o Select existing WPC file
- Or
- Select *EWC File Converter* program (optional and user-provided)
 - o Specify converter parameters: equivalent origin, location tolerance, date and time info
 - o Select EWC input file to EWC File Converter program
 - o Specify name of WPC file to create
- Enter *Coordinate Information* for each external flow path
 - o Airflow paths
 - o Duct terminals
 - o Outdoor air intakes of simple air handling systems
- From the Simulation menu select Run Simulation or Create ContamX Input File for batch processing. Either option will activate the WPC file implementation routines (as needed) that will create a PLD file, call the converter program (if one has been selected), compare WPC and PLD files and notify user of discrepancies between them (if a converter program has been selected).

3.18.1 WPC Usage Parameters

The following parameters are used in specifying the usage of a WPC file when performing a simulation and/or using an *EWC File Converter* to create a WPC file. Access these parameters via the **Weather→WPC File...** menu item. The values in the group labeled *WPC File* are required, while the others are only necessary if implementing an *EWC File Converter* program. For an overview of WPC files see Working with Wind Pressure and Contaminant (WPC) Files.

NOTE: WPC files can be used during a simulation in conjunction with the transient weather and transient contaminant files as follows:

- If using wind pressures from a WPC file, you may not use a WTH file.
- If using contaminant concentrations from a WPC file, you may not use a CTM file.

WPC File

Use this group of data to specify the name of a WPC file to create and/or the WPC file to use when performing a simulation and the type of data contained within the file.

Wind Pressures: Check this box to use wind pressures from a WPC file and/or to have a converter create a WPC file that contains wind pressure data.

If this box is checked during a simulation then the wind pressures in the WPC file take precedence over other wind pressure settings of flow paths, junctions, etc.

Contaminant Concentrations: Check this box to use contaminant concentrations from a WPC File and/or to have a converter create a WPC file that contains contaminant data. This box will only be activated if you have already defined contaminants within the current project.

If this box is checked during a simulation then the contaminant concentrations in the WPC

file take precedence over other transient contaminant settings. For those contaminants that are defined within the PRJ file but not included in the WPC file, the steady-state contaminant data will be used, i.e., the default concentrations.

Name: This is the name of the WPC file to create and/or use during the simulation. You can either use the *Browse...* button to select an existing file, or enter the name you would like the *EWC File Converter* to give to a new file.

Description: This will display the description line of the WPC file if it exists, or you can enter a description you want an *EWC File Converter* to write when creating a new WPC file.

External Wind and Contaminant Data

The *External Wind Pressure and Contaminant (EWC)* file contains the external contaminant concentrations and pressures that the *EWC File Converter* (which must be developed for your specific application) will convert into a WPC file.

File Name: Use the *Browse...* button to select an existing EWC file.

Program to Create WPC File: Use the *Browse...* button to select a program to use to create a WPC File for example an *EWC File Converter*.

Coordinate Transformation Data

These parameters can be used by an *EWC File Converter* to establish the relationship between the coordinates of the EWC file and the CONTAM project file. No transformation is required if the coordinate systems for the PRJ and EWC files are consistent. These values are stored in the PLD file by ContamW.

Origin (X, Y and Z): The location of the origin of the CONTAM PRJ file with respect to the origin of the EWC file.

Rotation Data: The rotation of the x and y axes of the EWC coordinate system about the z axis to align with the x and y axes of the CONTAM coordinate system. Counter-clockwise is considered the positive direction.

Conversion Tolerance

These are the tolerances that an *EWC File Converter* might use to determine how closely information in the EWC file and the PLD file must match in order to resolve contaminant (species) and location data. These values are stored in the PLD file by ContamW.

Species: Species can be resolved by their molar mass, i.e., the molar mass of each species defined in ContamW and the WPC file must not differ by more than this amount to be considered the same species.

Location: Locations can be resolved by their distance as determined by an *EWC File Converter*, e.g., distance to center of a grid cell. Units within the PLD file are in meters.

Date and Time

These parameters can be used by an *EWC File Converter* to determine the date and time values written to the WPC file for the convenience of working according to CONTAM's schedules. These values are stored in the PLD file by ContamW.

Data Time Shift: The format for this value is hh:mm:ss. An *EWC File Converter* could add this value to the EWC file time. For example an *EWC File Converter* could create a file that

begins at 00:00:00 and set the next time to be the *Data Time Shift* you enter here. The data for these two times would be the initial values in the EWC file.

Start and End Date: The format for this value is mmmdd. For example, enter January 1 as Jan01. An *EWC File Converter* could use this to output the date(s) to use when running transient simulations.

3.18.2 Envelope Opening Locations

When using *WPC Files* you must enter the coordinates for each opening connected to the ambient. ContamW now provides a means to do this for *Airflow Paths*, *Duct Terminals*, and *Outdoor Air Intakes of Simple Air Handling Systems*. This information is provided on the *Wind Pressure Property Pages* of the *Airflow Path Properties* and *Duct Junction Properties* and on the *AHS Property Page* of the *Simple Air Handling System Properties* input dialog boxes. The location information will be used to create the *PLD File* which in turn is used to verify there is matching information within the *WPC File* prior to performing simulations using the *WPC File*.

NOTE: Even though every opening connected to ambient requires location data when simulating with *WPC Files*, they do not necessarily have to be unique. They simply must match a location within the *WPC File*.

3.18.3 Airflow Paths and Duct Terminals

Enter the X and Y coordinates and units for the selected airflow path or duct terminal. The Z coordinate will be calculated by ContamW to be the *Relative Elevation* of the airflow path or duct terminal plus the *Elevation* of the building level on which the path or terminal is located.

3.18.4 Outdoor Air Intakes of Simple Air Handling Systems

Enter the X, Y and Z coordinates and units for the outdoor air intake of the selected *Simple AHS*. Enter the Z coordinate as the height of the midpoint of the outdoor air intake with respect to the building reference height.

3.18.5 Running Simulations using a WPC File

To run a simulation using a WPC file, you must have checked either the *Wind Pressures* or *Contaminant Concentrations* check box on the *Wind Pressure and Contaminant (WPC) File Parameters* dialog box. When you choose **Run Simulation** from the **Simulation** menu, ContamW will create a *Path Location Data* file (<project file name>.pld) perform a series of checks to make sure that 1) all the path locations have been defined and match those in the WPC file if it exists, and 2) the species match between the PRJ and the WPC files. It will call the *EWC File Converter* and create a WPC file if an EWC file converter and EWC file have been identified.

If a converter is identified, ContamW will provide error messages as needed and highlight paths on the SketchPad to reveal those for which location data are not defined or the PLD and WPC coordinates do not match within the specified tolerance. If there are errors with specific openings, you can take this opportunity to correct them.

Another option is available to delay simulation for example if you wish to perform batch simulation runs using ContamX. That option is the **Create ContamX Input File** selection of the **Simulation** menu. If you select this option, ContamW will still create the PLD file, but will not automatically perform the checks to verify that the PLD and WPC files match unless you have

selected an *EWC File Converter*. You can also have ContamW perform the checks by selecting the **Wind Pressure** option from the **View** menu (see Checking Wind in the Defining Weather and Wind section).

If there are errors between the WPC file and the PLD file, then ContamX will not be able to run a simulation. If there are any flow paths that do not map to a location in the WPC file, a list of these locations will be written to the CONTAMW2.LOG file. Look for list of Paths/Terminals/Junctions not found in WPC file. ContamW will not initiate ContamX. If ContamX is run from the command line on a project having discrepancies, it will also exit with an error message.

3.18.6 WPC File Format

This file provides ambient pressure and contaminant concentrations values for every flow path that connects to ambient.

Pressures for each flow path (P_{path}) must be input as absolute pressures in units of Pa. Wind pressure will be computed from the ambient pressure (P_{ambt}) and density (ρ_{ambt}) provided at each time interval in the WPC file and the absolute pressure (P_{path}) and elevation (Z_{path}) of each flow path as follows:

$$P_{\text{wind}} = P_{\text{path}} - P_{\text{ambt}} + \rho_{\text{ambt}} \cdot g \cdot Z_{\text{path}}$$

The ambient pressure (P_{ambt}) and density (ρ_{ambt}) specified at each time step are akin to the absolute pressure and temperature values specified in the Steady state weather data.

Contaminant concentrations must be provided in units of kg of contaminant per kg of air. See Concentration Conversions in the Contaminant Files subsection of the Working with Species and Contaminants section for conversion factors.

The first line of the WPC file is used to identify the type of file. It is exactly:

```
WPCFile ContamW 2.1
```

The second line is a description of the file entered by the user:

```
desc[]          // file description (I1) {W}; may be blank
```

Succeeding lines contain:

```
NumPath        // number of flow paths (I2)
NumCont        // number of contaminants (species) (I2) [0 possible]
UsePres        // 1 = use pressures, 0 = don't (I2)
dtmin          // minimum time step (I2) (Not used by ContamX)
StartDate      // first date for WPC data (mm/dd → IX)
EndDate        // last date for WPC data (mm/dd → IX)
I1 Name[1]     // name of contaminant 1 (I1 - 16 characters max)
I1 Name[2]     // name of contaminant 2 (I1)
...
Name[NumCont]  // name of contaminant NumCont (I1)
```

The next NumPath lines describe each flow path:

```
    number      // sequence number (I2)
    X           // X-coordinate (R4)
```

```

Y          // X-coordinate (R4)
Z          // Z-coordinate (R4)
map        // mapping [0 = OK; 1 = error] (I2)

```

Data from 00:00:00 of StartDate to 24:00:00 of EndDate:

At each time:

First line:

```

date       // date (mm/dd → IX)
time       // time of day (hh:mm:ss → I4)
pres       // ambient pressure (Pambt) [Pa] (R4) (if pressures are used)
dens       // air density (ρambt) [kg/m3] (R4) (if pressures are used)

```

Second line: (if pressures are used)

```

pres[1]    // absolute ambient pressure at path[1] (Ppath) [Pa] (R4)
...
pres[NumPath] // absolute ambient pressure at path[NumPath] [Pa] (R4)

```

Third line (for first contaminant, if NumCont > 1):

```

conc[1]    // concentration at path[1] [kg/kg air] (R4)
...
conc[NumPath] // concentration at path[NumPath] [kg/kg air] (R4)

```

Fourth line (for second contaminant, if needed):

...

Last line (for contaminant NumCont, if needed):

```

conc[1]    // concentration at path[1] [kg/kg air] (R4)
...
conc[NumPath] // concentration at path[NumPath] [kg/kg air] (R4)

```

The file description may not begin with a '!'. The StartDate and EndDate are used to verify that the file data covers the entire period to be simulated. The StartDate may not be later than the EndDate.

The data must start at time 00:00:00 on the StartDate and end at 24:00:00 on the EndDate. The times must be in consecutive order, but the difference between successive times need not be constant.

Example:

```

WPCfile ContamW 2.1
For WPCtest3.prj
2      ! flowpaths
1      ! contaminants
1      ! use pressure flag
0      ! time step
01/01 ! start date
01/01 ! end date
C1
!nr   X      Y      Z      map
1     0.000 4.000 1.500 0

```

```

2      8.000 4.000 1.500 0
01/01 00:00:00    101325    1.204
101308.30    101306.30
0.0    1.0e-6
01/01 24:00:00 101325 1.2041
101308.30    101306.30
0.0    1.5e-6
    
```

3.18.7 PLD File Format

The PLD (Path Location Data) File will consist of the locations of any flow paths that connect the building to the external environment, as well as additional information for directing the behavior of the EWC File Converter. This file is created by ContamW and stored in the same directory as the project file having the same name as the project file but with the .pld extension replacing the .prj extension. It is created as needed for use in comparing WPC file information with that in the CONTAM project file. (See Working with WPC Files)

The file is in an ASCII line-delimited format.

The reference location, defined by (Xref,Yref,Zref), represents the EWC file location that is equivalent to the origin of the CONTAM coordinate system for the path locations. The user must coordinate between the EWC file and PRJ file to determine this location’s coordinate values.

The rotation angle is about the Z (vertical) axis. The user must coordinate the direction of rotation between EWC and PRJ file coordinates. It is assumed there are no rotations about the X or Y axes.

The six coordinates on line 6 define a bounding box that surrounds the flow paths within this file, which could be used to assist the EWC File Converter in rapidly excluding unneeded points (e.g., that are too distant from the building) during the mapping process. Note that this bounding box does not necessarily surround the entire building.

The simulation time step provides a recommendation to the EWC File Converter for reducing the size of the WPC file, so that only the time steps that coincide with the simulation time step could be included (which could reduce interpolation required).

Each flow path is uniquely identified by a combination of flow path type and flow path ID values.

Comments begin with an exclamation point (!) and may begin at the start of any line (so that the entire line will be ignored) or after all fields of a line (so that the remainder of the line will be ignored).

For user readability, fields should be commented whenever possible.

The Precision column for the real data types may be interpreted similar to a C scanf() statement’s conversion specification.

HEADER SECTION:

Line	Field	Data Type	Precision	Notes
1	EWC filename	character		filename includes full path
2	WPC filename	character		filename includes full path
3	file comments	character		user-defined notes about this file

4	coordinate headings	character		one line of comments for the next section; "!Xref Yref Zref angle"
5	Xref	real	%7.3f	units of meters
5	Yref	real	%7.3f	units of meters
5	Zref	real	%7.3f	units of meters
5	rotation angle	real	%8.3f	rotation about Z axis in degrees; positive for CCW direction
6	latitude	real	%8.4f	units of degrees; sign is positive for north
6	longitude	real	%8.4f	units of degrees; sign is positive for east
7	bounding box headings	character		one line of comments for the next section; "!Xmin Xmax Ymin Ymax Zmin Zmax"
8	Xmin	real	%7.3f	units of meters
8	Xmax	real	%7.3f	units of meters
8	Ymin	real	%7.3f	units of meters
8	Ymax	real	%7.3f	units of meters
8	Zmin	real	%7.3f	units of meters
8	Zmax	real	%7.3f	units of meters
9	time step headings	character		one line of comments for the next section; "!step shift start end"
10	time step	integer		units of seconds
10	data shift	time		format hh:mm:ss; this is the starting time for EWC time steps when converted to WPC
10	start date	date		format of mm/dd
10	end date	date		format of mm/dd
11	wind pressures	integer		1: include pressures in WPC; 0: don't
11	number of species	integer		
11	species map tolerance	real	%7.3f	for use by EWC file converter file to match species between EWC and PLD files by molar mass.

CONTAMINANT DEFINITION SECTION:

12	species headings	character		one line of comments for the next section; "!name m.wt"
----	------------------	-----------	--	--

The next (number of species) lines contain this data:

for each species...

name	character		
molar mass	real	%6.2f	

end for each species

FLOWPATH DEFINITION SECTION:

-	number of flow paths	integer		
-	path map tolerance	real	%7.3f	for use by EWC file converter to match flow paths that are more than this distance away; units of meters; this field is on the same line as the above field
	flow path headings	character		this is one line of comments for the next section; "!type ID X Y Z"

The next (number of flow paths) lines contain this data:

for each flow path...

flow path ID number	integer		
x	real	%7.3f	units of meters
y	real	%7.3f	units of meters
z	real	%7.3f	units of meters

end for each flow path

last line marker -999	integer		
-----------------------	---------	--	--

Example:

```

C:\Program Files\Contamw2\Prjs\WPCcube.ewc ! EWC file
C:\Program Files\Contamw2\Prjs\WPCcubel.wpc ! WPC file
WPC description
! Xref      Xref      Yref      angle
  0.000    0.000    0.000    0.00
  0.0000   0.0000   ! no latitude/longitude
! Xmin      Xmax      Ymin      Ymax      Zmin      Zmax
  0.000    5.000    2.500    10.000    1.000    1.000
! step      shift      start      end
  0        00:00:00  1/1        1/1
  1        2          0.01 ! pressures flag, # species, mapping tolerance
! name      m.wt
  CO       28.00
  CO2      44.00
  4        1.00 ! number of flow paths and mapping tolerance
! id#       X          Y          Z
  1        5.000    10.000    1.000
  2        0.000    7.500    1.000
  3        0.000    5.000    1.000
  4        0.000    2.500    1.000
-999
    
```

3.19 Working with Schedules

Schedules provide you with the ability to modify various parameters based on the time of day and day of the week. Schedules apply when performing both transient and steady-state simulations. The steady-state simulation date and time will be used to access the schedule values. You use schedules to provide a fractional multiplier to parameters at times which you define. You can schedule items such as the flow through airflow paths, the outdoor air intake of simple air handling systems, contaminant source/sinks, the location of an occupant within a building and zone temperatures.

CONTAM provides occupant schedules and non-occupant schedules which are split up into week schedules which are in turn made up of one or more day schedules. Non-occupant week schedules can be associated with as many schedulable items as you need, whereas occupant week schedules can only be applied to a single occupant. Only non-occupant schedules can be shared between CONTAM project files using the CONTAMW Library Manager. Occupant schedules can not be shared between projects, because they are dependent on the level and zone names which may be different between projects. The following sections provide detailed information related to the creation, modification and deletion of schedules in ContamW.

NOTES on using Schedules:

The schedule values used during simulation are dependent on the type of simulation and the source of weather file data during simulation. See weather data related parameters of the Run Control Properties in the Working with Simulations section.

If a schedule and a control signal (see Working with Controls) are both associated with the same building element icon, then the control signal will take precedence over the schedule during simulation.

3.19.1 Creating Schedules

You can create schedules using either the CONTAMW Library Manager (see Working with Data and Libraries) or wherever a "New Schedule" button is provided when working with various building component property sheets (e.g. airflow paths, ducts, source/sinks, and occupant icons).

Non-occupancy Schedules

You create non-occupant schedules by first selecting to create a new week schedule as outlined above which will display the Week Schedule dialog box. You must then define one or more day schedules that you then associate with each day of the week schedule. Each week schedule consists of 12 days – one for each day of the week and five more days that you can use for special situations such as holidays. New day schedules are created from the Week Schedule dialog box by clicking on the "New Schedule" button. This will display the Day Schedule dialog box whose features are explained below.

Temperature Schedules

Temperature schedules are a type of non-occupant schedule. Unlike other non-occupant schedules they have units associated with them. Temperature schedules are only associated with zones and are considered separately from other non-occupant schedules when creating and editing. For instance, when displaying a list of temperature schedules, only those non-occupancy schedules having temperature units will be displayed and not those that are unitless.

❑ Occupancy Schedules

You create occupant schedules by first creating an occupant exposure icon (see Creating Occupants). You then click on the "Edit Occupancy Schedule" button that is provided on the Occupant Data property page of the "Occupant Exposure Properties" property sheet.

3.19.2 Modifying Schedules

You modify existing non-occupancy week and day schedules through the Week Schedule and Day Schedule dialog boxes respectively. Edit occupancy schedules by editing the properties of existing occupant exposure icons. The details of editing schedules are presented in the following sections.

3.19.3 Deleting Schedules

You can only delete occupancy week schedules by deleting the associated occupant exposure icon. Deleting all other schedules can only be accomplished using the CONTAMW Library Manager (see Working with Data and Libraries) which will only allow you to delete those schedules that are no longer associated with any occupants, zones or other building components.

3.19.4 Week Schedule Properties

These are the properties and methods you use to define non-occupancy week schedules. You must associate a day schedule with each day of a week schedule (7 days of the week plus 5 extra days). The extra days are provided for any day that may have a non typical schedule, such as a holiday. Once defined, you may display the schedule graphically by pressing the "Display Graphically" button.

Name: This is the name you give to this week schedule. This name must be unique among the week schedules within this project.

Units: Select the temperature units to use when defining a zone temperature schedule. This control will only be available if these properties are accessed from a zone property sheet, Zone Data or via the **Data** → **Week Schedules** → **Temperature...** menu selection.

Description: Field for entering a more detailed description of the specific schedule.

Day Schedules: Every "Week Schedule" consists of 12 days that must be defined. The days are defined by "Day Schedules". These day schedules may or may not be unique for each of the 12 days. If the project has no available day schedules a new day schedule must be defined. Do this by pressing the "New Day Schedule" button, then editing the day schedule as explained in the section entitled "Day Schedule." You can edit existing day schedules by selecting them in the "Available Schedules" list box and then pressing the "Edit Day Schedule" button, then following the procedure explained in the Day Schedule section. If a day has already been associated with a day schedule you can change it by first highlighting the day to be changed in the Day Schedules list box, highlighting the day schedule to replace the existing schedule, and then pressing the "<< Replace <<" button.

Only those day schedules of the same type as the week schedule will appear in the list of day schedules, i.e., dimensionless or temperature.

3.19.5 Day Schedule Properties

These are the properties and methods you use to define day schedules. You may display the schedule graphically by pressing the "Display Graphically" button.

Name: This is the name you give to this day schedule. This name must be unique among the day schedules within this project.

Description: Field for entering a more detailed description of the specific schedule.

Shape: Use this check box to set how CONTAM interprets values between the data points you define. A rectangular graph will create a step function for the data points. A trapezoidal graph will generate a sloped line between the data points provided by interpolating a straight line between the data points.

Schedule Data: These are the time/multiplier pairs that you set to define a day schedule. The times are entered in the hh:mm:ss format, and you must provide leading and trailing zeros; for example 9:30 am would be entered as 09:30:00 not as 9:30:00.

For dimensionless schedules, the value is a multiplier between 0.0 and 1.0 used to adjust the entered value for the simulation feature at the associated time defined in the schedule otherwise enter the desired temperature and units for temperature schedules.

To insert a new data point:

Enter the new time/multiplier pair in the "Time" and "Value" edit box next to the insert button, press the "<< Insert <<" button and the data point will be inserted into the schedule in the correct sequence.

To modify an existing data point:

Select the data point that you wish to modify from the list, enter the revised data, and press the "<< Insert <<" button to overwrite the previously entered data point.

To delete an existing data point:

Select the data point you want to delete from the list and click the "Delete" button. You can not delete 00:00:00 or 24:00:00.

3.19.6 Occupant Week Schedule Properties

These are the properties and methods you use to define occupant week schedules.

Description: Field for entering a more detailed description of the specific schedule.

Day Schedules: Every "Occupancy Week Schedule" consists of 12 days that must be defined. The days are defined by Occupant Day Schedules. These schedules may or may not be unique for each of the 12 days. If the project has no available day schedules a new day schedule must be defined. This is done by pressing the "New Day Schedule" button then editing the day schedule as explained in the Occupant Day Schedule section. You may edit existing day schedules by selecting them in the "Available Schedules" list box and then pressing the "Edit Schedule" button, then following the procedure explained in the Occupant Day Schedule section. If a day has already been associated with a day schedule, you may change it by first highlighting the day to be changed in the Day Schedules list box, then highlighting the day schedule to replace the existing schedule, and finally pressing the "<< Replace <<" button.

3.19.7 Occupant Day Schedule Properties

These are the properties and methods you use to define occupant day schedules.

Name: This is the name you give to this occupant day schedule. This name must be unique among the occupant day schedules within this project.

Description: Field for entering a more detailed description of the specific schedule.

Schedule Data: These are the time/location data points that you set to define an occupancy day schedule. The times are entered in the hh:mm:ss format, and you must provide leading and trailing zero's; for example 9:30 am would be entered as 09:30:00 not as 9:30:00. The location consists of a level and zone in which the occupant resides for the indicated time period. Select a value of "null" from the zone list to indicate the occupant is not in the building for a given time period.

Location data will also include zone coordinates if the occupant enters a 1D convection diffusion zone. For such cases, enter the X, Y and Relative Elevation (Z) coordinates and units for each 1D zone the occupant enters. X and Y are absolute coordinates and Rel Elevation is relative to the level of each 1D zone that the occupant enters.

To insert a new data point:

Enter the new time and location data in the "Modify Data" block, press the "<< Insert <<" button and the data point will be inserted into the schedule in the correct sequence.

To modify an existing data point:

Select the data point that you wish to modify from the list, enter the revised data, and press the "<< Insert <<" button to overwrite the previously entered data point.

To delete an existing data point:

Select the data point you want to delete from the list and click the "Delete" button. You can not delete 00:00:00 or 24:00:00.

3.20 Working with Simulations

CONTAM allows you to perform several types of simulations. The type of simulation you wish to perform depends on the goal of your analysis. You can perform simulations to determine just interzonal airflows and pressure differences, or you can also have CONTAM solve for contaminant concentrations within each zone of a building. CONTAM provides the ability to perform several combinations of steady state, transient and cyclical simulations of both airflow and contaminant dispersal.

The following table shows the different combination of simulation methods you can apply using CONTAM.

Airflow Simulation	Contaminant Simulation	Purpose
Steady State	None	Used to obtain airflows and pressure differentials under constant building system and/or weather conditions.
Steady State	Steady State	Used to obtain equilibrium contaminant concentrations under conditions of steady airflow.
Steady State	Transient	Used to obtain a time history of contaminant concentrations under conditions of steady airflow. This is similar to the previous table entry but will allow you to observe the contaminant concentrations for each time step leading up to the final time step.
Steady State	Cyclic	Used when it's important for initial contaminant concentrations to match the final concentrations under steady airflow conditions. This provides a method of simulating a typical pattern of operation by simulating only a representative time period (e.g. one day).
Transient	None	Used to obtain a time history of airflows and pressure differentials under changing building system and/or weather conditions.
Transient	Steady State	Not allowed
Transient	Transient	Used to obtain time histories of airflows, pressure differentials and contaminant concentrations under changing building system and/or weather conditions.
Transient	Cyclic	Used when it's important for initial contaminant concentrations to match the final concentrations under transient airflow conditions. This provides a method of simulating a typical pattern of operation by simulating only a representative time period (e.g. one day).
Steady State OR Transient	Short Time Step Method (Transient OR Cyclic)	The short time step contaminant solution method is available beginning with version 2.4. This is an explicit solution method useful when interested in rapidly changing conditions that require short time steps to capture. It is required when using the 1D zone and duct models.
Duct Balance	n/a	Use this feature to have ContamX adjust duct terminal balancing coefficients to provide user-defined design airflow rates at each balancing terminal. This does not perform a simulation per-say; it balances the system for upcoming simulations.
Building Airflow Test	n/a	Generates a set of data, mostly related to building ventilation, that enables you to review the reasonableness of your building model and the input values.
Pressurization Test	n/a	Performs an automated, idealized fan pressurization test to determine building envelope tightness of your building model.

To perform a simulation, you *set the simulation parameters, run the simulation, and view simulation results*. Simulation parameters are categorized by *run control, numerics, and output properties*. *Run control* properties include the simulation methods, simulation dates and times, type of weather to use (steady or transient), and simulation time step. *Output properties* are used to set the type and amount of data you wish to have sent to the simulation results files. *Numerics properties* include convergence criteria, maximum iterations and relaxation coefficients for both airflow and contaminant equation solvers. ContamW provides default settings for these parameters. However, you will probably work more with the *run control* and *output parameters* than the *numerics parameters*.

❑ **Setting Simulation Parameters**

To set simulation parameters you use the **Set Simulation Parameters...** selection of the **Simulation** menu or the **F4** keyboard shortcut. This will display the "Simulation Parameters" property sheet. The property sheet contains three different pages of parameters including: Run Control, Numerics, and Output. Detailed descriptions of all simulation properties are given in the Simulation Parameters section of this manual.

❑ **Running Simulations via ContamW**

Once you have set the simulation parameters, you can run the simulation. To run a simulation select **Run Simulation** from the **Simulation** menu. This actually launches a separate program, ContamX, which in turn displays the "CONTAMX Simulation Control" dialog box.

While there are many factors that can affect the ability of ContamX to quickly and successfully obtain a solution (*converge*), there are situations that can cause the simulation to not converge or to converge very slowly. The relative magnitudes of the flow coefficients have some effect on the convergence of the nonlinear equation solver of ContamX [Wray and Yuill 1993]. Airflow networks containing paths whose resistances differ by many orders of magnitude will typically require more iterations than networks with paths of similar resistances. Reducing the relaxation coefficient from its default value of 0.75 to a value nearer one-half can often help convergence of these more difficult cases. The maximum number of iterations can also be increased. However, the most effective way to improve convergence is to remove those very low resistance paths that are not likely to effect the computed airflow rates. For example, when simulating a shaft (See Shafts), represent a shaft using phantom zones as opposed to using airflow elements with large cross-sectional areas between zones on adjacent levels.

❑ **Running Simulations via Command Line**

CONTAM utilizes the ContamX stand-alone simulation engine, `contamx2.exe`. ContamX utilizes the same `.PRJ` file that you create using ContamW. With the ContamX program, you can perform batch mode operations to process multiple project files. Or you can create your own file processor to modify project files in order to perform sensitivity analysis by performing multiple runs while varying desired building parameters. If you wish to modify the `.PRJ` file in such a manner, you should become familiar with the `.PRJ` file format that is now annotated with comment lines to improve the legibility of the file. (See Appendix A - PRJ File Format)

You can simply save project files in the usual manner using ContamW or use the **Create CONTAMX Input File** selection of the **Simulation** menu. The recommended course of action would be to perform a simulation by activating ContamX from within ContamW before implementing the batch process. This will provide some building check routines that insure your project is "well formed." At the very least, using the "Create CONTAMX Input File" option will

perform this *building check* prior to saving the .PRJ file. Note this building check is also performed prior to initiating a simulation from within ContamW via the **Simulation → Run Simulation** menu command. If during the *building check* routine ContamW detects problems with your project, it will provide you with feedback to assist you in correcting the problems. This feedback will be provided by displaying dialog boxes and highlighting problematic icons on the SketchPad as necessary.

❑ Viewing Simulation Results

Once you have successfully run a simulation, you can use ContamW to view simulation results or to create files that can be imported into a spreadsheet. These capabilities are explained in the Working with Results section.

3.20.1 CONTAMX Simulation Control

The "ContamX Simulation Control" dialog box allows you to control and view the status of a simulation. Once you have set the simulation parameters, activate the CONTAMX Simulation Control dialog via the **Simulation → Run Simulation** command from the ContamW menu. Press the "Start Simulation" button to run the simulation. If at any point during the simulation you want to terminate the simulation, press the "Stop Simulation" button.

Simulation Status: The current status of the simulation is displayed in this section. The date, time, day, and the current iteration of the simulation are shown. A check box is provided to terminate cyclic simulations at the end of the next cycle/day. Use this check box if it appears the simulation is not converging, or you feel enough iterations have been performed.

Simulation Settings: This field displays a summary of the current Run Control simulation parameters. This is provided for convenience to enable you to review the type of simulation that CONTAM is currently configured to perform.

Start Simulation: Press this button to start a simulation. Once you start a simulation, any existing results files for the current project file will be deleted. If you wish to save the current result files, you should either rename the files, move them to a location other than the directory in which the project file exists, or rename the current project file so that new results files are created associated with the new name.

Stop Simulation: Press this button to immediately terminate a simulation.

3.20.2 Simulation Parameters

Simulation parameters are those used to control the type of simulation that you wish ContamX to perform and the type of results output you wish ContamX to provide. This section provides detailed descriptions of the specific simulation parameters.

3.20.2.1 Simulation – Run Control Properties

Simulation Method: Set the type of simulation you wish to perform for both Airflows and Contaminants by checking the appropriate radio buttons. Refer to the Working with Simulations section for a table of allowable combinations of simulation method settings.

Airflows: You can use one of the following airflow calculation methods .

Steady: A steady state simulation will utilize the weather set by pressing the Edit Weather Button on the Run Control property page. This will calculate a single set of airflows. If any

building components have schedules associated with them, then schedule values will be determined by the *Day Type* set in the *Steady State Weather Data* (see Weather Properties under Working with Weather and Wind) and the *Steady Simulation Time*.

Transient: Transient simulations performs calculations over a user-defined period of time and regular time interval referred to as the calculation time step. When performing a transient simulation CONTAM can use either steady state weather data or weather and contaminant files that define the weather and ambient contaminant concentrations for the time period of the simulation. See Defining Transient Weather and Wind in the Working with Weather and Wind section for more details.

Duct Balance: Use this feature when working with detailed duct systems to have CONTAM adjust terminal duct balancing coefficients to provide user-defined airflow rates at the duct terminals. See Working with Ducts.

Building Airflow Test: The *building airflow test* method performs calculations that aid in verifying data input to your building model. Use this test prior to performing analysis with your building model to verify system and total airflow rates to each zone, zone temperatures and whole building air change rate. You must have only one contaminant set to be used in the simulation, and it must be a trace contaminant, when you run this test. Results of the test will be written to the validation file having the same name as the project file and the .val extension (See Working with Results). This file will be overwritten each time either the *Building Airflow Test* or *Building Pressurization Test* is performed.

Building Pressurization Test: The building pressurization method performs an automated whole-building fan pressurization test (See Special Applications of CONTAM) to determine building envelope airtightness under idealized test conditions. See the Special Applications of CONTAM section for a detailed discussion of building pressurization tests.

ContamX will temporarily set conditions for the duration of the building pressurization calculation as follows:

- no-wind
- ambient pressure set to steady state barometric pressure
- ambient and interior zones set to 20 °C
- conditioned zones (included in building volume) set to constant test pressure
NOTE: This will eliminate the effect of airflow between building zones as explained in the Special Applications of CONTAM section.
- unconditioned zones (not included in building volume) set to variable pressure
- “close” all duct terminals and duct leaks
- “turn off” all forced flow elements

When you perform a simulation using this method, ContamX will perform a steady state airflow calculation under the above conditions and calculate the total airflow rate between conditioned and unconditioned zones of the project. Results of the test will be written to the validation file having the same name as the project file and the .val extension (See Working with Results). This file will be overwritten each time either the *Building Airflow Test* or *Building Pressurization Test* is performed.

Contaminants: You can choose one of the following contaminant simulation options:

None: Do not perform contaminant analysis, only airflow analysis.

Steady: The steady contaminant option can only be used in conjunction with steady airflows and will run until the system reaches equilibrium. Contaminant sources are modeled as continuous sources with a constant generation rate. If any sources have schedules associated with them, then schedule values will be determined by the *Day Type* set in the *Steady State Weather Data* (see Weather Properties under Working with Weather and Wind) and the *Steady Simulation Time*. Burst sources will be ignored. Ambient contaminant concentrations will be the *Default Concentration* values of each contaminant.

Transient: Transient simulations performs calculations over a user-defined period of time and at a regular time interval referred to as the *calculation time step*. When performing a transient simulation CONTAM can use either constant ambient concentrations as given by default contaminant concentrations or dynamic concentrations from external contaminant files. See Contaminant Files and Working with WPC Files.

Cyclic: A cyclic simulation repeats a 24 h cycle until steady-periodic conditions are achieved. As with transient simulations, cyclic simulations can implement either steady state or dynamic weather and contaminant data. If using dynamic weather or contaminant data files (CTM and WTH), data will be obtained from the file for the date indicated by *Steady & Cyclic Simulations* date.

Short Time Step: You can use this method with either the Transient or Cyclic contaminant simulation methods. This method is required to perform simulations of one-dimensional convection diffusion zones and/or ducts. See Contaminant Numerics Properties for specific control in applying the short time step method to 1D ducts and 1D zones.

NOTE: This is an explicit solution method that should only be used for short time steps, therefore there are some checks performed to help avoid known stability problems. ContamX will check the zone airflow rates (of all zones having non-zero volumes) during airflow simulation to ensure that they do not exceed one air change per time step. If this limitation is exceeded for any zone, then the simulation will be terminated with an error message indicating the offending zone(s). The problem could also manifest itself with an error message indicating an invalid zone density was calculated due to the calculation of a negative absolute pressure.

Reset initial contaminant concentrations to...: Use this feature to reset the initial contaminant concentrations of all zones, including the implicit supply and return zones of simple air handling systems, prior to running a simulation. You can reset initial concentrations to *Zero*, *Ambient* or *Current Results Concentrations*. If you do not check one of these boxes, ContamX uses the Initial Concentrations indicated on the Contaminant Property page for each zone (See Contaminant Data under the Working with Zones section). Checking one of these boxes causes the Initial Concentration values for each zone to be overwritten.

Zero: This option sets the initial contaminant concentrations to zero.

Ambient Concentrations: This option sets the initial contaminant concentrations to the default contaminant concentrations as set when defining Project Species (See Species Properties).

Current Results Concentrations: This option sets the initial contaminant concentrations to be the same as the concentrations in each zone at the time step for which ContamW is currently displaying results on the SketchPad. You must be in the results viewing mode to utilize this feature. The current time step is displayed in the middle pane of the Status Bar (See the Status Bar section). This option is especially useful for cyclic simulation, where you may continue a simulation that has been interrupted before convergence, continue simulation with tighter convergence criteria, or create a project file for archive which can be rerun as a one-day transient simulation to quickly compute cyclic results.

Simulation Dates and Times: These dates and times are used to set the date and time labels put on steady state simulation results and to control the duration of transient simulations by allowing you to set the start and stop dates and times. Depending on the simulation method chosen different simulation "Dates and Times" fields will be enabled or disabled.

Enter dates and times according to the following formats.

Date Format – MMMDD Example - enter January 18 as Jan18

Time Format – HH:MM:SS Example - enter 1:00 AM as 01:00:00

enter 1:00 PM as 13:00:00

Steady & Cyclic Simulations: For *steady simulations*, the date field is only used to label the simulation results, however, the time field is used to determine values of scheduled items for the associated day type for which the simulation is set to utilize. You set the *day type* via the steady state Weather and Wind Parameters property sheet (See the Weather Properties in the Working with Weather and Wind section). ContamX will then utilize the day type and time to select control values of any of the elements that have schedules associated with them.

For *cyclic simulations* with steady airflows, the date field is only used to label the simulation results, and the *day type* of the steady state data is used to determine schedule values. For cyclic simulation with transient airflows, the date field is used to determine schedule values and obtain data from weather and contaminant files if being used, otherwise steady state data is used. Cyclic simulations always begin at time 00:00:00 and end at time 24:00:00, so the time field is irrelevant.

Transient Simulation, Start and Stop: These date and time fields are used to set the starting and stopping date and time of transient simulations. You should be sure to use only those dates available in both the weather and ambient contaminant files. The available dates in these files will be displayed in the Transient Weather Data and Transient Contaminant Data sections below.

Use restart file: Each transient and cyclic simulation produces a restart file that contains the status of the simulation at midnight for each day of the simulation. The restart file will have the same name as the PRJ file but have the RST extension.

To use this feature, you must first run a transient or cyclic simulation. Transient simulations must begin with time 00:00:00 and end with time 24:00:00 to create a new restart file. If a restart file is available that has the same name as the project file, the dates for which restart

data is available will be listed as "Restart data from ___ to ___" below the "Use restart file" check box. Enter the date in the edit box labeled "Restart data beginning." If a restart file is used in a simulation, the existing restart file will not be overwritten.

This feature can be useful when running long multi-day simulations with the potential for generating impractically large detailed simulation results files. You can avoid the generation of large results files by running the multi-day simulation outputting only summary data files, reviewing the results to determine days of interest, e.g., highest concentration, then rerunning the simulation for the days of interest using the restart file and outputting detailed results.

You can also use this feature to load *boundary layer source/sinks* by first running a cyclic simulation for a single day, then using the restart file to run a transient simulation for the desired period of time.

Simulation Time Steps: These are the time increments that CONTAM uses when performing transient and cyclic simulations.

Enter time steps according to the following format.

HH:MM:SS Example - enter 1 minute as 00:01:00

Calculation: The *calculation time step* is that used by ContamX when performing transient airflow and contaminant calculations for the transient and cyclic simulations. Shorter time steps provide for greater accuracy of results but require more computation time. There must be an integral number of *calculation time steps* in an hour. The minimum allowable time step is 1 second and the maximum is 1 hour.

Output: Simulation results are written to results files (See Working with Results) at the *output time step* which must be an even multiple of the *calculation time step*. Shorter time steps allow you to see the results in more detail, but also create larger results files. Utilizing an *output time step* that is longer than the *calculation time step* can still retain relatively high accuracy while reducing the size of the results files. Using these parameters you can refine your simulation period to provide detailed simulation results where needed.

Status: CONTAM uses the *status time step* to determine how often the simulation progress is updated on the CONTAMX Simulation Control dialog box. The *status time step* must be a multiple of the *calculation time step*.

3.20.2.2 Simulation – Weather Properties

Steady State Weather Data: This is a summary of the steady state weather data settings. You can change this data by pressing the "Edit Weather Data..." button which will display the weather and wind related property sheet (see Defining Steady State Weather and Wind). You can have CONTAM use steady state weather data for both steady state and transient simulations. Note that a transient simulation that uses steady state weather or WPC file will still cycle through the first seven days of *schedules* beginning on the *day type* provided by the steady state weather data.

Transient Weather Data: Dynamic weather data can be in the form of a *CONTAM weather file* (WTH file) or a *wind pressure and contaminant file* (WPC file).

Select a WTH file by pressing the associated "Browse..." button and finding the file you want to use during the simulation. Once you have selected a weather file, the date range for the particular

file will be displayed in the Weather data from ___ to ___ fields. You can also check the “Use steady state weather data” check box to reset the simulation to use the steady state data as opposed to a transient weather file. Note that if a .WTH file is being used, then *schedule* values are determined from the *day type* data provided within the .WTH file.

Transient Contaminant Data: Dynamic contaminant data can be in the form of a *CONTAM contaminant file* (CTM file).

Select an ambient contaminant file to be used by pressing the “Browse...” button and finding the file you want to use during the simulation. Once you have selected a contaminant file, the date range for the particular file will be displayed in the Contaminant data from ___ to ___ fields. You can also check the “Use steady state contaminant data” check box to reset the simulation to use the steady state data as opposed to a transient contaminant file.

WPC Data: Dynamic weather and contaminant data can also be in the form of a *wind pressure and contaminant file* (WPC file). Click the “Edit WPC Data...” to access the WPC File Parameters dialog box that is also accessible via the **Weather→WPC File...** menu item (See Working with WPC Files).

NOTES on Using WPC Files:

Note that WPC files can be used during a simulation in conjunction with the transient weather and transient contaminant files as follows:

If using wind pressures from a WPC file, you may not use a WTH file.

If using contaminant concentrations from a WPC file, you may not use a CTM file.

If the simulation is set to use a WPC file then *schedule* values will be obtained by cycling through the first seven days of schedules beginning on the *day type* provided by the steady state weather data.

When performing a cyclic simulation and using a WPC file, the WPC file must contain only one day having the same date as indicated for *Steady & Cyclic Simulations*.

3.20.2.3 Simulation – Output Properties

When you run a simulation, results are saved to files as the simulation proceeds. Detailed results for airflows and/or zone contaminant concentrations are saved to the .SIM file that is created when ever a simulation is run. The SIM file is a binary file which is not "human-readable" as opposed to human-readable text files. The data in the SIM file will be used by ContamW to display airflow results on the SketchPad, to display contaminant concentrations in the results display window, to create export and report files, and to generate charts of transient simulation results. Output is created at every *output time step*. Shorter *time steps* allow you to see the results in more detail, but lead to larger simulation results file. ContamX can create several results files in addition to the SIM file (See Working with Results), and you can control the data that will be saved during a simulation with the following parameters.

In most cases, ContamX can generate both **detailed** and/or **box-whisker** results. Detailed results are written to the corresponding file at each *output time step*. Box-whisker results include daily average, standard deviation, maximum and minimum values. Maximum and minimums are also accompanied by the time of day at which they occur for daily results, or day on which they occur for the simulation summary results. Note that box-whisker data is based on calculations

performed at the *calculation time step* but can require significantly less storage space. They can be of use when performing long simulations to find days and times of particular interest which can then be simulated in greater detail in conjunction with the *restart* (RST) file.

Airflow Simulation Results: These are the results available related to airflow calculations.

Airflow Rates: Output airflows through and pressure differences across each flow link (airflow paths, duct segments, duct leakage paths, airflow terminals and simple air handler supplies and returns) to the SIM file. This data is necessary to display or export results of airflow and pressures including those presented as SketchPad results and charts.

Building Air Change Rate: Output the whole building air change rate to the ACH file. Select **Use Standard Density** to have ContamX use standard air density for the incoming outdoor air in the air change rate calculations, otherwise the outdoor air density will be based on the outdoor temperature.

Ages of Air: Output age of air for each zone to the AGE file.

WARNING: This calculation involves the inversion of the flow matrix and could be quite demanding on computational resources for projects involving a large number of zones.

Contaminant Simulation Results: These are the results available related to contaminant calculations.

Contaminant Concentrations: Output contaminant concentrations of each contaminant node (zones and duct junctions) to the SIM file. This data is necessary to display or export results of airflow and pressures including those presented as SketchPad results and charts. Note that for 1D zones the average concentrations of all sub-cells will be saved to the SIM file. You can use zone sensors and report control nodes within 1D zones to obtain concentration time histories at specific locations within 1D zones to output to the *Controls LOG file* (See Results Files).

Occupant Exposure: Output occupant exposure data to the EBW file. This data is necessary to display occupant exposure summary information when you highlight an exposure icon. You can still plot transient exposure data without this summary information.

CONTAMX Display and Logging Options: These properties control the manner in which the simulation engine, ContamX, appears during execution and the amount of data written to the *CONTAMX.LOG* file during a simulation.

CONTAMX Display Mode: Select whether you want to interact with ContamX via a Windows dialog box or Console window (DOS window). The dialog box method is the typical choice, but you might want to run it in a console window if you have trouble with ContamX executing properly particularly for large project files.

CONTAMX Log Mode: Select the level of detail to output to the *CONTAMX.LOG* file. This is provided mainly for diagnostic purposes (for program developers) in the event that a problem is occurring during simulation (See Getting Help).

Echo .PRJ as read to .LOG: Check this box to output the project file to the *CONTAMX.LOG* file. This is mainly provided for diagnostic purposes.

3.20.2.4 Simulation – Airflow Numerics Properties

These are the parameters that you can use to adjust the methods used to solve the airflow equations. Typically, you won't need to change these values from their defaults. However, certain types of problems may benefit from using different solution techniques that are available. For instance, large projects consisting of many zones and duct junctions could benefit from the use of the PCG linear equation solver.

Non-Linear Equation Solver: Calculation of airflows requires the solution of simultaneous non-linear equations. Two algorithms are available: Newton-Raphson (N-R) with under-relaxation (SUR) and N-R using a *simple trust region* method. The latter method is more robust and is the default method. The first method is retained for compatibility with earlier versions of CONTAM. Convergence of the N-R solution of the non-linear simultaneous flow equations is controlled by four parameters:

Maximum Iterations: Use this value to prevent endless calculations if the flow iterations are not converging. This should not be a problem with an appropriate relaxation coefficient.

Relaxation Coefficient: This coefficient modifies the adjustment of the pure N-R method. Tests have found values near 0.75 to work well. Values near 1.0 may lead to non-convergent solutions. Values closer to 0.5 may be more reliable but take longer to solve. This coefficient is not used with the simple trust region method.

Relative Convergence Factor: The flow iterations are determined to have converged when the sum of all flows in and out of the zone are less than the *relative convergence factor* times the sum of the absolute values of those flows.

A test is also provided in the event of very low flows to insure that the sum of the absolute values of the flows is less than an absolute convergence factor based on a very small zone air change rate.

Linear Equation Solver: Select the method for solving the linear equations generated by the Newton-Raphson (N-R) method. ContamX uses either the Skyline algorithm (also called the profile method), which is a direct solution of the equations, or the Pre-conditioned Conjugate Gradient algorithm (PCG), which is the iterative biconjugate gradient algorithm. The execution time of the skyline method is determined by the sparsity and fill pattern of the solution matrix.

Maximum Iterations: Use this value to prevent endless calculations if the flow iterations are not converging. This is only available for the PCG method.

Relative Convergence Factor: The flow iterations are determined to have converged when the sum of all flows in and out of the zone are less than the *relative convergence factor* times the sum of the absolute values of those flows. This is only available for the PCG method.

Resequence Linear Equations: The default is to resequence the linear equations in order to speed the solution by reducing the sparsity of the solution matrix. ContamX determines whether or not the resequencing actually improves the matrix. A failure in the resequencing algorithm is not a serious error because it will not change the numerical results. The resequencing uses algorithm 582, Collected Algorithms from ACM [ACM 1982].

NOTE: If ContamX "crashes" during execution, it might work if you select not to resequence the equations or to select the *Console window* ContamX Display Option.

Linear Airflow Initialization: An initial estimate of the pressures and flows is needed to start the non-linear equation solver. The linear initialization is usually faster than starting from zero.

Adjust temperatures in flow elements: This modifies the element flow coefficients to account for the actual properties of the air flowing through the elements. This may change the computed flows by a few percent.

Variable Zone Density Parameters: These parameters are provided to control the variation of zone density during simulations due to varying temperature or accumulation/reduction of mass within a zone due to non-flow processes (e.g. non-trace contaminant source/sinks). Allowing for the variation of zone density now provides "true" transient analysis as opposed to quasi-steady analysis that does not provide for non-flow processes to take place (See Basic Equations of the Theoretical Background section).

Vary density during time-step: Check this box if you want to allow the zone density to vary during a transient simulation. If you have implemented a temperature schedule within a zone, this box will be automatically checked for you.

Max Time Step Iterations: This is the maximum number of iterations for convergence of the density values.

Under-relaxation Factor for dM/dt : Currently not implemented.

3.20.2.5 Simulation – Contaminant Numerics Properties

These are the parameters that you can use to adjust the methods used to solve the contaminant equations. Typically, you won't need to change these values from their defaults. However, as discussed below, certain types of problems may benefit from using different solution techniques that are available.

Trace Contaminant Solver and Non-Trace Contaminant Solver: For each zone there is an equation for each contaminant species that you may have selected to be either trace or non-trace contaminants. The mass fractions of the non-trace contaminants are solved within an iterative loop that also determines the airflows because the non-trace contaminants affect the zone air densities and pressures. The mass fractions of the trace contaminants are solved after the flows have been computed since they do not affect the zone air densities. These calculations require the solution of simultaneous, non-symmetric, linear algebraic equations. Four methods are provided – two direct and two iterative. The iterative algorithms are successive over-relaxation (SOR) and biconjugate gradient (BCG). The direct methods are a skyline (or 'profile') algorithm and a Gaussian LU-decomposition method. These four methods have been listed in order of increasing memory use. The iterative methods may or may not be faster than the more reliable direct methods. Before doing a long transient simulation of a large project, it may be useful to test the different methods to determine which will give optimum performance.

Skyline: The skyline algorithm is the most reliable solution, and it is very fast for small problems but can be slow for large problems.

BCG: Try the iterative biconjugate gradient algorithm for problems having a large number of zones for which the SOR algorithm might not converge.

SOR: The successive over-relaxation algorithm requires much less memory and is faster than the Skyline method unless there are convergence problems.

LU: LU decomposition using a full (non-sparse) matrix and is for verifying the solutions of the other methods rather than general usage.

The iterative methods use the following parameters.

Maximum Iterations: Used to prevent endless calculations if the iterations are not converging.

Convergence Factors: The SOR iterations are assumed converged when the relative difference between the values on successive iterations are less than the relative convergence, $e1$. The BCG method is determined to have converged when $|A \cdot x - b|/|b| < e1$.

Relaxation Coefficient: An optimum over-relaxation coefficient, ω , speeds the convergence of the SOR algorithm. Determination of an optimum will require some experimentation with different values (usually near the low end of the range 1.0 to 2.0).

The Trapezoidal Integration Factor: Determines the relative portions of the explicit and implicit terms in the solution of the differential equations. This is the parameter γ presented in the Contaminant Analysis section of the Theoretical Background.

Resequence Linear Equations: The default is to do this in order to speed the solution by reducing the sparsity of the solution matrix.

Convergence for Cyclic Simulations: Convergence is assumed when the relative difference in the peak concentrations (for all contaminants in all zones) is less than the relative convergence, $e1$. The absolute convergence, $e2$, is used to keep zones and contaminants with negligible mass fractions from dominating the solution.

1D Options: These options provide control over the 1D simulation capabilities that can be implemented when using the *Short Time Step Method* to perform contaminant simulation.

1D Zones: Select this method to implement the explicit one-dimensional convection/diffusion cell solution method for zones that have been defined as 1D zones. This will subdivide the zones into cells along user-specified 1D axes.

1D Duct Model: Select this method to implement the explicit one-dimensional convection/diffusion cell solution method for ducts. This will subdivide all duct segments into cells along their length based on the flow rate through the duct segment. Effective diffusion coefficient will be based upon the Reynolds number and the molecular diffusion coefficients of the contaminants.

If the product of the velocity u and the time step Δt in a given duct segment is less than the **Cell Size** parameter, then an Eulerian method is used with a cell size equal to that of the *Cell Size* parameter, otherwise a Lagrangian method is implemented with an effective cell size of $u \cdot \Delta t$ leading to the solution of fewer simultaneous equations as the velocity increases.

Variable Junction Temperatures: When using the 1D Duct Model, you can have CONTAM implement variable junction temperatures to account for the convective heat transfer of the air within the duct system based on inlet terminal temperatures. When using this option, the junction temperature properties are ignored and terminal inlet temperatures are obtained from the zones in which they are located.

3.21 Working with Simulation Results

CONTAM can provide you with several types of simulation results. The results available after a simulation depend on the simulation method and output parameter settings (See Output Properties in the Working with Simulations section). Once the results are available, you can view results using ContamW as described below or import files to a spreadsheet program for further analysis. Some files are created as text files while others are binary (non human-readable files). The text files can simply be imported into a spreadsheet program. The main results file is a binary file. ContamW provides the ability to generate tab-delimited text files from this binary file which can then be imported into spreadsheets. NIST also provides some post-processing tools to further manipulate the binary data and visualize results.

3.21.1 Results Files

When ContamX performs a simulation, it creates a set of results files within the same directory that the project file is stored. These results files have the same file name as the project file with different extensions appended to indicate the type of results file. The following table lists the different file extensions and gives a brief description of the files. Details of each file are provided following the table.

Extension	Type	File Description
.ACH	Text	Whole building air change rate – broken down into airflow path and ventilation system components and the combined value. This file can contain results for each time step and/or daily box-whisker results.
.AGE	Text	Age of air – age of air results for each zone at each time step and/or daily box-whisker results.
.BAL	Text	Balance – results of duct balancing calculation.
.CBW	Text	Contaminant box whisker results – only created for daily box-whisker results.
.CSM	Text	Contaminant source/sink summation – total mass of contaminants generated and/or accumulated by source/sinks and filters during a simulation.
.EBW	Text	Occupant exposure results – total exposure of each occupant to each contaminant for each time step and/or daily box-whisker results.
.LOG	Text	Controls log file – contains the output of all Report control nodes.
.RST	Binary	Simulation restart file – only for use by ContamX for restarting simulations at the beginning of intermediate days of a simulation.
.SIM	Binary	Detailed simulation results – airflow rates and pressure difference of flow links (paths and duct segments), pressures and temperatures of airflow nodes (zones and junctions), and contaminant concentrations of contaminant nodes at each time step. This is the file from which SketchPad results and graphs are produced and displayed using ContamW. It is also the file from which Export and Report files can be generated.
.VAL	Text	Model “validation” test results file – contains the results of either the Building Airflow Test or Building Pressurization Test airflow simulation methods.

Table – CONTAM Results Files

□ **<project name>.ACH – Whole Building Air Change Rate**

The air change rate is calculated based on the summation of airflow rates from the ambient zone into the *conditioned space* of the building including ducts and simple air handlers. Results are presented based on total flows into the mechanical systems (ducts and simple AHS), airflow paths and total of all flows. The *conditioned space* includes all mechanical system volume and the volume of those zones which you select to *include in building volume* (See Zone Data in the Working with Zones section).

$$ACR = 3600 \cdot \rho_{air} / V_{bldg} \cdot \sum \dot{m}$$

where

- ACR = whole building air change rate [h^{-1}]
- \dot{m} = mass flow rate into the conditioned space [kg/s]
- ρ_{air} = density of incoming air [kg/m^3]
- V_{bldg} = volume of conditioned space [m^3]

If there is a mass-imbalance for any airflow zone, ContamX will generate a warning message indicating the zone in question.

File Format (tab-delimited text):

The first line of the ACH file contains:

```
StartDate      // first date of simulation (mm/dd → IX)
EndDate        // last date of simulation (mm/dd → IX)
SimStep        // simulation time step [s] (IX)
achsaven       // save detailed results flag [0/1] (IX)
abwsaven       // save box-whisker results flag [0/1] (IX)
Vcond          // volume of conditioned space [m^3] (R4)
```

If achsave is 1, then a header line is written for the detailed data:

```
"day time path duct total"
```

If abwsaven is 1, then a header line is written for daily box-whisker data:

```
"day b-w avg dev min min max max"
```

Then for each day

If achsave is 1, then a line of data is written for each output time step for a day:

```
Date          // date (mm/dd → IX)
Time          // time (hh:mm:ss → I4)
ACRpath       // air change rate via flow paths (R4)
ACRduct       // air change rate via ducts (R4)
ACRtotal      // ACRpath + ACRduct (R4)
```

If abwsaven is 1, then three lines of daily box-whisker data are written – one for paths, ducts and total respectively. Each line contains:

```
date          // date (mm/dd → IX)
type          // "path" "duct" or "total"
avg           // average value for day (R4)
dev           // standard deviation (R4)
tmin          // time at which min occurs (hh:mm:ss → I4)
min           // minimum value for day (R4)
```

```
tmax      // time at which max occurs (hh:mm:ss → I4)
max       // maximum value for day (R4)
```

At the end:

If abwsave is 1, then three lines of summary box-whisker data for the entire simulation – one for paths, ducts and total respectively. Each line contains:

```
"final"
  type           // "path" "duct" or "total"
avg             // average value for simulation (R4)
dev            // standard deviation for sim (R4)
dmin           // date on which min occurs (mm/dd → IX)
min            // minimum value for simulation (R4)
dmax           // date on which max occurs (mm/dd → IX)
max            // maximum value for simulation (R4)
```

Example File:

```
1/1 1/1 3600 1 0 847.44
day time      path  duct  total
1/1 01:00:00  2.911 0.010 2.921
1/1 02:00:00  2.932 0.098 3.030
```

□ <project name>.AGE – Age of Air

The mean age of air of a zone is an indication of the residence time of air within a zone. ContamX can calculate these values for all the zones in a project and provide them in detailed or box-whisker form. ContamX calculates these values using the method presented in AIVC Technical 34 [AIVC 1991].

File Format (tab-delimited text):

The first line of the AGE file contains:

```
StartDate      // first date of simulation (mm/dd → IX)
EndDate        // last date of simulation (mm/dd → IX)
SimStep        // simulation time step [s] (IX)
zaasave       // save detailed zone age of air results flag [0/1] (IX)
zbwsave       // save box-whisker results flag [0/1] (IX)
```

If zaasave is 1, then a header line is written for the detailed data:

```
"day time" zone[] // ...list of zone numbers
```

If zbwsave is 1, then a header line is written for daily box-whisker data:

```
"day zone avg dev min min max max"
```

Then for each day

If zaasave is 1, then a line of data is written for each output time step for a day:

```
Date          // date (mm/dd → IX)
Time          // time (hh:mm:ss → I4)
age[]        // age of air for each zone [1/hr]
```

If zbwsave is 1, then a line of daily box-whisker data is written for each zone. Each line contains:

```
date          // date (mm/dd → IX)
zone          // zone number (IX)
```

```

avg          // average value for day (R4)
dev          // standard deviation (R4)
tmin        // time at which min occurs (hh:mm:ss → I4)
min         // minimum value for day (R4)
tmax        // time at which max occurs (hh:mm:ss → I4)
max         // maximum value for day (R4)

```

At the end:

If zbwsave is 1, a line of summary box-whisker data is written for each zone for the entire simulation. Each line contains:

```

"final"
zone        // zone number (IX)
avg         // average value for simulation (R4)
dev         // standard deviation for sim (R4)
dmin        // date on which min occurs (mm/dd → IX)
min         // minimum value for simulation (R4)
dmax        // date on which max occurs (mm/dd → IX)
max         // maximum value for simulation (R4)

```

Example File:

```

3      1/1  1/1  300  0    1
day   zone  avg  dev  min  min  max  max
1/1   1     1.000 0.000 00:05:00  1.000 00:05:00  1.000
1/1   2     2.000 0.000 00:05:00  2.000 00:05:00  2.000
1/1   3     3.000 0.000 00:05:00  3.000 00:05:00  3.000
final 1     1.000 0.000 1/1   1.000 1/1   1.000
final 2     2.000 0.000 1/1   2.000 1/1   2.000
final 3     3.000 0.000 1/1   3.000 1/1   3.000

```

❑ <project name>.BAL – Duct Balance Results File

This file contains the results of a duct balance operation if performed by ContamX. It contains the status of the balance operation along with detailed results and error tracking information if required.

File Format (tab-delimited text):

The first line of the BAL file contains the status of the balance procedure:

```

Code          // balance completion code [0-5] (IX)
!Message[]    // comment containing completion message (I1)

```

The next two lines are headers for fan and terminal data:

```

!fan  duct #    flow [kg/s]    Prise [Pa]    RPMratio
!trm  jctn #    rel-flow [-]   Cb [-]

```

Then for each balancing terminal and associated fans:

If terminal:

```

"trm"        // indicates terminal data
junc         // junction number of terminal (IX)
rflow        // relative flow Fbal/Fdes (R4)
Cb           // balance coefficient (R4)

```

If fan:

```

"fan"        // indicates fan data
duct         // duct segment number of fan(IX)

```

```

flow          // total fan flow (R4)
prise        // pressure rise across fan (R4)
If fan performance curve fan:
frpm         // RPM ratio required to achieve balance
            // flow rate with design fan (R4)

```

The last line is exactly:

```
end
```

Example File:

```

0 ! Flow balance iterations converged
!fan  duct #  flow [kg/s]  Prise [Pa]  RPMratio
!trm  jctn #  rel-flow [-]  Cb [-]
fan   4      0.334472    68.0745
trm   1      1.000025     4.38419
trm   2      0.999974     4.2013
fan   8      0.334472    107.938    0.78357
trm  11      1.000007     7.76847
trm  12      0.999991     7.58544
end

```

☐ <project name>.CBW – Contaminant Box Whisker

This file contains contaminant daily box-whisker results and summary box-whisker results for the entire simulation period. Results are presented for all zones within a project file as well as for each contaminant simulated.

File Format (tab-delimited text):

The first line of the CBW file is a header describing the amount of data in the file:

```

NumZones      // number of zones (IX)
NumCont       // number of contaminants (IX)
StartDate     // first date of simulation (mm/dd → IX)
EndDate       // last date of simulation (mm/dd → IX)

```

The second line is a header:

```
"day zone ctm avg dev min min max max"
```

Then daily box-whisker results for each day

for each zone

for each contaminant:

```

date          // date (mm/dd → IX)
zone          // zone number (IX)
ctm           // contaminant number (IX)
avg           // average value for day (R4)
dev           // standard deviation (R4)
min           // minimum value for day (R4)
tmin          // time at which min occurs (hh:mm:ss → I4)
max           // maximum value for day (R4)
tmax          // time at which max occurs (hh:mm:ss → I4)

```

After all daily values comes the summary for the entire simulation

for each zone

for each contaminant:

```
"final"
```

```

zone           // zone number (IX)
ctm            // contaminant number (IX)
avg           // average value for simulation (R4)
dev           // standard deviation for sim (R4)
min           // minimum value for simulation (R4)
dmin          // date on which min occurs (mm/dd → IX)
max           // maximum value for simulation (R4)
dmax          // date on which max occurs (mm/dd → IX)
    
```

Example File:

```

2 1 1/1 1/1
day zone ctm avg dev min min max max
1/1 0 1 0.000e+000 0.000e+000 00:05:00 0.000e+000 00:05:00 0.000e+000
1/1 1 1 1.036e-004 0.000e+000 00:05:00 1.036e-004 00:05:00 1.036e-004
1/1 2 1 6.839e-005 4.049e-005 00:05:00 0.000e+000 24:00:00 1.019e-004
final 0 1 0.000e+000 0.000e+000 1/1 0.000e+000 1/1 0.000e+000
final 1 1 1.036e-004 0.000e+000 1/1 1.036e-004 1/1 1.036e-004
final 2 1 6.839e-005 4.049e-005 1/1 0.000e+000 1/1 1.019e-004
    
```

□ <project name>.CSM – Contaminant Source/Sink Summation

This file contains summary data related to contaminant simulation. This file is created automatically by ContamX whenever transient contaminant simulations are performed. Source/sink, filter and zone numbers are provided for referencing back to ContamW. The file consists of six sections:

Contaminant source/sink summary – This section provides the mass of contaminant generated, removed and stored for each source/sink during a simulation.

Filter challenge summary – This section provides the mass of contaminant to which each filter is exposed during a simulation (upstream challenge).

Filter loading summary – This section provides the mass of contaminant “loaded” onto each filter during a simulation (not including initial loading).

Filter breakthrough summary – This section provides indication of whether or not filter breakthrough applies. If breakthrough applies and occurs then the date and time when it occurs and the breakthrough efficiency are provided, otherwise the efficiency at the end of the simulation is provided. Breakthrough only applies to *Simple Gaseous Filters* for which a non-zero breakthrough efficiency is provided.

Interactions with ambient – This section provides the mass of each contaminant that enters and exits the building volume, i.e., those that are set to be included in the building volume.

Simulation time – This section provides information on the duration of the simulation.

File Format (tab-delimited text):

The first two lines of the CSM file display the PRJ file name and the date and time the simulation was run:

```

"Project:"   FilePath[] // the full path of the PRJ file (I1)
"Time:"      SimTime[]  // PC time the simulation was started (I1)
    
```

```
-- Blank line -
```

This is followed by the number of contaminants simulated.

```
NumCont "contaminants" // number of contaminants (I4)
```

```
-- Blank line -
```

This is followed by the **Contaminant source/sink summary section**.

NOTE: the following section is only presented if source/sinks exist within the PRJ file.

The first line of the source/sink section is a header:

```
"Contaminant source/sink summary."
```

The next line contains the number of source/sinks:

```
NumSS "source/sinks" // number of source/sinks(IX)
```

The next line is a header:

```
s/s# zn# species released removed stored [kg]
```

Then a line of data for each source/sink:

```
SSNum // source/sink number (IX)
ZoneNum // zone number (IX)
Name[] // contaminant name (I1)
MassRel // mass released(R4)
MassRem // mass removed [kg] (R4)
MassAcc // mass accumulated including previous storage (R4)
```

```
-- Blank line -
```

This is followed by the **Filter challenge section**.

NOTE: this section is only presented if filters exist within the PRJ file.

The first line of the section is a header:

```
"Filter challenge summary."
```

The next line contains the number of filters:

```
NumFilt "filters" // number of filters (IX)
```

The next line is a header:

```
"path" Ctm[nc] "element" // Ctm[nc] is a list of contaminants where
// nc is the number of contaminants.
```

Then a two lines of data for each filter:

The first line is:

```
FltNum // number of flow component that contains the filter
// and letter indicating type (p:path, d:duct,
// t:terminal)
MassUp[nc] // challenge mass of each contaminant [kg] (R4)
Name[] // filter element name (I1)
```

The second line is:

```
"--" // indicates same filter as previous
MassDown[nc] // unfiltered mass of each contaminant [kg] (R4)
"downstream" // indicates downstream mass
```

```
-- Blank line -
```

This is followed by the **Filter loading section**.

NOTE: this section is only presented if filters exist within the PRJ file.

The first line of the section is a header:

```
"Filter loading summary."
```

The next line contains the number of filters:

```
NumFilt // number of filters (IX)
```

The next line is a header:

```
"path" Ctm[nc] "element ns" // Ctm[nc] is a list of contaminants
                                // nc is the number of contaminants.
```

Then a line of data for each filter:

```
FltNum          // number of flow component that contains the filter
                // and letter indicating type (p:path, d:duct,
                // t:terminal)
MassLoad[nc]    // mass filtered during simulation [kg] (R4)
Name[]          // filter element name (I1)
ns              // number of sub-elements, ns>0 for super-filter (I4)
```

If ns>0, then a line of data for each sub-filter of the super-filter:

```
"--"           // indicates sub-filter
MassLoad[nc]   // mass filtered during simulation [kg] (R4)
Name[]         // filter element name (I1)
"0"            // zero sub-elements (I4)
```

-- Blank line -

*This is followed by the **Filter breakthrough section**.*

NOTE: this section is only presented if filters exist within the PRJ file.

The first line of the section is a header:

```
"Filter breakthrough summary."
```

The next line contains the number of filters:

```
NumFilt "filters" // number of filters for which for which breakthrough
                                // is possible, currently on Simple Gaseous types (IX)
```

The next two lines are headers:

```
"path element"
"  contaminant date time efficiency"
```

Then a line of data for each filter:

```
FltNum          // number of flow component that contains the filter
                // and letter indicating type (p:path, d:duct, t:terminal)
Name[]          // filter element name (I1)
nc              // number of contaminants (I4)
```

Then a line of data for each contaminant for which breakthrough is specified.

Each line contains:

```
"--"           // indicates continuation of filter
Ctm[]          // contaminant name (I1)
```

If breakthrough occurred:

```
Date[]         // date on which breakthrough occurs (MMDD → I4)
Time           // time at which breakthrough occurs (hh:mm:ss → I4)
bkeff          // breakthrough efficiency
```

If breakthrough does not occur:

```
"--"           // indicates no breakthrough date
"--"           // indicates no breakthrough time
eff            // efficiency at end of simulation
```

-- Blank line -

This is followed by the *Interactions with ambient section*.

The next two lines of the section are headers. They are exactly:

```
"Interactions with ambient."
"contaminant flow from flow to [kg]"
```

Then a line of data for each contaminant:

```
Ctm[]          // contaminant name (I1)
MassFrom       // total mass exiting from interior zones [kg] (R4)
MassTo         // total mass entering from ambient zone [kg] (R4)
```

-- Blank line --

The final section is the *Simulation time section*:

It contains a header and three lines of data:

```
"Simulation time:"
tsec "seconds" // simulation time in seconds [s] (IX)
thr "hours,"   // simulation time in hours [h] (R4)
tday "days"   // simulation time in days [days] (R4)
```

Example File:

```
Project: C:\CONTAM96\samples\CW2.4\Filters\TestBK3.prj
Time: Mon Jul 18 10:17:22 2005
```

5 contaminants

Contaminant source/sink summary.

```
4 source/sinks
s/s# zn# species released removed stored [kg]
1 3 CO 7.200e-001 0.000e+000 0
2 3 Cl2 1.000e-004 0.000e+000 0
4 4 Cl2 0.000e+000 1.044e-001 0
3 4 Cl2 8.656e-003 8.669e-003 1.231e-005
```

Filter challenge summary.

```
3 filters
path CO CO2 Cl2 part0.10 part1.00 element
6p 1.164e-001 1.829e+000 2.361e-001 1.204e-001 1.204e-001 Both
-- 6.215e-002 1.829e+000 1.079e-001 1.039e-001 7.389e-002 downstream
1d 1.164e-001 1.829e+000 2.361e-001 1.204e-001 1.204e-001 MERV10
-- 1.164e-001 1.829e+000 2.361e-001 1.039e-001 7.389e-002 downstream
2t 1.164e-001 1.829e+000 2.361e-001 1.039e-001 7.388e-002 GF0
-- 6.213e-002 1.829e+000 1.079e-001 1.039e-001 7.388e-002 downstream
```

Filter loading summary.

```
3 filters
path CO CO2 Cl2 part0.10 part1.00 element ns
6p 5.427e-002 0.000e+000 1.282e-001 1.649e-002 4.652e-002 Both 2
-- 0.000e+000 0.000e+000 0.000e+000 1.649e-002 4.652e-002 MERV10 0
-- 5.427e-002 0.000e+000 1.282e-001 0.000e+000 0.000e+000 GF0 0
1d 0.000e+000 0.000e+000 0.000e+000 1.648e-002 4.651e-002 MERV10 0
2t 5.427e-002 0.000e+000 1.282e-001 0.000e+000 0.000e+000 GF0 0
```

Filter breakthrough summary.

```
2 filters
path element nc
6p Both 2
-- CO Jan01 01:01:00 0.3
-- Cl2 Jan01 00:58:00 0.5
2t GF0 2
-- CO Jan01 01:01:00 0.3
-- Cl2 Jan01 00:58:00 0.5
```

Interactions with ambient.

```
contaminant  mass from      mass to [kg]
CO           0.000e+000    2.067e-001
CO2          0.000e+000    4.938e+000
Cl2          0.000e+000    3.011e-001
part0.10    0.000e+000    2.954e-001
part1.00    0.000e+000    2.414e-001
```

```
Simulation time:
7200  seconds
2    hours
0.0833333  days
```

□ **<project name>.EBW – Occupant Exposure**

Occupant exposure can only be calculated for transient contaminant simulations. The occupant exposure is calculated based on the contaminant concentrations within the zones (or zone cells when using 1D zones with the short time step method) occupied by each occupant. These zones (or cells) are determined based on the associated occupancy schedules. The exposure is determined by integrating the concentration vs. time using the mean-value of the contaminant concentrations to which the occupant is exposed during each simulation time step. The resultant integral is divided by the output time step then written to the file in units of $\text{kg}_{\text{cont}}/\text{kg}_{\text{air}}$.

File Format (tab-delimited text):

The first line of the EBW file is a header describing the data in the file:

```
NumExps      // number of exposure icons (IX)
NumCont      // number of contaminants (IX)
StartDate    // first date of simulation (mm/dd → IX)
EndDate      // last date of simulation (mm/dd → IX)
SimStep      // simulation time step [s] (IX)
expsave     // save detailed results flag [0/1] (IX)
ebwsave     // save box-whisker results flag [0/1] (IX)
```

If expsave is 1, then a header line is written for the detailed data:

```
"day time ctm" exp[] // exp[] is a list of exposure icon numbers
```

If ebwsave is 1, then a header line is written for daily box-whisker data:

```
"day pexp ctm avg dev min min max max"
```

Then for each day

If expsave is 1, then a line of data is written for each for each contaminant and output time step for a day:

```
Date      // date (mm/dd → IX)
Time      // time (hh:mm:ss → I4)
exp[]     // exposure for each exp icon for timestep [kg/kg]
```

If ebwsave is 1, then a line of daily box-whisker data is written for each exposure icon:

```
date      // date (mm/dd → IX)
exp       // exposure icon number (IX)
ctm       // contaminant number (IX)
avg       // average value for day (R4)
dev       // standard deviation (R4)
tmin      // time at which min occurs (hh:mm:ss → I4)
min       // minimum value for day (R4)
tmax      // time at which max occurs (hh:mm:ss → I4)
max       // maximum value for day (R4)
```

At the end:

If *ebwsave* is 1, a line of summary box-whisker data is written for each exposure icon and for each contaminant for the entire simulation. Each line contains:

```
"final"
exp // exposure icon number (IX)
ctm // contaminant number (IX)
avg // average value for simulation (R4)
dev // standard deviation for sim (R4)
dmin // date on which min occurs (mm/dd → IX)
min // minimum value for simulation (R4)
dmax // date on which max occurs (mm/dd → IX)
max // maximum value for simulation (R4)
```

Example File:

```
1 2 1/1 1/1 300 1 1
day time ctm 1
day pexp ctm avg dev min min max max
1/1 01:00:00 1 2.811e-007
1/1 01:00:00 2 9.970e-008
1/1 02:00:00 1 2.571e-007
1/1 02:00:00 2 6.109e-008
1/1 1 1 3.021e-007 0.000e+000 02:00:00 2.571e-007 00:05:00 3.592e-007
1/1 1 2 1.061e-007 1.945e-008 02:00:00 6.109e-008 00:05:00 1.562e-007
final 1 1 3.021e-007 0.000e+000 1/1 2.571e-007 1/1 3.592e-007
final 1 2 1.061e-007 1.945e-008 1/1 6.109e-008 1/1 1.562e-007
```

❑ <project name>.LOG

If you perform a transient simulation and implement *report control elements*, ContamX will create a file in the same directory as your project file with the name of your project file and the .LOG extension appended (See Control Element Type: *Report a Value*). This file will contain a line of data at each *output time step* and a column of signal values for each *report control element*.

❑ <project name>.RST – Simulation Restart File

Restart is an alternative to normal initialization of a simulation. It contains the state of all airflow and contaminant nodes, flow rates and pressure differences for all flow paths, contaminant storage terms for sinks, and signal values of various control node types. State data will be stored at 24:00:00 of all days simulated.

File Format (binary):

Header data:

```
m[0] = 0L; /* TURBO C++ messes up the first bytes written; */
m[1] = (I4)_nzone; /* therefore, send 4 unused bytes. */
m[2] = (I4)_npath;
m[3] = (I4)_nctm;
m[4] = (I4)_njct;
m[5] = (I4)_ndct;
m[6] = (I4)_ncss;
m[7] = (I4)_nctrl;
m[8] = (I4)_rcdat.date_0;
m[9] = (I4)_rcdat.date_1;
m[10] = (I4)_rSizeData;
m[11] = (not set)
```

m[1] thru m[7] allow ContamW to check for some changes in the project.

m[8] and m[9] are the date limits displayed to the user.

m[11] will allow reading all dates on the file -- could be used to create a selection box of available dates.

RESTART DATA:

For all AF_NODES:

```
R8 T - temperature
R8 P - pressure
R8 M - mass
R4 Mf[_nctm] - mass fractions
```

For all AF_PATHs:

```
R8 Flow[0] - primary flow
R8 Flow[1] - secondary flow
R8 dP - pressure drop
```

For CSS_DSC:

```
CSE_EDS (I4)pss->local (stored as R4, converted to I4 in simulation)
CSE_BLS (R4)pss->local
```

For CT_NODES:

```
SNSDAT: R4 oldsig
PICDAT: R4 oldsig, R4 olderr
```

❑ **<project name>.SIM – Detailed Simulation Results**

The SIM file is the main results file of ContamX containing detailed airflow and contaminant results. The data in the SIM file is used by ContamW to display airflow results on the SketchPad, to display contaminant concentrations in the results display window, to create export and report files, and to generate charts of transient simulation results. Output is created at every *output time step*. Shorter *output time steps* allow you to see the results in more detail, but lead to larger simulation results file.

The format of this file has been slightly modified from the previous version to accommodate a greater number of building components (zones, paths, ducts, etc.). This is reflected in the “nr” fields that have changed from I2 to I4. It is still a binary file (not “human-readable” as opposed to a text file) to provide for faster access and smaller file size than a text file.

Several **post-processor programs** are available from the NIST website (www.bfrl.nist.gov/IAQanalysis) that can be used to read the .SIM file and generate various forms of output from it. These programs include: SimRead2, SimReadW and ContamRV.

SimRead2 is a DOS-based command-line program that can be used to generate user-selected output from the SIM file. Output is in the form of tab-delimited files. SimRead2 can be used in a batch processing mode to minimize user-interaction when faced with the redundancy of manipulating multiple data sets.

SimReadW is a Windows program that can also generate user-selected output from the SIM file. SimReadW also provides the added capability of performing averaging of values over time and/or groups of building components, e.g., zones. It also enables the selection of either column-wise or row-wise orientation of the output.

The CONTAM Results Viewer, **ContamRV**, enables the visualization of contaminant concentrations on a color-coded image of the SketchPad. The program provides you with a color-coded display of zone contaminant concentrations, enables the animation of transient simulation results and displays zone concentrations in numerical format. A whole-building view is provided along with the ability to select individual levels of a project for detailed review.

File Format (binary):

Items that appear in **red** are those whose data size has been increased from 2-bytes to 4-bytes to allow for an increased number of building components. This change took place between versions 2.3 and 2.4.

The first 16 lines of the simulation results file contain data (32-bit integers) to help assure that the results apply to the project file currently in ContamW and to set the array sizes necessary to process the results.

```

24          // CONTAM version number id (I4)
_nzone     // number of airflow zones (excluding ambient) (I4)
_npath     // number of airflow paths (I4)
_nctm      // number of contaminants (I4)
_njct      // number of junctions and terminals (I4)
_ndct      // number of duct segments (I4)
_time_list // listing time steps [s] (I4)
_date_0    // start of simulation - day of year (I4)
_time_0    // start of simulation - time of day (I4)
_date_1    // end of simulation - day of year (I4)
_time_1    // end of simulation - time of day (I4)
_pfsave    // if true, write path flow results (I4)
_zfsave    // if true, write zone flow results (I4)
_zcsave    // if true, write zone contaminant results (I4)
_nafnd     // number of airflow nodes (zones + junctions) (I4)
_nccnd     // number of contaminant nodes (zones + junctions) (I4)
_nafpt     // number of airflow paths (paths + ducts) (I4)
    
```

This is followed by `_nafnd` lines of airflow node cross-reference data:

```

typ        // source of node [zone or junction] (I4)
nr         // zone or junction number (I4)
    
```

The next `_nafnd` lines give the contaminant node cross-reference data:

```

typ        // source of node [zone or junction] (I4)
nr         // zone or junction number (I4)
    
```

The next `_nafpt` lines give the airflow path cross-reference data:

```

typ        // source of path [path, duct, or leak] (I4)
nr         // path, duct, or leak number (I4)
    
```

The simulation results for each day consist of:

The results for each time step consist of:

A line of time and ambient data:

```

dayofy     // day of year [1 to 365] (I2)
daytyp     // type of day [1 to 12] (I2)
sim_time   // time value [s] [0 to 86400] (I4)
Tambt     // ambient temperature [k] (R4)
P          // barometric pressure [Pa] (R4)
Ws        // wind speed [m/s] (R4)
Wd        // wind angle [deg] (R4)
    
```

```

CC[0]      // ambient mass fraction of species 0 [kg/kg] (R4)
...
CC[n]      // ambient mass fraction of species n [kg/kg] (R4)

```

A line of data for each airflow path:

```

nr         // path number; use as check (I4)
dP         // pressure drop across path [Pa] (R4)
Flow0     // primary flow value [kg/s] (R4)
Flow1     // alternate flow value [kg/s] (R4)

```

A line of data for each airflow node (excluding ambient):

```

nr         // node number; use as check (I4)
T         // node temperature [K] (R4)
P         // node reference pressure [Pa] (R4)
D         // node air density [kg/m^3] (R4)

```

A line of data for each contaminant node (excluding ambient):

```

nr         // node number; use as check (I4)
CC[0]     // mass fraction of species 0 [kg/kg] (R4)
...
CC[n]     // mass fraction of species n [kg/kg] (R4)

```

The time step data is followed by summary data for the day.

It begins with the following line of ambient data:

```

dayofy    // day of year [1 to 365] (I2)
daytyp    // type of day [1 to 12] (I2)
Tamax     // maximum ambient temperature [k] (R4)
Tamin     // minimum ambient temperature [k] (R4)
Pavg      // average barometric pressure [Pa] (R4)
Wsmax     // maximum wind speed [m/s] (R4)
Wsaveg    // average wind speed [m/s] (R4)
CC[0]     // maximum ambient mass fraction of species 0 [kg/kg] (R4)
...
CC[n]     // maximum ambient mass fraction of species n [kg/kg] (R4)

```

A line of data for each airflow path:

```

nr         // path number; use as check (I4)
dPmax     // maximum pressure drop across path [Pa] (R4)
Flowmax   // maximum primary flow value [kg/s] (R4)
0.0       // place holder (R4)

```

A line of data for each airflow node (excluding ambient):

```

nr         // node number; use as check (I4)
T         // node temperature [K] (R4)
P         // node reference pressure [Pa] (R4)
D         // node air density [kg/m^3] (R4)

```

A line of data for each contaminant node (excluding ambient):

```

nr         // node number; use as check (I4)
CCmax[0]  // maximum mass fraction of species 0 [kg/kg] (R4)
...
CCmax[n]  // maximum mass fraction of species n [kg/kg]

```

Programmer's Note: this file requires that the structures in ContamW and ContamX be compiled using no greater than 2-byte member alignment (under Visual C++). The file is unreadable if the default structure member alignment is used.

□ **<project file>.VAL**

The VAL file contains the results of either the Building Airflow Tests or Building Pressurization Test airflow simulation methods. The file will be created or overwritten each time you perform one of these tests on a project file, so be sure to rename or copy any files you wish to keep.

If you perform a **Building Airflow Test** calculation, the data written to this file will be divided into five sections: Building info and ACH, Zones, Junctions, Classified Flows and Volumes. The first and last sections are always written, but the other three are controlled by the simulation output parameters for the Airflow Test simulation method.

Building Airflow Test – File Format (tab-delimited text):

The first line of the file is a header providing the name of the PRJ file:

```
"Airflow Summary for" file[] // PRJ file name (I1)
-- Blank line --
```

Weather conditions section:

```
"Ambient T:" Tmbt "C" // ambient temperature [C] (R4)
"Pressure:" Pbar "Pa" // barometric pressure [Pa] (R4)
"Wind spd:" wspd "m/s" // wind speed [m/s] (R4)
"Wind dir:" wdir "deg" // wind direction [deg] (R4)
"Time:" time day[] // sim time (hh:mm:ss → I4) day of week (I1)
-- Blank line --
```

*Whole building air change rate – total flow from conditioned to unconditioned zones divided by the total volume of conditioned zones. Air change rate is calculated as $ACH = F * 3600 / (ZoneDensity * ZoneVol)$. ACH can be calculated from data provided in the Classified Flow section :*

```
"Bldg ACH:" ach // building air change rate [1/hr] (R4)
-- Blank line --
```

Zones section:

This section is only written to the file if the simulation output parameter BldgFlowZ is equal to 1.

```
"Zones:" nZones "flows [ACH]" // number of zones (I4) plus header
```

Data header:

```
zone C/U Supply Ret/Exh OA sys OA tot Circ tot P [Pa] T [C] Vol [m^3]
```

A line of data for each zone (excluding implicit zones of simple air handling systems, i.e., cu = S) – all flows provided as air changes per hour, ACH [1/hr]:

```
nr // zone number (I4)
cu // C:conditioned, U:unconditioned, A:ambt, S:AHS
Qs // Total supply airflow from system to zone.
// Includes flow from: terminals, simple ahs, duct leaks
// and forced flow paths
Qrex // Total return + exhaust airflow from zone to system.
// Includes flow to: terminals, simple ahs, duct leaks
// and forced flow paths
Qoasys // Total outdoor airflow from system to zone.
// Includes flow from: terminals, simple ahs, duct leaks
// and forced flow paths connected to Ambt
Qoatot // Total outdoor air intake including all paths (non-forced flow)
```

```

// connected to ambient: Qoasys + direct infiltration
QcircTot // "Circulation flow" sum of all flows out of zone
// Includes flow to: terminals, simple ahs, duct leaks,
// conditioned zones, unconditioned zones and Ambt
P // Zone pressure = absolute zone pressure minus the ambient
// pressure at the reference elevation of the level on
// which the zone is located [Pa]
T // Zone temperature [C]
V // Zone volume [m³]
-- Blank line --

```

Junctions section:

This section is only written to the file if the simulation output parameter BldgFlowD is equal to 1.

```
"Junctions:" nJct // number of junctions (I4)
```

Data header:

```
junction J/T OA [%] Flow [kg/s] P [Pa] T [C] Vol [m³]
```

A line of data for each junction:

```

nr // junction number (I4)
jt // J:junction, T:terminal
poa // Percent outdoor air [%]
w // Airflow rate [kg/s]
P // Junction pressure [Pa]
T // Junction temperature [C]
V // Junction volume [m³]
-- Blank line --

```

Flow classification section:

This section is only written to the file if the simulation output parameter BldgFlowC is equal to 1.

```
"Classified Zone Airflows for " file[] ".prj" // PRJ file name (I1)
```

```
-- Blank line --
```

```
"Zones:" nZones "flows [kg/s]" // number of zones (I4) plus header
```

Data header:

```

"zone C/U Vol [m³] Dens [kg/m³] fmTerm fmLeak fmAHS fmFan fmCzone fmUzone
fmAmbt toTerm toLeak toAHS toFan toCzone toUzone toAmbt oaTerm oaLeak oaAHS
oaFan"

```

A line of data for each zone (excluding implicit zones of simple air handling systems, i.e., cu = S) – all flows provided in kg/s:

```

nr // zone number (I4)
cu // C:conditioned, U:unconditioned, A:ambt, S:AHS
V // Zone volume (volume for A is total cond volume
// including zones, ducts and simple AHS) [m³]
Dens // Zone density [kg/m³]
FfmTrm // Airflow rate into zone from terminals
FfmLeak // Airflow rate into zone from duct leaks
FfmAHS // Airflow rate into zone from simple AHS
FfmFan // Airflow rate into zone from forced flow elements
FfmZnC // Airflow rate into zone from conditioned zones
FfmZnU // Airflow rate into zone from unconditioned zones (except Ambt)
FfmAmbt // Airflow rate into zone from ambient (except forced flows)

```

```

FtoTrm // Airflow rate out of zone to terminals
FtoLeak // Airflow rate out of zone to duct leaks
FtoAHS // Airflow rate out of zone to simple AHS
FtoFan // Airflow rate out of zone to forced flow elements
FtoZnC // Airflow rate out of zone to conditioned zones
FtoZnU // Airflow rate out of zone to unconditioned zones (except Ambt)
FtoAmbt // Airflow rate out of zone to Ambt
FoaTrm // Outdoor airflow rate into zone from terminals
FoaLeak // Outdoor airflow rate into zone from duct leaks
FoaAHS // Outdoor airflow rate into zone from simple AHS
FoaFan // Outdoor airflow rate into zone from forced flow elements
-- Blank line --

```

Volume section:

```

"Volumes:"
"Conditioned zones"
VZCm "m^3" // Total conditioned zone volume [m3]
VZCft "cuft" // Total conditioned zone volume [ft3]
"Ducts & AHS (conditioned)"
VDCm "m^3" // Total system volume [m3]
VDCft "cuft" // Total system volume [ft3]
"Unconditioned zones"
VZUm "m^3" // Total unconditioned zone volume [m3]
VZUft "cuft" // Total unconditioned zone volume [ft3]

```

Example File:

Airflow Summary for DEMO7_AirflowTest.prj

```

Ambient T: 20.00 C
Pressure: 101325.00 Pa
Wind spd: 0.00 m/s
Wind dir: 0.00 deg
Time: 00:00:00 Sunday

```

Bldg ACH: 0.450

Zones: 5		flows [ACH]								
zone	C/U	Supply	Ret/Exh	OA sys	OA tot	Circ tot	P [Pa]	T [C]	Vol [m ³]	
1	C	0.78505	0	0.18038	0.18038	0.79	949.0	20.0	216.0	
2	C	3.1401	0	0.7215	0.7215	3.14	949.0	20.0	54.0	
3	C	1.5702	6.0457	0.36079	0.36079	7.85	949.0	20.0	108.0	
4	C	9.4165	0	2.1636	2.1636	9.42	949.0	20.0	18.0	
5	C	4.7083	0	1.0818	1.0818	4.71	949.0	20.0	36.0	

Junctions: 12							
junction	J/T	OA [%]	Flow [kg/s]	P [Pa]	T [C]	Vol [m ³]	
1	T	23.0	0.056717	78.0	20.0	0.00	
2	T	23.0	0.056715	63.3	20.0	0.01	
3	T	23.0	0.056721	20.3	20.0	0.01	
4	T	23.0	0.056692	0.0	20.0	0.01	
5	T	100.0	0.065148	-0.0	20.0	0.00	
6	J	23.0	0.28354	-0.2	20.0	0.11	
7	J	23.0	0.28354	79.3	20.0	0.05	
8	J	23.0	0.22682	64.5	20.0	0.06	
9	J	23.0	0.17011	21.4	20.0	0.09	
10	J	23.0	0.11338	1.1	20.0	0.07	
11	T	0.0	0.21839	-880.2	20.0	0.04	
12	T	23.0	0.056692	0.0	20.0	0.01	

Classified Zone Airflows for DEMO7_AirflowTest.prj

Zones: 6		flows [kg/s]								
zone	C/U	Vol [m ³]	Dens [kg/m ³]	fmTerm	fmLeak	fmAHS . . .	oaLeak	oaAHS	oaFan	
0	A	432.45	1.2041	0	0	0 . . .	0	0	0	
1	C	216	1.2041	0.056717	0	0 . . .	0	0	0	
2	C	54	1.2041	0.056715	0	0 . . .	0	0	0	
3	C	108	1.2041	0.056721	0	0 . . .	0	0	0	
4	C	18	1.2041	0.056692	0	0 . . .	0	0	0	
5	C	36	1.2041	0.056692	0	0 . . .	0	0	0	

Volumes:

Conditioned zones

432.00 m³

15255.9 cuft

Ducts & AHS (conditioned)

0.45 m³

15.8 cuft

Unconditioned zones

0.00 m³

0.0 cuft

Building Pressurization Test – File Format (tab-delimited text):

The first line of the file is a header providing the name of the PRJ file:

```
"Building Pressurization Test for" file[] // PRJ file name (I1)
-- Blank line --
"Pressurization:"
Ptest      "Pa"           // Test pressure (R4)
Ptest      "in.H2O"      // positive => pressurization
                          // negative => depressurization

-- Blank line --
"Mass flow rate:"
F "kg/s"    // Flow rate required to obtain Test pressure (R4)
F "kg/h"
F "scfm"
-- Blank line --
"Volume flow rate:"
Q "L/s"     // Flow rate required to obtain Test pressure (R4)
Q "m^3/h"
Q "cfm"
-- Blank line --
"Ambient conditions:"
"Pressure"
Pambt "Pa"   // Barometric pressure (R4)
Pambt "in.H2O"
"Temperature"
Tambt "C"    // Outdoor temperature (R4)
Tambt "F"
"Air Density"
Dambt "kg/m^3" // Outdoor air density (R4)
Dambt "lb/ft^3"
-- Blank line --
"Volumes:"
"Conditioned zones"
VZC "m^3" // Total volume of conditioned zones
VZC "cuft"
```

```

"Ducts & AHS (conditioned)"
VD "m^3" // Total volume of ducts and AHS
VD "cuft"
"Unconditioned zones"
VZU "m^3" // Total volume of conditioned zones
VZU "cuft"
-- Blank line --
"Airflow details:"
"Path Czone Uzone dP [Pa] [in.H2O] F [kg/s] [kg/h] [scfm] Q [L/s] [m^3/h] [cfm]"

```

A line of data for each flow path contains airflow details. All flows and pressure differences are absolute values. If test is pressurization test, then all flows are from conditioned to unconditioned zones. If a depressurization test, then flows are from unconditioned to conditioned zones.

```

nr // path number
Czone // number of conditioned zone
Uzone // number of unconditioned zone
dP_Pa // pressure difference across path [Pa]
dP_iH2O // [in.H2O]
F_kgs // mass flow through path [kg/s]
F_kgh // [kg/h]
F_scfm // [scfm]
Q_Ls // volume flow through path [L/s]
Q_m3h // [m^3/h]
Q_cfm // [cfm]

```

Example File:

Building Pressurization Test for Test2_BldgPress_Auto.prj

Pressurization:

```

50.0 Pa
0.201 in.H2O

```

Mass flow rate:

```

0.4674 kg/s
1683 kg/h
822.6 scfm

```

Volume flow rate:

```

388.2 L/s
1398 m^3/h
822.6 cfm

```

Ambient conditions:

```

Pressure
101320 Pa
29.920 in.H2O
Temperature
20 C
68 F
Air Density
1.2040 kg/m^3
0.07517 lb/ft^3

```

Volumes:

```

Conditioned zones
445.44 m^3
15730.6 cuft
Ducts & AHS (conditioned)

```

USING CONTAM – WORKING WITH SIMULATION RESULTS

17.11 m³
604.3 cuft
Unconditioned zones
384.89 m³
13592.3 cuft

Airflow Details:

Path	Czone	Uzone	dP [Pa]	[in.H2O]	F [kg/s]	[kg/h]	[scfm]	Q [L/s]	[m ³ /h]	[cfm]
3	4	1	49.95	0.2007	0.01077	38.78	18.95	0.008946	32.21	18.96
4	2	1	49.95	0.2007	0.009482	34.14	16.69	0.007875	28.35	16.69
5	3	1	49.95	0.2007	0.009482	34.14	16.69	0.007875	28.35	16.69
. . .										
205	16	0	50.00	0.2009	0.009616	34.62	16.92	0.007987	28.75	16.92
206	16	0	50.00	0.2009	0.001923	6.924	3.384	0.001597	5.75	3.385

❑ **CONTAMX2.LOG**

The simulation engine will always produce the ASCII log file *CONTAMX2.LOG* within the CONTAM program directory. This file mostly contains simulation performance data typically not relevant to most users. However, if you encounter an error during simulation, this file can be useful to the program developer in tracing the error.

3.21.2 Viewing Results

This section describes how to activate the results display mode of ContamW and to display the results at different time-steps upon the SketchPad of transient simulations.

❑ **Results Display Mode**

To view results using ContamW you activate the results display mode by checking either the **SketchPad Results** or the **Results Display Window** selection of the **View** menu. This will occur automatically after the successful completion of a simulation or after opening a project file that has simulation result files associated with it. Once the results display mode is activated, the status bar will display result-related information (See Status Bar) and you will be able to utilize all of the *Results Display Methods* outlined below.

❑ **Time Step**

If you have performed a transient simulation, results will be available for each time step. However, the SketchPad results (as described in the following section) can only be displayed for a single time-step at a time. You can change the time-step for which results are currently displayed upon the SketchPad using the following menu and keyboard shortcut commands.

Show Next Time Step

Menu Command: **View → Next Time Step**

Keyboard Shortcut: **End**

Show Previous Time Step

Menu Command: **View → Previous Time Step**

Keyboard Shortcut: **Home**

Show Last Time Step (of the day)

Menu Command: **View → Last Time Step**

Keyboard Shortcut: **Ctrl+End**

Show First Time Step (of the day)

Menu Command: **View → First Time Step**

Keyboard Shortcut: **Ctrl+Home**

3.21.3 Results Display Methods

This section describes the various methods that ContamW uses to present simulation results. These methods include SketchPad for airflow paths, a Results Display Window for more detailed information on the icon currently highlighted on the SketchPad, plotting transient results on graphs, shaft reports, and exporting results to various types of external files. You can also print copies of the graphs you view with ContamW or copy them to the Windows clipboard to be pasted into other applications (e.g. word processor applications for written reports).

3.21.4 SketchPad Results

ContamW will display color-coded bars indicating the relative airflow rates and pressure drops associated with each airflow path, duct terminal, or the supplies and returns of simple air-handling systems for the current time-step on the current level of the SketchPad. By default, **Airflow rates are displayed with green lines** and **pressure differences with red lines**. Display colors are user-selectable (See Configuring ContamW). These result-bars will only appear if you have selected the Airflows option on the Output page of the Simulation Parameters property sheet. If you have set flow or pressure limits for a given airflow path (see Airflow Path Properties), and they have been exceeded, the icon will be highlighted in red as a warning indicator.

You can also highlight individual icons on the SketchPad to display results related to the specific icon. Results can be viewed in the ContamW Status Bar and Floating Status Bar (See Status Bar in the Getting Started section of the documentation). The following is a brief explanation of the results displayed for each building component.

- Airflow path – pressure difference, airflow rate and direction of pressure drop and airflow
- Duct segment – pressure difference, airflow rate, velocity and direction of pressure drop and airflow where velocity is calculated from the area of the duct segment
- Simple air handling system – outdoor, recirculation and exhaust airflow rate
- Supply and return of simple air handling system – airflow rate
- Zone – zone temperature and reference pressure, i.e., the absolute pressure of the zone relative to the absolute pressure of the ambient at the elevation of the level on which the zone is located. Prior to CONTAM 2.4, this pressure was relative to the ambient pressure at the reference elevation of the building. So now these pressures more closely resemble a gage pressure.
- Duct junction – junction temperature and static pressure relative to the pressure at the height of the junction in the zone in which the junction is located
- Duct terminal – airflow rate and velocity based on the terminal free face area, temperature and static pressure relative to the pressure at the height of the terminal in the zone in which the terminal is located

3.21.5 Results Display Window

The ContamW Results Display window displays different types of information depending on the type of simulation that was performed (steady state or transient) the type of output that you selected (See Output Properties) and the currently (or most recently) highlighted icon on the SketchPad. The results display window will display information associated with zone icons and exposure icons. Once in the *results* mode, you can control the visibility of the Results Display window using the **Results Display Window** selection of the **View** menu. However, you can only display the Results Display window while the SketchPad results are displayed.

For steady-state simulations the steady-state results will be displayed. For transient simulations, results will be displayed for the current time step. Select the simulation time-step for which you want results to be displayed using the Simulation Time Step commands outlined in the previous section on Viewing Results.

□ Contaminant Concentration Results

If the currently highlighted icon is a zone icon, then contaminant results will be displayed in the Results Display Window. The concentration for each contaminant (species used in the simulation) will be displayed in the units of the default concentration of each contaminant (See Species Properties in the Working with Species and Contaminants section).

□ Zone Airflow Results

If the currently highlighted icon is a zone icon, then for each adjacent zone the sum of the airflows between the highlighted zone the adjacent zone will be displayed in the *Default Units of Flow* set for the project (See Default Units in the Configuring ContamW section).

□ Exposure Results

Summary exposure results are now available in the exposure box whisker (.EBW) file (See Results Files).

3.21.6 Plotting Results

ContamW provides you with some charting features for plotting transient simulation results. You can plot airflow, contaminant and exposure results using these features. Charting is only available for transient simulation results of the currently loaded CONTAM project. You can plot results for airflows and pressure differences, contaminant concentrations, and occupant exposure and potential dosage. You access these charting features from the **Simulation** menu when in the *results* mode.

□ Plotting Airflow Results

Menu Command: **Simulation** → **Plot Airflow Results...**

This command will display the Chart Control dialog box that you use to select and modify the chart settings for plotting airflow and pressure difference results for airflow paths. With this charting tool you can select multiple flow paths; plot one or more days of simulations; select the units of display; and plot either airflow, pressure difference or both on the same plot. You can also plot two-way flow rates for those flow paths that implement two-way airflow elements. If you select not to show two-way flows, ContamW will plot the net flow rate for airflow paths that implement two-way airflow elements.

NOTE: If you select this menu item when the ContamW caret is highlighting an icon having airflow results associated with it (airflow path icon, duct flow icon, duct junction icon, or supply/return icon of a simple air-handling system), it will appear in the list of paths to plot when the Chart Control dialog box is displayed. This feature allows you to plot results for a particular flow path by highlighting it on the SketchPad as opposed to finding it in the list of flow paths on the Chart Control dialog box.

□ Plotting Contaminant Results

Menu Command: **Simulation** → **Plot Contaminant Results...**

This command will display the Chart Control dialog box that you use to select and modify the chart settings for plotting the time-history of contaminant concentrations of zones. With this charting tool you can select which contaminant to plot, plot multiple zones on the same graph, plot one or more days of results, and select the units of display.

NOTE: If you select this menu item when the ContamW caret is highlighting a zone icon, it will appear in the list of paths to plot when the Chart Control dialog box is displayed. This feature

allows you to plot results for a particular zone by highlighting it on the SketchPad as opposed to finding it in the list of zones on the Chart Control dialog box.

□ Plotting Exposure Results

Menu Command: **S**imulation → **P**lot Exposure Results...

This command will display the Chart Control dialog box that you use to select and modify the chart settings for plotting the time-history of occupant exposure to and potential dosage of contaminants. With this charting tool you can select which contaminant to plot; plot multiple occupants on the same graph; plot one or more days of results; select the units of display; and plot either exposure, potential dosage or both on the same plot.

NOTE: If you select this menu item when the ContamW caret is highlighting an exposure icon, it will appear in the list of occupants to plot when the Chart Control dialog box is displayed. This feature allows you to plot results for a particular occupant by highlighting it on the SketchPad as opposed to finding it in the list of occupants on the Chart Control dialog box.

3.21.7 Chart Control: Transient Airflow Results

Use this dialog box to control how transient airflow and pressure results are plotted.

Paths to Plot/Paths Available: The plotted data for each airflow element will appear as a separate trace, the "Paths to Plot" lists the airflow elements that will be graphed. To add an item to this list select the appropriate airflow element type (paths, ducts, or junctions), scroll through the list of available elements or enter the element number directly if known, then press the "← Add" button. To remove an item from the Paths to Plot field select the zone to be deleted and then press the "Remove" button.

Date(s) to Plot: This field allows the you to select the dates of data you wish to plot on the graph. The available dates for the given simulation file are shown for your reference. You must set the dates to fall within this range of dates. Enter dates in the following format:

MMMDD - the first 3 letters of the month and then the day of the month with or without spaces

Data to Plot: You can plot airflow rates, pressure differences, or both. If you plot both airflow and pressure difference, then ContamW will produce a graph with two vertical axis - one for airflow and one for pressure difference. You can select the units of these axis using the "Airflow Units" and "Pressure Units" drop-down combo boxes on this dialog box.

You can also show two-way flows for flow elements that utilize two-way flow models. If you leave "Show two-way flows" unchecked, then ContamW will plot the net flow rate for all flow paths. If you check "Show two-way flows," then a zero component will be plotted for all flow paths that do not implement two-way models.

Comment for chart title: Enter a comment you want to appear at the top of the chart to help distinguish various plots from one another.

3.21.8 Chart Control: Transient Contaminant Results

Use this dialog box to control how transient contaminant results are plotted.

Contaminant to Plot: This lists all the currently defined project contaminants. Select the contaminant for which you want results plotted.

Moisture Units: If the contaminant to plot is "H₂O," then this section will be activated to provide the ability to plot moisture in units of *mass fraction*, *humidity ratio*, or *% relative humidity*. Mass fraction is the same units in which all other contaminants are reported, i.e., mass of contaminant per mass of air. Humidity ratio is in units of mass of water vapor per mass of dry air, i.e., mass of air less the mass of water vapor. %RH will utilize the equations presented in Chapter 6 of ASHRAE 2005 Fundamentals [ASHRAE 2005] to calculate the relative humidity based upon the mass fraction of H₂O, temperature and absolute pressure in each of the *Zones to Plot*. If you select *Mass Fraction* or *Humidity Ratio* you can select from the set of units available in the Units list. When plotting humidity ratio, you should only plot in units of kg/kg, g/kg or lb/lb and not in volumetric units as the conversion will not be correct due to the assumed standard air density in CONTAM's conversion routines.

Zones to Plot/Zones Available: The plotted contaminant data for each zone will appear as a separate trace, the "Zones to Plot" lists the zones that will be graphed. To add an item to this list select the appropriate level and zone name from the drop down boxes in the Zones Available field and then press the "← Add" button. To remove an item from the Zones to Plot field first select the zone to be deleted and then press the "Remove" button.

Date(s) to Plot: This field allows the user to plot the contaminant results from a specified date range within the simulations results file. The available dates for the given simulation file are given, the user provided date range must fall between these dates. The dates are entered in the form: the first 3 letters of the month and then the day of the month with or without spaces.

Comment for chart title: Enter a comment you want to appear at the top of the chart to help distinguish various plots from one another.

3.21.9 Chart Control: Transient Exposure Results

Use this dialog box to control how transient exposure results are plotted.

Occupant to Plot: This is a list of the occupants for which you want to plot exposure results. Use the "Occupants Available" section to select occupants to add to this list and click the "← Add" button or highlight an occupant in this list and remove it using the "Remove" button.

Contaminant to Plot: This lists all the currently defined project contaminants. Select the contaminant for which you want results plotted.

Date(s) to Plot: This field allows the user to plot the contaminant results from a specified date range within the simulations results file. The available dates for the given simulation file are given, the user provided date range must fall between these dates. The dates are entered in the form: the first 3 letters of the month and then the day of the month with or without spaces.

Comment for chart title: Enter a comment you want to appear at the top of the chart to help distinguish various plots from one another.

3.21.10 Exporting Results

ContamW provides you with the ability to export results to external files. You can either generate *report files* or *export files*. *Report files* are text files that are formatted for ease of reading. *Export files* are *tab-delimited*, so you can easily import them into a spreadsheet. This feature provides you with greater charting flexibility as well as the ability to perform more sophisticated data analysis of your simulation results. You access this feature of ContamW by

selecting **Export Results...** from the **Simulation** menu or use the **F5** shortcut key when the results-display mode is active. This will display the "Export Data" dialog box as described in the following section.

3.21.11 Export Data

This section provides detailed descriptions of the specific properties associated with generating result reports and exporting results to external files.

Transient Results:

Results available from/to: This shows the range of dates/times for which results are available in the simulation results file (.SIM) of the latest simulation.

Export transient results from/to: Select the range for the data to be exported. Enter the date as the first 3 letters of the month followed by the day of the month with or without spaces, e.g., Jan01, followed by the time in the following format HH:MM:SS.

Airflow Data: Options for reporting/exporting airflow data. Airflow rates will be reported in the *Default Units of Flow* as selected on the Default Units page of the Project Configuration properties (See Configuring CONTAMW).

Report airflows of all paths for date/time: Create a text file formatted for ease of reading that lists the airflows and pressure differences for all paths, zone pressures and temperatures at the selected date and time.

Export airflows of all zones for date/time: Create a tab-delimited file containing a table of the total airflows between each zone of the building. The leftmost column lists the level and name of the zones from which airflows and the topmost row lists the level and names of the zones into which air flows. Therefore, all airflows are listed as positive values.

Export transient flow for airflow path number: Create a tab-delimited file listing all the airflow data for a specific airflow path. Select the airflow path number(s) for which you want to output the airflow and pressure drop data for the range of time you select in the "Export transient results from/to" dates and times indicated above.

Export average transient flow for each zone between export result times: Create a tab-delimited file listing the average airflow rates between zones averaged over the range of time you select in the "Export transient results from/to" dates and times indicated above.

Contaminant Concentration Data: Options for reporting and exporting contaminant results. Concentrations will be reported in the units associated with the default concentration of each species/contaminant (See Species Properties).

Report transient concentrations of all zones for date/time: Create a text file formatted for ease of reading that lists the contaminant concentrations for all zones and contaminants at the selected date and time.

Report transient concentrations of all zones: Create a text file formatted for ease of reading that lists the contaminant concentrations for all zones at the selected date and time.

Export transient concentrations of all zones: Create a tab-delimited file listing the transient contaminant concentrations for all zones and contaminants for the range of time you select in the "Export transient results from/to" dates and times indicated above.

Export transient concentrations for zone: Create a tab-delimited file listing the transient contaminant concentrations for each contaminant in the selected zone for the range of time you select in the "Export transient results from/to" dates and times indicated above.

3.21.12 Shaft Report

CONTAM is well suited for the analysis of smoke control systems. Because of the importance of shafts for some smoke control systems [Klote and Milke 2002], ContamW provides a special reporting feature for shafts.

□ Defining Shafts

You define shafts during the drawing phase of project development by placing zone icons directly below one another (in the same column and row) on adjacent levels of a building. If there is a zone icon in the same position on the level above or below the current level, those zones are part of the shaft when ContamW generates a shaft report. The shaft consists of zones that exist on contiguous levels having a zone icon in the same location on the SketchPad.

You create the shaft report by selecting **Generate Shaft Report** from the **Simulation** menu. This will display a dialog box prompting you to execute the following three steps to define the shaft zone icon and primary and secondary flow paths associated with the shaft across which you want ContamW to report the airflows and pressure drops. The zone icon would typically represent a stairwell, and the primary flow path would typically represent a stairwell door.

Steps to select a shaft for a shaft report:

- 1) Left-click on the zone icon that defines the shaft
- 2) Left-click on the primary airflow path icon
- 3) Left-click on the secondary airflow path icon

NOTE: Click the right mouse button to cancel the shaft report generation process.

Once you have completed the above steps, the shaft report will be displayed in a dialog box. The report will display pressure drop, airflow rates and direction of airflow on every level of the shaft where there are airflow path icons in the same location on the SketchPad as that of the primary and secondary airflow paths. You can use the Shaft Report dialog box to select the units in which the airflow rate and pressure drops are reported, and you can save a copy of the report to a text file.

3.21.13 TRNSYS Input Files

You can use ContamW to generate input data for the TRNSYS airflow module TYPE96. Use the **Create TRNSYS Input File** selection of the **Simulation** menu. The data will be written to a file having the same name as the current project but the .PRJ extension will be replaced and .AIR extension. The following CONTAM airflow elements are currently supported when creating TRNSYS input files. If any other elements exist within the current project, you will receive a warning message from ContamW that unsupported airflow elements exist within the project.

Powerlaw Models:

$$Q = C(\Delta P)^n$$

$$F = C(\Delta P)^n$$

Leakage Area Data

Connection (ASCOS) Data

Orifice Area Data

Crack Description

Test Data (1-point)

Test Data (2-points)

Stairwell

Shaft

Two-way flow model:

Two-opening Model

In addition, the supply and return paths associated with the simple air handling system model are also included. The AHS implicit paths – recirculation, exhaust, and outside air – are not included. All the CONTAM building zones, AHS supply and return nodes, and the ambient zone are included. A complete list of the inputs to and the outputs from the TYPE96 subroutine are listed at the end of the .AIR file.

The inputs from TRNSYS to TYPE96 are:

Temperatures for all zones including ambient [C]

Absolute barometric pressure [Pa]

Wind speed at the building reference elevation [m/s]

Wind direction [degrees clockwise from north]

All AHS supply and return flows [kg/s]

Any other specified mass flow rates [kg/s]

The relative areas of any variable openings

All flow and relative area inputs must be greater than zero.

The outputs from the TYPE96 subroutine to TRNSYS are the interzone air flows including flows to and from ambient. There are a maximum of two flows reported between any two zones – the sum of all flows from zone 1 to zone 2 and the sum of all flows from 2 to 1. There may be multiple paths between any two zones.

□ **TRNSYS Input File Format:**

I indicates an integer value and R indicates a four-byte real number

HEADER DATA:

```
Line 1:  debug control
         I LIST    - non-zero value initiates data dumps.
Line 2:  counts
         I NWPP    - number of wind pressure profiles.
         I NZONE   - number of zones/simultaneous equations.
         I NPATH   - number of airflow paths.
         I NELMT   - number of airflow elements.
Line 3:  run control
         I ITMAX   - maximum number of iterations.
         R RELAX   - under- relaxation coefficient.
         R ACNVG1  - relative airflow convergence factor.
         R ACNVG2  - absolute airflow convergence factor.
         I TADJ    - non-zero to do temperature adjustment.
Line 4:  weather data
         R PBAR    - absolute barometric pressure [Pa].
         R WSPD    - wind speed [m/s].
         R WDIR    - wind direction [deg clockwise from north].
Line 5:  end of section indicator
         I J       - should equal -999
```

WIND PRESSURE PROFILES:

```
One line for each wind pressure profile consisting of
         I NR      - profile number, starting at 1.
         I TYPE    - profile type.
         I NPTS    - number of data points.
         followed by one line with 5 real values for each data point.
Last line: end of section indicator
         I J       - should equal -999
```

AIR FLOW ELEMENTS:

```
One line with 3 real values for each flow element.
Last line: end of section indicator
         I J       - should equal -999
```

ZONE DATA:

```
One line for each zone consisting of
         I NR      - zone number, starting at 1.
         I CP      - 1 indicates constant pressure zone.
         R TZ      - initial zone temperature [K].
         R PZ      - initial zone pressure [Pa].
         R VOL     - zone volume [m^3].
         R HT      - zone elevation [m].
Last line: end of section indicator
         I J       - should equal -999
```

PATH DATA:

```
One line for each airflow path consisting of
         I NR      - path number, starting at 1.
         I NZ      - number of zone air flows from (to be positive).
         I MZ      - number of zone air flow to.
```

USING CONTAM – WORKING WITH SIMULATION RESULTS

R HT - path elevation [m].
I TYPE - path type: 0=flow, 1 = PLR, 2=QCN, 3=FCN.
I NE - number of the air flow element.
R MULT - path multiplier.
I XIN - index (>0) of XIN input value (type 0).
I PWP - index (>0) of the wind pressure profile.
R WIND - wind pressure modifier.
R AZM - surface azimuth angle.
R CTRL - flow or control value depending on type.
Last line: end of section indicator
I J - should equal -999
INPUT & OUTPUT DATA:
First line:
I NIN - number of input variables.
I NOUT - number of output variables.
One line for each output variable consisting of
I NR - output number (index to XOUT).
I NZ - number of zone the air flows from.
I MZ - number of zone the air flows to.
Last line: end of section indicator
I J - should equal -999

3.22 Working with Project Annotations

Project Annotations and the *Project Description* provide you with simple project documentation features.

□ Annotations

You simply place an annotation icon onto the SketchPad (see Drawing Building Component Icons) and then use the icon definition procedure (see Defining Building Component Icons) to edit the text for the annotation. Once you have defined the text associated with the annotation icon, the text will appear in the status bar whenever you highlight the icon with the caret. You can edit, delete, and move the icon as you would any other building component icon.

□ Project Description

The project file description is provided to help you manage your project files. This description will be available when you are opening project files from within ContamW (see Working with Project Files).

3.23 Getting Help

This help manual is available in both printed form and accompanying the ContamW program as "on-line" help. There are several ways to access the on-line help system. You can either use the **Help Contents...** selection of the **Help** menu to display the contents of the help manual or use the context-sensitive feature to access help.

□ Help Contents

You can browse the contents or index of the help system to view the topic of your choice and use the search feature to display a list of help topics that contain the keyword in which you are interested.

□ Context-Sensitive Help

The context-sensitive help feature is designed to provide you with specific information related to the currently active ContamW window or dialog box. To activate the context-sensitive help system, press the **F1** key.

□ Help Index and Search

Be sure to use the on-line help index and search features if you are having trouble locating a specific topic.

□ Obtaining Technical Support

If you need to contact the CONTAM developers for guidance on using the program or to report a technical problem with the program you can do so via email at the addresses below. If you encounter an error while working with CONTAM, and you require assistance solving the problem, you should immediately (before running ContamW or ContamX again) make copies of the CONTAMW2.LOG, CONTAMX2.LOG, PROJECT.BKP and WEATHER.BKP files located in the CONTAM program directory. You can email these files to NIST so that we may be better equipped to address your particular problem.

NIST Contact Information

- | | | |
|------------------------------|------------------|--------------|
| 1. George Walton | gwalton@nist.gov | 301-975-6421 |
| 2. W. Stuart Dols | wsdols@nist.gov | 301-975-5860 |
| 3. IAQ and Ventilation Group | | 301-975-6431 |

4 Special Applications of CONTAM

This section includes instructions, or considerations, for using CONTAM to simulate several different building features (e.g. atria, stairwells and chimneys) and perform special building analysis such as fan pressurization. There are also available, several references that present the application of CONTAM in various studies [Emmerich and Persily 1996 and 1998; Emmerich and Nabinger 2000; Fang and Persily 1995; Musser and Yuill 1999; Musser 2000; Persily 1998; Persily and Martin 2000; Persily and Ivy 2001]. Also, check the bibliography provided on the NIST IAQ Analysis website <http://www.bfrl.nist.gov/IAQanalysis/pubs.htm>. Some documents are available directly from the website.

4.1 Building Pressurization Test

The building pressurization test (sometimes referred to as fan pressurization or blower door test) is a common way of determining the airtightness of a building envelope. Fan pressurization is relatively quick and inexpensive and it characterizes building envelope airtightness independent of weather conditions. In this procedure, a large fan or blower is mounted in a door or window and induces a large and roughly uniform pressure difference across the building shell [ASHRAE 2005, p 27.12; ASTM 1999]. A common airtightness rating is the airflow rate at 50 Pa divided by the building volume to give units of air changes per hour. CONTAM now provides an automated method of performing the *Building Pressurization Test* (See Working with Simulations), but there are also “manual” methods of performing this test presented below for informational purposes.

You may wish to do a simulated pressurization test to determine the airtightness rating of your idealized building. This is easily done by setting one of the zones on the zero elevation level to a constant pressure of 50 Pa, running the simulation, and then observing the total airflow out of that zone. (Don't forget flows to other building levels.) You could also define an airflow path using a constant volume airflow element through which you can vary the airflow until you achieve the desired level(s) of pressurization. Use standard m^3/h flow units and divide by the building volume to get the air change rate at 50 Pa. Alternatively, the flow rate at 4 Pa can be converted to an effective leakage area (ELA) using equations in Chapter 27 of the 2005 ASHRAE Fundamentals Handbook [ASHRAE 2005]. Use the CONTAM results display to check that the building has been properly pressurized by moving the cursor to each path on the envelope to get the pressure drop reported in the status bar. You can use the report of all flows to get a listing of all flow and pressure drops. The pressure difference (dP) across every path to the ambient zone should be close to -50 Pa. The value is negative, because the report considers flow and dP to the zone listed in the “zone” column from the zone in the “from” to be positive. All flows are to the “from” zone.

The reported pressure drop across various flow paths will not be exactly 50 Pa because of density effects. Minimize these differences by running all zones and ambient at the same (standard) temperature and no wind. Differences in pressure due to elevation make it critical that the constant pressure zone be on the level with zero elevation.

In a multizone building the effects of partitioning (internal walls and floors) usually are significant. There are two solutions depending on the goal of the simulation. If the goal is only to achieve a uniform pressure difference across the entire building shell, add a very large opening across each partition. If the goal is to reproduce a field test, simulate the internal doors as being open to get a relatively uniform pressure inside the building.

Prior to performing these tests, you should make a copy of the project file, so you do not have to reverse the changes to restore the original project file.

4.2 Smoke Control Systems

For a stairwell pressurization system, you may want to record the pressurization flows by creating a simple AHS model with supply points in the shaft. Make these supply points the secondary path. You may use supply point icons aligned on every level with some or most of the flows set to zero, or you may use icons only on those levels with a pressurization flow.

Alternatively, for shafts on an exterior wall of the building, you may wish to use the constant flow rate path elements through the exterior wall to provide the pressurization air. This type of flow path will also report the pressure difference between the shaft and ambient. You may want to align these constant flow paths above the open doorway on the first level. For details on viewing shaft reports see Working with Simulation Results in the Using CONTAMW section of the manual.

4.3 Shafts

Buildings contain several architectural features which offer very low resistance to airflow between building levels. These features include atria, elevator shafts, and stairwells. CONTAM provides two ways of modeling such a low resistance path: the phantom zone and the low-resistance (or large) opening. Select the most appropriate model based on how this path interacts with the rest of the model.

This interaction can be understood by using three flow resistances in series to model a shaft between two levels. There is a resistance to flow through the shaft and resistances representing the paths (doors) connecting the shaft to the two different levels. Using the powerlaw relationship, $Q = C(\Delta P)^{1/2}$, and assuming $C = 3.10$ for the shaft, 1.55 for an open door, and 0.01 for a closed door, the following equivalent flow coefficients, $C_e = (\sum 1/C_i^2)^{-1/2}$, are computed for four different assumptions:

- (1) both doors closed, ignore shaft resistance: $C_e = 0.007070$
- (2) both doors closed, include shaft resistance: $C_e = 0.007066$
- (3) both doors open, ignore shaft resistance: $C_e = 1.096$
- (4) both doors open, include shaft resistance: $C_e = 0.984$

Cases (1) and (2) show that when flow resistance is dominated by the closed doors, it is not necessary to include the shaft flow resistance. Cases (3) and (4) show that when the resistances at the (open) doors is of the same order as the shaft resistance, it is important to include the shaft resistance in the model. For a quick estimate of relative flow coefficients, remember that the flow coefficient is usually roughly proportional to the opening area.

4.3.1 Atria

If an atrium were modeled as individual zones on each level, the inter-level flow resistances would be very low. A cursory comparison of the inter-level opening areas to the sum of all openings from the atrium to other zones will usually indicate that an atrium should be modeled using phantom zones. This is done by placing a standard zone icon at the atrium's lowest level. You use this zone icon to define the atrium temperature and total volume. You should ignore the floor area of the normal zone icon that is used to define the atrium zone. Place a phantom zone icon at all higher levels which contain parts of the atrium. A phantom zone icon indicates that

the region within the walls on this level are actually part of the zone on the level below the icon. If it is necessary to have different temperature or contaminant concentrations at the different levels in an atrium, then those levels must be modeled as normal zones connected by flow paths with very large openings.

4.3.2 Stairwells

Data to describe the airflow characteristics of a stairwell in terms of an equivalent orifice is provided in [Achakji and Tamura 1988]. This representation is very well suited to being implemented by CONTAM, which allows you to create a powerlaw flow element from the physical characteristics of the stairwell. Stairwell characteristics include the cross-sectional (horizontal) area of the stairwell shaft, whether the fronts of the stair treads are open or closed, and the number of people on the stairs. This last item might be important in an evacuation scenario. Since the equivalent orifice area of the shaft is on the same order as its cross-sectional area, and this area is likely to be smaller than the sum of the areas of the doors entering the stairwell, it is generally best to model a stairwell as individual zones on each level connected by openings which have been defined using the CONTAM stairwell airflow element that implements the Achakji/Tamura model.

4.3.3 Elevator Shafts

Elevator shafts (hoistways) will generally lie in some modeling regime that falls between stairwells and atria. Measured data for the flow resistance of an elevator shaft have not yet been identified. CONTAM provides a powerlaw flow element based on flow resistance from the Darcy/ Colebrook model of a conduit or duct. A significant difference between an elevator shaft and a stairwell is that the door between the shaft and the building is never fully open because, when it is open, air must flow through openings in the elevator car to pass between the building and the shaft. Table 6.1 of [Klote and Milke 2002] presents orifice areas for closed elevator doors which tend to be fairly small compared to the cross-sectional area of the shaft, especially for hoistways with multiple cars, so it is likely that the airflow in most elevator shafts can be modeled with sufficient accuracy by phantom zones. However, it may be necessary to split the shaft into several sections to achieve sufficient accuracy in modeling contaminant transport.

4.3.4 Chimneys

You can use a duct to create a simple chimney model. At its simplest this will involve only a single duct connecting the inside of the building to the outside with the Darcy-Colebrook duct element used to model the flow resistance of the chimney. The height of the duct terminal points must reflect the height of the chimney and a special high temperature zone must be created around the inside terminal point so that the proper stack effect will result. Be sure to allow an appropriate low resistance path from this special zone to the rest of the building.

5 Theoretical Background

This section provides theoretical background of the CONTAM program. There are basically three subsections: the first provides a summary of underlying assumptions of the model and the others address the contaminant and airflow analysis respectively.

5.1 Model Assumptions

CONTAM is a powerful tool that models airflow and contaminant dispersal in buildings. It is important to realize that this tool implements mathematical relationships to model airflow and contaminant related phenomenon and therefore incorporates assumptions that simplify the model from that of the modeled phenomenon. The following is a brief description of these modeling assumptions.

Well-mixed zones – This assumption refers to the treatment of each zone as a single node, wherein the air has uniform (well-mixed) conditions throughout. These conditions include temperature, pressure (which does vary hydrostatically) and contaminant concentrations. Therefore, localized effects within a given zone cannot be accounted for using CONTAM. For example, if you utilize a contaminant source that introduces a mass of contaminant into a zone at a certain time (burst source), the contaminant will be diluted to the entire volume of the zone within a single time step.

One-Dimensional Convection/Diffusion Zones – In versions prior to CONTAM 2.4 all zones were considered to be *well-mixed*. However, beginning with version 2.4, zones can be preconfigured by the user to be one-dimensional convection/diffusion zones in which contaminants can be allowed to vary along a user-defined axis. When operating under the newly added *short time step method* ContamX will provide for the ability of contaminant concentration gradients to occur in the direction of the convection/diffusion axis by programmatically subdividing the zone into a series of *well-mixed* cells along the axis.

Duct Systems – Typically, during contaminant simulation, there are similarities between duct junctions and well-mixed zones and between duct segments and airflow paths. In this case the volumes of the duct junctions are determined from the duct segments to which they are connected. However, in CONTAM 2.4 the entire duct system can modeled to account for one-dimensional convection/diffusion flow through the system. This feature is available as an option under the new *short time step* contaminant simulation method.

Conservation of mass – When performing a steady-state simulation, the mass of air within each zone is conserved by the model. This implies that air can neither be created nor destroyed within a zone. However, when performing a transient simulation, CONTAM now provides the option of allowing the accumulation or reduction of mass within a zone due to the variation of zone density/pressure and the implementation of non-trace contaminants within a simulation. This is further addressed in this Theory section (See Airflow Analysis and Contaminant Analysis), as well as in a previous section addressing simulation settings (See Airflow Numerics Properties).

Trace contaminants – Trace contaminants are those that are found in low enough levels that they do not affect the density of air within a zone. You must be careful not to rely on the model to handle contaminant concentrations that would cause a change in the density of air. The program will allow for contaminants to reach levels that would, in actuality, affect the density, but the program will still treat them as if they were trace contaminants.

Non-trace contaminants – Non-trace contaminants are those that are present in such quantities that they can influence the air density, e.g., water vapor. In CONTAM you define those contaminants that are components of the air within the building. CONTAM will treat them as a mixture of ideal gases.

Thermal effects – The model does not handle heat transfer phenomenon per se, but does provide for the scheduling of zone temperatures. Zone temperatures can be either constant or allowed to change during transient simulations according to user-defined temperature schedules. CONTAM will determine airflows and non-trace contaminant mass fractions induced by temperature differences between zones including ambient (e.g., as caused by the stack effect). You can also vary the outdoor temperature for transient simulations using weather files.

Airflow paths – Airflow through various airflow elements provided by CONTAM is modeled using either a powerlaw or quadratic relationship between airflow and pressure difference across the flow path. These relationships are models themselves, and care should be taken when implementing them to represent building features within your idealized buildings. See Airflow Elements for detailed explanations of these models.

Source/sink models – CONTAM provides several different source/sink elements or representations of contaminant generation/removal processes. These elements are based upon models found throughout the literature. You should be sure to utilize models that are appropriate for the contaminant source/sink that you want to represent. See Contaminant Source/Sink Elements for detailed explanations of these models.

The previous assumptions relate to the mathematical representations utilized by CONTAM in performing analysis. These assumptions should be distinguished from assumptions made by you, the user, when creating a model of a building referred to as a building idealization (See User Tasks in the Getting Started section). Engineering judgment is required on your part to insure that your building representation is adequate for the purposes of your analysis.

5.2 Contaminant Analysis

The CONTAM contaminant dispersal model is an implementation of Axley's methods [Axley 1987 and 1988]. He states: "The central concern of indoor air quality analysis is the prediction of airborne contaminant dispersal in buildings. Airborne contaminants disperse throughout buildings in a complex manner that depends on the nature of air movements in-to, out-of, and within the building system; the influence of the heating, ventilating, and air-conditioning (HVAC) systems; the possibility of removal, by filtration, or contribution, by generation, of contaminants; and the possibility of chemical reaction, radio-chemical decay, settling, or sorption of contaminants. In indoor air quality analysis we seek to comprehensively model all of these phenomena."

The basis for contaminant dispersal analysis is the application of conservation of mass for all species in a control volume (c.v.). A c.v. is a volume of air which may correspond to a single room, a portion of a room, or several well-coupled rooms (a CONTAM *zone*) or the ductwork (where a *junction*, under the well-mixed assumption, has half the volume of each of the adjacent duct segments). The representation of building spaces as CONTAM zones is a matter of engineering judgment.

CONTAM 2.4 adds the capability to model convection-diffusion in the ductwork and user-selected zones instead of treating them as *well-mixed* control volumes. In CONTAM this is done within the new *short time step method*.

5.2.1 Properties of Air

In CONTAM air is treated as an ideal gas with properties computed from the ideal gas law. The density of air is given by

$$\rho = m/V = P/RT \quad (1)$$

where

- m = the mass of air in
- V = a given volume,
- P = the absolute pressure,
- R = the gas constant for air, and
- T = the absolute temperature.

The mass of air in c.v. i is the sum of the masses of the individual contaminants, α , in the c.v.

$$m_i = \sum_{\alpha} m_i^{\alpha} \quad (2)$$

The concentration of contaminant α in c.v. i is defined as

$$C_i^{\alpha} = m_i^{\alpha} / m_i \quad (3)$$

In CONTAM *concentration* refers to a mass ratio rather than a volumetric ratio unless otherwise specified.

Air is a mixture of several different species. The value of the gas constant for the air in a c.v. is given by:

$$R_i = \sum_{\alpha} R^{\alpha} C_i^{\alpha} \quad (4)$$

where R^α = the gas constant of species α which equals the universal gas constant, 8 314.41 J/(kmol·K), divided by the molar mass of α (kg/kmol). Similarly, for thermal calculations (an option in CONTAM for duct flow under short-time-step method only), the specific heat of air in a duct segment is given by the weighted sum of the specific heats of the individual species:

$$Cp = \sum_{\alpha} Cp^{\alpha} C_i^{\alpha} \quad (5)$$

Under typical conditions only water vapor has an impact on the properties of air and even that can be ignored as an initial approximation. There is a standard definition of species concentrations for *dry air* which yields an effective molar mass of 28.9645 kg/kmol and a gas constant of 287.055 J/(kg·K). ASHRAE often refers to dry air at *standard conditions* which are 101.325 kPa and 20 °C and notes the density of such air is 1.20 kg/m³ [ASHRAE 2004 p 18.4]. More precisely, the density is 1.20410 kg/m³ as computed by equation (1).

NOTE: ASHRAE considers water vapor in terms of *humidity ratio* instead of mass concentration. The humidity ratio, W , is defined as the ratio of the mass of water vapor to the mass of dry air in the volume: $W = m_w / m_{da}$ [ASHRAE 2005 p 6.8]. The CONTAM mass concentration, which ASHRAE refers to as *specific humidity*, is: $C_w = m_w / (m_w + m_{da})$. The conversions between humidity ratio and mass concentration are $C_w = W / (1+W)$ and $W = C_w / (1-C_w)$.

In many cases we are interested in species concentrations that are too small to significantly affect the air density (or specific heat). These are referred to as *trace concentrations*. When a simulation involves only trace contaminants, CONTAM uses dry air to compute the air properties.

5.2.2 Contaminant Concentrations

Within CONTAM a contaminant may be added to c.v. i by:

- inward airflows through one or more paths at the rate $\sum_j F_{j \rightarrow i} (1 - \eta_j^\alpha) C_j^\alpha$ where $F_{j \rightarrow i}$ is the rate of air mass flow from c.v. j to c.v. i and η_j^α is the filter efficiency in the path, and
- species generation at the rate G_i^α .

A species may be removed from the c.v. by:

- outward airflows from the zone at a rate of $\sum_j F_{i \rightarrow j} C_i^\alpha$ where $F_{i \rightarrow j}$ is the rate of air mass flow from c.v. i to c.v. j , and
- species removal at the rate $R_{\alpha,i} C_{\alpha,i}$ where $R_{\alpha,i}$ is a removal coefficient.

A species may be added *or* removed by first-order chemical reactions with other species at the rate $\sum_{\beta} \kappa^{\alpha,\beta} m_i^\beta$ where $\kappa^{\alpha,\beta}$ is the kinetic reaction coefficient in c.v. i between species α and β .

(Sign convention: positive κ for generation and negative κ for removal). This linear expression currently limits the kinds of reactions that can be modeled.

Combining these processes into a single equation for the rate of mass gain of species α in c.v. i gives:

$$\frac{dm_i^\alpha}{dt} = \sum_j F_{j \rightarrow i} (1 - \eta_j^\alpha) C_j^\alpha + G_i^\alpha + m_i \sum_\beta \kappa^{\alpha, \beta} C_i^\beta - \sum_j F_{i \rightarrow j} C_i^\alpha - R_i^\alpha C_i^\alpha \quad (6)$$

The transient conservation of species mass in a control volume is given by:

$$\begin{aligned} & \text{(mass of contaminant } \alpha \text{ in c.v. } i \text{ at time } t + \Delta t) = \\ & \text{(mass contaminant } \alpha \text{ in c.v. } i \text{ at time } t) + \\ & \Delta t \times \text{(rate gain of contaminant } \alpha \text{ – rate loss of contaminant } \alpha) \end{aligned}$$

Or in equation form as:

$$\rho_i V_i C_i^\alpha \Big|_{t+\Delta t} \approx \rho_i V_i C_i^\alpha \Big|_t + \Delta t \cdot \left[\sum_j F_{j \rightarrow i} (1 - \eta_j^\alpha) C_j^\alpha + G_i^\alpha + m_i \sum_\beta \kappa^{\alpha, \beta} C_i^\beta - \sum_j F_{i \rightarrow j} C_i^\alpha - R_i^\alpha C_i^\alpha \right]_{t+\delta t} \quad (7)$$

5.2.3 Numerical Calculation of Contaminant Concentrations

Several possible solutions for equation (7) can be characterized by the choice of δt to determine the rate of gain or loss. CONTAM has traditionally chosen $\delta t = \Delta t$. Equation (7) becomes:

$$\left[\rho_i V_i + \Delta t \cdot \left(\sum_j F_{i \rightarrow j} + R_i^\alpha \right) \right] C_i^\alpha \Big|_{t+\Delta t} \approx \rho_i V_i C_i^\alpha \Big|_t + \Delta t \cdot \left[\sum_j F_{j \rightarrow i} (1 - \eta_j^\alpha) C_j^\alpha + G_i^\alpha + m_i \sum_\beta \kappa^{\alpha, \beta} C_i^\beta \right]_{t+\Delta t} \quad (8)$$

All concentrations C_i^α at time $t + \Delta t$ are functions of various other concentrations also at $t + \Delta t$.

This is the *standard implicit method*, and it requires that a full set of equations (8) must be solved simultaneously.

The number of equations, N, equals the number of species times the number of control volumes. In a traditional Gauss elimination (or LU decomposition) solution the computation time is proportional to N^3 , making it impractical for large problems. CONTAM offers three solution methods which take advantage of matrix sparsity to handle cases with large numbers of equations. These are a direct skyline algorithm, an iterative biconjugate gradient (BCG) algorithm, and an iterative successive over-relaxation (SOR) algorithm. (LU decomposition is provided only for testing and benchmarking.) The skyline algorithm is very fast for problems of intermediate size but can be slow for large problems. The SOR algorithm requires much less memory and may be faster for large problems unless there are convergence difficulties. In such cases try the BCG solution, although it may also experience convergence difficulties. It can be useful to test the different methods to determine which will give optimum performance before doing a long transient simulation.

A more accurate solution can be obtained by choosing $\delta t = \Delta t/2$ which means average conditions during the time step. This has been implemented in CONTAM by a trapezoidal integration which still requires solving the full set of simultaneous equations.

We can also choose $\delta t = 0$. In this case equation (7) becomes:

$$\rho_i V_i C_i^\alpha \Big|_{t+\Delta t} \approx \rho_i V_i C_i^\alpha \Big|_t + \Delta t \cdot \left[\sum_j F_{j \rightarrow i} (1 - \eta_j^\alpha) C_j^\alpha + G_i^\alpha + m_i \sum_\beta \kappa^{\alpha, \beta} C_i^\beta - \sum_j F_{i \rightarrow j} C_i^\alpha - R_i^\alpha C_i^\alpha \right]_t \quad (9)$$

Every concentration C_i^α at time $t + \Delta t$ is a function of various other known concentrations at time

t. This is the *standard explicit method* which has the tremendous computational advantage of not requiring the solution of simultaneous equations. That advantage is offset by instability under some conditions. That is, the concentrations at successive time steps may diverge wildly from the analytically correct solution. Stability is determined by the magnitudes of the coefficients in equations (7). For example, when the sum of the flows into or out of the c.v. in one time step is greater than the mass of air in the c.v., the solution becomes unstable, that is, values at successive time steps begin to oscillate around the true solution and eventually reach impossible values. Instability will also result when $\kappa^{\alpha,\beta} \cdot \Delta t$ or $R_i^{\alpha} \cdot \Delta t$ exceed a certain magnitude. The standard implicit method is stable at all time steps.

The stability question is so important that it is useful to review the time scales that can be expected in the normal operation of buildings. ASHRAE indicates that the air exchange rate to condition and ventilate rooms should be less than 12 air changes per hour in nearly all commercial applications. [ASHRAE 1999b, p. 3.2 Table 1 - General Design Criteria] This corresponds to a stability limit for the explicit method of 1/12 h, or 5 min. In CONTAM the volume of a junction is half the sum of the volumes of the duct segments meeting at the junction. A junction between two 2 m long duct segments where the air is flowing at a velocity of 2 m/s would have a stability limit of 1 second. Smaller limits are likely to occur because of shorter duct segments or higher velocities.

In early versions of CONTAM it was anticipated that the processes being modeled would require time steps down to about 5 minutes. The very short stability limit for modeling the ductwork drove CONTAM to use an implicit solution where the execution time for relatively few time steps was less than doing many shorter times steps using the explicit solution. The recent addition of control system modeling and the need to track quick contaminant releases both require time steps no longer than a few seconds. We will therefore reconsider use of the explicit model for the ducts.

The following figure shows a CONTAM sketchpad representation of the typical features of a very simple building and its air handling system. The duct icons indicate the normal direction of flow. The ductwork consists of a return duct with terminals (R) in each room, an exhaust to ambient (X), a path for recirculation, an outdoor air intake (OA), and a supply duct with terminals (S) in each room.

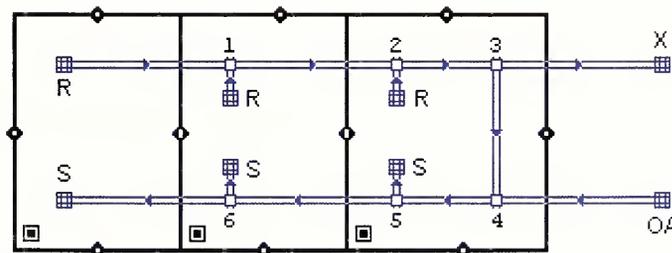


Figure – Schematic of a basic air handling system

When the simulation time step is sufficiently short, the *explicit method* can be used to compute the contaminant concentrations in the zones. That same time step can be used with the *implicit method* to compute the concentrations in the duct junctions if the calculations are performed in the proper sequence where equation (8) is solved one node at a time following the direction of

airflow. The following discussion will show how the duct junction concentrations can be solved implicitly, thereby avoiding stability problems, and without solving simultaneous equations, thereby producing a fast computation.

The process begins by computing all airflows based on conditions known at time t . The contaminant calculation sequence then begins by using a modification of equation (9) to compute the concentrations C_i^α at time $t+\Delta t$ in all zones:

$$(\rho_i V_i + \Delta t \cdot \sum_d F_{i \rightarrow d}) C_i^\alpha \Big|_{t+\Delta t} \approx \rho_i V_i C_i^\alpha \Big|_t + \Delta t \cdot \left[\sum_j F_{j \rightarrow i} (1 - \eta_j^\alpha) C_j^\alpha + G_i^\alpha + m_i \sum_\beta \kappa^{\alpha, \beta} C_i^\beta - \sum_j F_{i \rightarrow j} C_i^\alpha - R_i^\alpha C_i^\alpha \right] \quad (10)$$

In equation (10) the flows out of the zone are divided into two parts

- $\sum_j F_{i \rightarrow j}$, flows to other zones, and
- $\sum_d F_{i \rightarrow d}$, flows into duct junctions through return terminals or leaks,

to conserve contaminant mass when computing concentrations at the return terminals.

The concentrations in the return terminals (R and OA) are then computed with a simplified form of equation (8), because sources, sinks, and reactions are not modeled in ducts, although these features could be added in the future.

$$\left[\rho_i V_i + \sum_j F_{i \rightarrow j} + R_i^\alpha \right] C_i^\alpha \Big|_{t+\Delta t} \approx \rho_i V_i C_i^\alpha \Big|_t + \Delta t \cdot \left[\sum_j F_{j \rightarrow i} (1 - \eta_j^\alpha) C_j^\alpha \right] \quad (11)$$

This calculation is particularly simple for terminals because there is only one airflow into each return terminal and an equal airflow out into the duct network. The concentration in the flow into the terminal, C_i^α at time $t+\Delta t$, comes from equation (10) for the surrounding zone, j . Once all return terminal concentrations have been computed the upstream concentrations are known for both ducts meeting at the left-most junction (# 1) in the return duct, and equation (11) can be solved directly because all concentrations on the right side of the equation are known. Once the junction #1 concentrations have been computed all upstream concentrations are known for the next junction to the right (# 2) along the return duct and its concentrations can then be computed. This process continues around the ductwork until the concentrations at all junctions and all supply terminals have been computed. This implicit calculation of the concentrations in the duct junctions is unconditionally stable, so overall stability is determined by conditions in the zones.

CONTAM 2.4 adds this process calling it the *short time step method* (STS).

Reactions may include coefficients that lead to instabilities for an explicit simulation. Very fast reactions would produce such coefficients. Therefore, the STS method processes reactions by an implicit calculation involving only the contaminants in the c.v. after all other calculations have been performed for the time step.

5.2.4 One-Dimensional Convection-Diffusion Flow

Although it is possible to use a time step as short as one second in CONTAM, this may not produce a more accurate simulation of transient concentrations because of the use of control volumes which are treated as well-mixed. For example, consider the sudden release of a contaminant at one end of a 10 m long zone with air flowing at a typical 0.2 m/s from the release

point toward a doorway at the other end. It would take about 50 seconds for the contaminant to reach that doorway and begin entering the next zone. In the well-mixed zone model the sudden release will instantly produce a uniform concentration (equal to the mass released divided by the mass of air) everywhere in the zone, even at the far end 10 m from the source, and begins entering the adjacent zone within one time step.

The well-mixed zone model is appropriate when the time step is longer than the mixing time of the zone. Conventional HVAC systems attempt to produce well mixed zones, but the mixing time is on the order of a few minutes rather than seconds.

The standard numerical solution to this problem is to create smaller control volumes whose size is similar to the distance traveled in one time step. This is done in the computational fluid dynamics (CFD) models that consider mass, momentum, and energy in computing the flow field in a space that has been divided into *many* control volumes. Conventional CFD might divide a volume into 30 cells along each of the three directions. This increases the computation effort from one large “cell” per zone to 27,000 ($=30 \times 30 \times 30$) cells per zone. This analysis approach is not practical for a building containing many zones.

One-dimensional convection-diffusion flow has been introduced to CONTAM as a compromise between simple, and fast, well-mixed zones and a full CFD simulation. CONTAM 2.4 includes the option of modeling detailed contaminant migration in one, user-defined direction through a zone and or an entire duct system. This one-dimensional model is obviously appropriate for flow through a duct and reasonably appropriate for a long hallway or a zone using a displacement ventilation system. Its use in more conventional well-mixed zones is problematic because of the presence of supply air jets and areas of recirculation.

Contaminant flow in one direction consists of a mixture of *convection*, the bulk movement of air, and *diffusion*, the mixing of the contaminant within the air. CONTAM’s primary 1-D convection diffusion model is taken directly from the finite volume method developed by Patankar [Patankar 1980] and described in more detail by Versteeg and Malalasekera [Versteeg and Malalasekera 1995]. This model divides the zone into a number of equal-length cells and uses an implicit method (with a fast tri-diagonal equation solver) to guarantee stability in computing the contaminant concentrations. It has been observed that the accuracy of this method declines as the ratio of the flow velocity \times time step to the length of the cell increases.

This loss of accuracy is a particular problem in the ducts where flow velocities can be quite high. CONTAM uses a Lagrangian model to handle high speed flows in ducts. In the Lagrangian model, air flowing at velocity u will create a cell $u\Delta t$ long at the inlet end of the duct segment and cause the cell at x_j to move to $x_j + u\Delta t$ during a time step of Δt . The length of the cell, Δx_j , is unchanged. This process of adding cells at the inlet end of the duct and deleting cells at the outlet end handles convection exactly. During that time step the contaminant will diffuse between adjacent cells due to molecular diffusion and turbulent mixing. That diffusion is solved by a standard implicit method using a tri-diagonal equation solver.

The cell at the outlet end of the duct segment will not necessarily have an edge at $x = L$ in which case an interpolation is necessary to compute the concentration at $x = L$, which becomes the input concentration to the next duct segment downstream. When two or more duct segments merge at a junction the contaminant concentration at the junction is the flow-weighted average of the concentrations at the end of each incoming duct.

High velocities produce long cells and, therefore, relatively few equations to be solved in a duct segment. At lower velocities more cells are required, and as the velocity approaches zero the number of cells approaches infinity. To prevent this, ContamX automatically switches to the Eulerian finite volume model when $u\Delta t$ is less than the user specified minimum cell length. This corresponds to the flow regime where the finite volume is most accurate.

The axial dispersion coefficient in ducts is computed from the following relations. For laminar flow ($Re < 2000$) the Taylor-Aris relation is used [Wen & Fan, p. 127]:

$$E = D_m + \frac{\bar{u}^2 d^2}{192 D_m}; \quad \frac{L}{d} < 0.04 \frac{\bar{u} d}{D_m} \quad (12)$$

where

- E = axial dispersion coefficient [m^2/s],
- D_m = molecular diffusion coefficient [m^2/s],
- \bar{u} = average fluid velocity [m/s],
- d = duct diameter [m], and
- L = length of duct [m].

The condition refers to a minimum length of duct for full development of laminar flow. Until another relation is found for undeveloped flow, equation (12) will be used for all laminar flows.

For turbulent flow E depends only on the Reynolds number, Re [Wen and Fan 1975, p. 149]:

$$\frac{E}{\bar{u}d} = \frac{3.0 \times 10^7}{Re^{2.1}} + \frac{1.35}{Re^{0.125}} \quad (13)$$

□ Duct Thermal Model

When performing simulations using the short time step method, a simple duct thermal model has been incorporated into the solution that may be implemented as an option to the user. This model addresses only convective heat transfer.

Conservation of thermal energy is analogous to the conservation of contaminant mass in a control volume:

$$(\text{energy at time } t + \Delta t) = (\text{energy at time } t) + \Delta t \times (\text{rate of energy gain} - \text{rate of energy loss})$$

Replace the concentration terms, C_i^α , in equation (8) with specific energy, $Cp_i T_i$:

$$\left(\rho_i V_i + \Delta t \cdot \sum_j F_{i \rightarrow j} \right) Cp_i T_i|_{t+\Delta t} \approx \rho_i V_i Cp_i T_i|_t + \Delta t \cdot \sum_j F_{j \rightarrow i} Cp_j T_j|_{t+\Delta t} \quad (14)$$

where

- Cp_i = specific heat of c.v. i , and
- T_i = temperature of c.v. i .

This is done only for ducts because the energy in the moving air dominates the solution in a well designed and well built duct system. That is not the case for zones where conductive and airflow heat transfers are of a similar scale. This model ignores heat exchange with the ductwork and its transient impact. However, the model should still be a useful help in computing the draft of a chimney.

5.2.5 Contaminant Source/Sink Elements

CONTAM allows you to define the generation, G_α , and removal, R_α , coefficients for some simple cases. Equations and parameters are provided here and explained in more detail in subsection Source/Sink Element Types of the Working with Sources and Sinks section.

□ Constant Coefficient Model

The constant coefficient or general source/sink model uses the following equation:

$$S_\alpha = G_\alpha - R_\alpha \cdot C_\alpha \quad (15)$$

where S_α is called the contaminant α "source strength". The CONTAM internal units for the terms in equation (16) are: C_α [$\text{kg}_\alpha / \text{kg}_{\text{air}}$], G_α , S_α [$\text{kg}_\alpha / \text{s}$], and R_α [$\text{kg}_{\text{air}} / \text{s}$]. You may express these values in a large number of units with automatic conversion to the internal values.

For a room air filtering device, $G = 0.0$ and $R = f \cdot e$ where f is the flow rate of the room air passing through the filter and e is the single pass removal efficiency of the device.

□ Pressure Driven Model

The pressure driven source/sink model is intended to model contaminant sources which are governed by the inside-outside pressure difference, such as radon or soil gas entry into a basement. In this case the source equation is:

$$S_\alpha = G_\alpha \cdot (P_{\text{ambt}} - P_i)^n \quad (16)$$

□ Cutoff Concentration Model

For volatile organic compounds the source equation is sometimes expressed in the form

$$S_\alpha = G_\alpha \left(1 - \frac{C_\alpha}{C_{\text{cutoff}}} \right) \quad (17)$$

where C_{cutoff} is the cutoff concentration at which emission ceases.

□ Decaying Source Model

Another source equation for volatile organic compounds is the exponentially decaying source which is expressed in the form

$$S_\alpha = G_\alpha e^{-t/t_c} \quad (18)$$

where

- S_α = the contaminant source strength,
- G_α = the initial emission rate,
- t = the time since the start of emission, and
- t_c = the decay time constant.

□ Boundary Layer Diffusion Controlled Model

The boundary layer diffusion controlled reversible source/sink model follows the descriptions presented by Axley 1991. The rate at which a contaminant is transferred into the sink is

$$h \cdot \rho \cdot A \left(C_i - \frac{C_s}{k} \right) \quad (19)$$

where

- h = average film mass transfer coefficient over the sink,
- ρ = film density of air, average of bulk and surface densities,
- A = surface area of the adsorbent,
- C_i = concentration in the air,
- C_s = concentration in the adsorbent, and
- k = Henry adsorption constant or partition coefficient.

□ **Burst Source Model**

A user-specified mass of contaminant is added to a zone in a single time step, effectively an instantaneous addition – nothing can be resolved at less than one time step.

□ **Deposition Velocity Sink Model**

The deposition velocity model provides for the input of a sink's characteristic in the familiar term of deposition velocity. The deposition velocity model equation is:

$$R_{\alpha}(t) = v_d A_s m \rho_{air}(t) C_{\alpha}(t) s(t)$$

where

- $R_{\alpha}(t)$ = removal rate at time t
- v_d = deposition velocity
- A_s = deposition surface area
- m = element multiplier
- $\rho_{air}(t)$ = density of air in the source zone at time t
- $C_{\alpha}(t)$ = concentration of contaminant α at time t [M_{α} / M_{air}]
- $s(t)$ = schedule or control signal value at time t [-]

□ **Deposition Rate Sink Model**

The deposition rate model provides for the input of a sink's characteristic in the familiar term of deposition or removal rate. The deposition rate model equation is:

$$R_{\alpha}(t) = k_d V_z \rho_{air}(t) C_{\alpha}(t) s(t) m$$

where

- k_d = Deposition rate [1/T]
 - V_z = Zone volume [M^3]
- other terms are the same as for the Deposition Velocity Sink Model.

5.3 Airflow Analysis

Over the years many methods have been developed to compute the building airflows which are necessary for the contaminant analysis. Feustel and Dieris report 50 different computer programs for multizone airflow analysis [Feustel and Dieris 1992]. Note that "zones" go by many other names in these programs, e.g., nodes, cells, and rooms are common alternatives. The airflow calculations in CONTAM are based on the algorithms developed in AIRNET [Walton 1989a and 1989b].

5.3.1 Basic Equations

The air flow rate from zone j to zone i , $F_{j,i}$ [kg/s], is some function of the pressure drop along the flow path, $P_j - P_i$:

$$F_{j,i} = f(P_j - P_i) \quad (1)$$

The mass of air, m_i [kg], in zone i is given by the ideal gas law

$$m_i = \rho_i V_i = \frac{P_i V_i}{RT_i} \quad (2)$$

where

- V_i = zone volume [m^3],
- P_i = zone pressure [Pa],
- T_i = zone temperature [K], and
- R = 287.055 [J/kg·K] (gas constant for air).

For a transient solution the principle of conservation of mass states that

$$\frac{\partial m_i}{\partial t} = \rho_i \frac{\partial V_i}{\partial t} + V_i \frac{\partial \rho_i}{\partial t} = \sum_j F_{j,i} + F_i \quad (3)$$

$$\frac{\partial m_i}{\partial t} \approx \frac{1}{\Delta t} \left[\left(\frac{P_i V_i}{RT_i} \right)_t - (m_i)_{t-\Delta t} \right] \quad (4)$$

where

- m_i = mass of air in zone i ,
- $F_{j,i}$ = airflow rate [kg/s] between zones j and zone i : positive values indicate flows from j to i and negative values indicate flows from i to j , and
- F_i = non-flow processes that could add or remove significant quantities of air from the zone.

CONTAM 1.0 did not provide for such non-flow processes and flows were evaluated by assuming quasi-steady conditions leading to the following equation

$$\sum_j F_{j,i} = 0 \quad (5)$$

CONTAM can now provide for such non-flow processes by allowing the density to vary during time steps when performing transient simulations.

You can activate this option with the *Vary Density During Time Step* setting under the Airflow Numerics Simulation Parameters. If this parameter is set then, equation 3 is implemented when performing airflow calculations; otherwise equation 5 is used.

5.3.2 Solving the Equations

The steady-state airflow analysis for multiple zones requires the simultaneous solution of equation (5) for all zones. Since the function in equation (1) may be, and usually is, nonlinear, a method is needed for the solution of simultaneous nonlinear algebraic equations. The Newton-Raphson (N-R) method [Conte and de Boor 1972 p. 86] solves the nonlinear problem by an iteration of the solutions of linear equations. In the N-R method a new estimate of the vector of all zone pressures, $\{\mathbf{P}\}^*$, is computed from the current estimate of pressures, $\{\mathbf{P}\}$, by

$$\{\mathbf{P}\}^* = \{\mathbf{P}\} - \{\mathbf{C}\} \quad (6)$$

where the correction vector, $\{\mathbf{C}\}$, is computed by the matrix relationship

$$[\mathbf{J}]\{\mathbf{C}\} = \{\mathbf{B}\} \quad (7)$$

where $\{\mathbf{B}\}$ is a column vector with each element given by

$$\mathbf{B}_i = \sum_j F_{j,i} \quad (8)$$

and $[\mathbf{J}]$ is the square (i.e. N by N for a network of N zones) Jacobian matrix whose elements are given by

$$\mathbf{J}_{i,j} = \sum_i \frac{\partial F_{j,i}}{\partial P_j} \quad (9)$$

In equations (8) and (9) $F_{j,i}$ and $\partial F_{j,i}/\partial P_j$ are evaluated using the current estimate of pressure $\{\mathbf{P}\}$. The ContamX program contains subroutines for each airflow element which return the mass flow rates and the partial derivative values for a given pressure difference input.

Equation (7) represents a set of linear equations which must be set up and solved for each iteration until a convergent solution of the set of zone pressures is achieved. In its full form $[\mathbf{J}]$ requires computer memory for N^2 values, and a standard Gauss elimination solution has execution time proportional to N^3 . Sparse matrix methods can be used to reduce both the storage and execution time requirements. A skyline solution process following the method presented in [Dhatt 1984] was chosen. This method can be used to solve equations with symmetric or asymmetric matrices. It stores no zero values above the highest nonzero element in the columns above the diagonal and no zero values to the left of the first nonzero value in each row below the diagonal. In this case the Jacobian matrix is symmetric. CONTAM provides two solution methods for the linear equations: Skyline (also called profile method) and Pre-conditioned Conjugate Gradient (PCG). PCG may be useful for problems with many zones and junctions.

Analysis of the element models will show that

$$|\mathbf{J}_{i,i}| = \sum_{j+i} |\mathbf{J}_{i,j}| \quad (10)$$

This condition allows a solution without pivoting, although scaling may be useful. Note that the degree of sparsity of the Jacobian matrix after factoring is dependent on the ordering of the zones. Ordering can be improved by various algorithms or rules-of-thumb. In AIRNET it was easy to define an airflow network which had no unique solution. The ContamW user interface insures the correct interconnection of the airflow elements in the network.

CONTAM allows zones with either known or unknown pressures. The constant pressure zones are included in the system of equations and equation (7) is processed so as to not change those zone pressures. This gives flexibility in defining the airflow network while maintaining the symmetric set of equations. A sufficient condition for the Jacobian to be nonsingular [Axley 1987] is that all of the unknown pressure zones be linked by pressure dependent flow paths to (a) constant pressure zone(s). In CONTAM the ambient (or outdoor) air is treated as a constant pressure zone. The ambient zone pressure is assumed to be zero for the flow calculation causing the computed zone pressures to be values relative to the true ambient pressure and helping to maintain numerical significance in calculating ΔP .

Conservation of mass at each zone provides the convergence criterion for the N-R iterations. That is, when equation (4) is satisfied for all zones for the current system pressure estimate, the solution has converged. Sufficient accuracy is attained by testing for relative convergence at each zone:

$$\frac{\left| \sum_j F_{j,i} \right|}{\sum_j |F_{j,i}|} < \varepsilon \quad (11)$$

with a test ($\sum |F_{j,i}| < \varepsilon 1$, the absolute convergence factor) to prevent division by zero. The magnitude of ε can be established by considering the use of the calculated airflows, such as in an energy balance. In any case, round-off errors may prevent perfect convergence ($\varepsilon = 0$).

Numerical tests of the N-R method solution indicated occasional instances of very slow convergence as the iterations almost oscillate between two different sets of values. In AIRNET, this was handled by a Steffensen acceleration process. More recent tests by the author and by Wray [Wray 1993] indicate that the use of a simpler constant under-relaxation coefficient produces a faster, reliable convergence acceleration process. Equation (6) for the iteration process becomes

$$\{\mathbf{P}\}^* = \{\mathbf{P}\} - \omega \{\mathbf{C}\} \quad (12)$$

where ω is the relaxation coefficient. A relaxation coefficient of 0.75 has been found to be usable for a broad range of airflow networks. This value is not a true optimum but appears to work quite well without the computational cost of finding the theoretically optimum value.

When convergence is progressing rapidly, under-relaxation ($\omega < 1$) slows convergence compared to no relaxation. To prevent this a global convergence value is computed:

$$\gamma = \frac{\sum_i \left| \sum_j F_{j,i} \right|}{\sum_i \sum_j |F_{j,i}|} \quad (13)$$

When $\gamma^* < \alpha \gamma$, ω is set to 1. Currently CONTAM uses $\alpha = 30\%$. This often reduces the number of iterations. This is simple under-relaxation. CONTAM also may alternatively use a simple trust region method implemented by David M. Lorenzetti based on [Dennis and Schnabel 1996].

Newton's method requires an initial set of values for the zone pressures. These may be obtained by including in each airflow element model a linear approximation relating the flow to the pressure drop:

$$F_{j,i} = c_{j,i} + b_{j,i}(P_j - P_i) \quad (14)$$

Conservation of mass at each zone leads to a set of linear equations of the form

$$[\mathbf{A}]\{\mathbf{P}\} = \{\mathbf{B}\} \quad (15)$$

Matrix $[\mathbf{A}]$ in equation (15) has the same sparsity pattern as $[\mathbf{J}]$ in equation (7) allowing use of the same sparse matrix solution process for both equations. This initialization handles stack effects very well and tends to establish the proper directions for the flows. The linear approximation is conveniently provided by the laminar regime of the element models used by CONTAM. When solving a set of similar problems, as when approximating a transient solution by successive steady-state solutions, it tends to be preferable to use the previous solution for the zone pressures as the initial values for the new problem.

5.3.3 Airflow Elements

Infiltration is the result of air flowing through openings, large and small, intentional and accidental, in the building envelope. Simulation programs require a mathematical model of the flow characteristics of the openings. For a general introduction see Chapter 27 of [ASHRAE 2005] and section 2.2 of [Feustel 1990].

Flow within each airflow element is assumed to be governed by Bernoulli's equation:

$$\Delta P = \left(P_1 + \frac{\rho V_1^2}{2} \right) - \left(P_2 + \frac{\rho V_2^2}{2} \right) + \rho g(z_1 - z_2) \quad (16)$$

where

ΔP	= total pressure drop between points 1 and 2
P_1, P_2	= entry and exit static pressures
V_1, V_2	= entry and exit velocities
ρ	= air density
g	= acceleration of gravity (9.81 m/s ²)
z_1, z_2	= entry and exit elevations.

The following parameters apply to the zones: pressure, temperature (to compute density and viscosity), and elevation. The zone elevation values are used to determine stack effect pressures. When the zone represents a room, the airflow elements may connect with the room at other than its reference elevation. The hydrostatic equation is used to relate the pressure difference across a flow element to the elevations of the element ends and the zone elevations, assuming the air in the room is at constant temperature. Pressure terms can be rearranged and a possible wind pressure for building envelope openings added to give

$$\Delta P = P_j - P_i + P_S + P_W \quad (17)$$

where

$$\begin{aligned} P_i, P_j &= \text{total pressures at zones } i \text{ and } j \\ P_S &= \text{pressure difference due to density and elevation differences, and} \\ P_W &= \text{pressure difference due to wind.} \end{aligned}$$

Equation (17) establishes a sign convention for direction of flow: positive is from zone j to zone i . Since the airflow elements will be described by a relationship of the form $w = f(\Delta P)$, the partial derivatives needed for $[J]$ in equation (9) are related by $\partial w / \partial P_j = -\partial w / \partial P_i$ which establishes the relation in equation (10). Many forms of airflow elements are available in CONTAM.

5.3.3.1 Powerlaw Flow Elements

Most infiltration models are based on the following empirical (powerlaw) relationship between the flow and the pressure difference across a crack or opening in the building envelope:

$$Q = C(\Delta P)^n \quad (18)$$

The volumetric flow rate, Q [m^3/s], is a simple function of the pressure drop, ΔP [Pa], across the opening. A common variation of the powerlaw equation is:

$$F = C(\Delta P)^n \quad (19)$$

where the mass flow rate, F [kg/s], is a simple function of the pressure drop. A third variation is related to the orifice equation:

$$Q = C_d A \sqrt{\frac{2\Delta P}{\rho}} \quad (20)$$

where

$$\begin{aligned} C_d &= \text{discharge coefficient, and} \\ A &= \text{orifice opening area.} \end{aligned}$$

Theoretically, the value of the flow exponent should lie between 0.5 and 1.0. Large openings are characterized by values very close to 0.5, while values near 0.65 have been found for small crack-like openings.

The primary advantage of equations (18-20) for describing airflow components is the simple calculation of the partial derivatives for the Newton's method solution of the simultaneous equations:

$$\frac{\partial F_{j,i}}{\partial P_j} = \frac{nF_{j,i}}{\Delta P} \quad \text{and} \quad \frac{\partial F_{j,i}}{\partial P_i} = \frac{-nF_{j,i}}{\Delta P} \quad (21)$$

The sign in equations (21) will agree with the sign of F . However, there is also a problem with equations (21): the derivatives become unbounded as the pressure drop (and the flow) go to zero. A simple way to avoid this problem is suggested by what physically happens at low flow rates: the physical character of the flow (and the form of the equation) changes. It goes from turbulent to laminar. Equations (18-20) can be replaced by

$$F = \frac{C_k \rho \Delta P}{\mu} \quad (22)$$

where

C_k = laminar flow coefficient, and
 μ = viscosity.

The partial derivatives are simple constants:

$$\frac{\partial F_{j,i}}{\partial P_j} = \frac{C_k P}{\mu} \quad \text{and} \quad \frac{\partial F_{j,i}}{\partial P_i} = \frac{-C_k \rho}{\mu} \quad (23)$$

The origin of this laminar relationship is shown by the duct equations in the next section. This technique has been independently discovered and used by several researchers [Axley 1987] and [Isaacs 1980]. Although there is physical reason for using equation (22) at low pressure drops, its purpose here is to assure convergence of the equations when ΔP approaches zero for one of the many flow paths in a complex network, instead of accurately representing airflows which are too small to be of interest. Because the linear flow expression is not used as a true flow model but as a mathematical artifice, it is not necessary to adjust its flow coefficient. Given the uncertainty in estimating the temperature of the air as it flows through an opening, especially a crack, this additional detail is of debatable usefulness.

The CONTAM functions for powerlaw elements calculate flows using both the laminar and the turbulent models and select the method giving the smaller magnitude flow. There is a discontinuity in the derivative of the $F(\Delta P)$ curve where the two equations intersect. This discontinuity is a violation of one of the sufficient conditions for convergence of Newton's method [Conte and de Boor 1972, p. 86]. However, numerical tests conducted by the author for flows at that point using a small airflow network have shown no convergence problem.

Temperature Dependence

It is useful to think of the coefficient C as a simple constant, $C\alpha$, evaluated at a particular set of conditions (μ_0, ρ_0 and $v_0 = \mu_0/\rho_0$) multiplied by a correction factor to account for actual air properties. Equations (18-20) are converted to a common form and summarized below with their appropriate temperature correction factors.

$\Delta P > 0$	$\Delta P < 0$	Correction Factor
$F_{j,i} = K_a C_a \rho_j (\Delta P)^n$	$F_{j,i} = -K_a C_a \rho_i (-\Delta P)^n$	$K_a = (\rho_o / \rho)^n (v_o / v)^{2n-1}$
$F_{j,i} = K_b C_b \sqrt{\rho_j} (\Delta P)^n$	$F_{j,i} = -K_b C_b \sqrt{\rho_i} (-\Delta P)^n$	$K_b = (\rho_o / \rho)^{n-1/2} (v_o / v)^{2n-1}$
$F_{j,i} = K_c C_c (\Delta P)^n$	$F_{j,i} = -K_c C_c (-\Delta P)^n$	$K_c = (\rho_o / \rho)^{n-1} (v_o / v)^{2n-1}$

CONTAM uses the following formulae for computing ρ and v :

$$\rho = P / (287.055 T)$$

$$\mu = 3.7143 \times 10^{-6} + 4.9286 \times 10^{-8} T$$

$$v = \mu / \rho$$

Using reference conditions of standard atmospheric pressure and 20°C gives $\rho_o = 1.2041 \text{ kg/m}^3$ and $v_o = 1.5083 \times 10^{-5} \text{ m}^2/\text{s}$.

Fitting Powerlaw Coefficients

Experimental data can be used to determine the coefficients in the orifice form of the powerlaw equation:

$$F = \pm C_b \sqrt{\rho_o} (\pm \Delta P)^n \quad (K_b = 1) \quad (25)$$

If n is known or can be assumed, C_b , in equation (24), can be computed from the inverse of equation (25)

$$C_b = \frac{F}{\sqrt{\rho_o} (\Delta P)^n} \quad (26)$$

When two points $(F_1, \Delta P_1)$ and $(F_2, \Delta P_2)$, are known, n can be computed from:

$$n = \frac{\ln(F_1) - \ln(F_2)}{\ln(\Delta P_1) - \ln(\Delta P_2)} \quad (27)$$

with C_b then computed from equation (26).

Leakage Areas

The powerlaw model can be used with the component leakage area formulation which has been used to characterize openings for infiltration calculations [ASHRAE 2001, p. 25.18]. The leakage area is based on a series of pressurization tests where the airflow rate is measured at a series of pressure differences ranging from about 10 Pa to 75 Pa. The effective leakage area is based on a rearrangement of equation (20)

$$L = \frac{Q_r \sqrt{\rho / 2 \Delta P_r}}{C_d} \quad (28)$$

where

- L = equivalent or effective leakage area [m^2],
- ΔP_r = reference pressure difference [Pa],
- Q_r = predicted airflow rate at ΔP_r (from curve fit to pressurization test data) [m^3/s], and
- C_d = discharge coefficient.

There are two common sets of reference conditions:

$$C_d = 1.0 \text{ and } \Delta P_r = 4 \text{ Pa}$$

or

$$C_d = 0.6 \text{ and } \Delta P_r = 10 \text{ Pa.}$$

A leakage area can be converted to the flow coefficient by

$$C_b = LC_d \sqrt{2} (\Delta P_r)^{1/2-n} \quad (29)$$

This equation requires a value for n . If it is not reported with the test results, a value between 0.6 and 0.7 is reasonable.

Stairwells

A stairwell will normally be modeled as a vertical series of zones connected by low resistance openings through the floors. The CONTAM model for airflow in stairwells is based on a fit to experimental data [Achakji and Tamura 1988]. They expressed the airflow resistance per floor as an effective area A_e in the orifice equation (20) with a 0.6 discharge coefficient. The effective area is expressed in terms of the area of the shaft A_s , the distance between floors h , the density of people on the stairs d , and whether the treads are open or closed. A large number of people on the stairs, as in an evacuation scenario, influences the flow resistance. The experiment used densities of 0, 1, and 2 persons/m². For open treads the effective area is approximately

$$A_e \approx A_s (0.089h) (1.0 - 0.14\sqrt{d}) \quad (30)$$

and for closed treads

$$A_e \approx A_s (0.083h) (1.0 - 0.24\sqrt{d}) \quad (31)$$

The coefficients for the powerlaw equation are

$$n = 0.5$$

$$n = 0.5 \quad \text{and} \quad C_b = 0.6 \cdot \sqrt{2} \cdot A_e \quad (32)$$

Cracks

A relationship for flow through cracks that can be converted directly into a powerlaw airflow element is presented in [Clarke 1985, p. 204] as:

$$Q = ka(\Delta P)^n \quad (33)$$

where

$$n = 0.5 + 0.5 \exp(-W / 2), \quad (34)$$

and

$$k = 0.0097 \cdot (0.0092)^n \quad (35)$$

with

W = crack width (mm), and

a = crack length (m).

Therefore, the coefficients in the powerlaw equation (19) are given by n in equation (34) and

$$C_b = \sqrt{\rho_0} \cdot a \cdot 0.0097 (0.0092)^n \quad (36)$$

5.3.3.2 Quadratic Flow Elements

Baker, Sharples, and Ward [Baker 1987] indicate that infiltration openings can be more accurately modeled by a quadratic relationship of the form

$$\begin{aligned} Q, \Delta P > 0 & & Q, \Delta P < 0 \\ \Delta P = AQ + BQ^2 & & \Delta P = AQ - BQ^2 \end{aligned} \quad (37)$$

This form can be used as an airflow element by solving the quadratic equation for $F (= \rho Q)$. Letting $a = A/\rho$ and $b = B/\rho^2$ allows equations (37) to be rewritten as

$$\Delta P = aF + bF^2 \quad \text{and} \quad \Delta P = aF - bF^2 \quad (38)$$

These quadratic equations solve as

$$F = \frac{\sqrt{a^2 + 4b\Delta P} - a}{2b} \quad \text{and} \quad F = \frac{a - \sqrt{a^2 - 4b\Delta P}}{2b} \quad (39)$$

with the partial derivatives given by

$$\frac{\partial F_{j,i}}{\partial P_i} = \frac{1}{a + 2b|F_{j,i}|} \quad \text{and} \quad \frac{\partial F_{j,i}}{\partial P_i} = \frac{-1}{a + 2b|F_{j,i}|} \quad (40)$$

Equations (39) require that b be nonzero to prevent a division by zero and equations (40) requires that a be nonzero to prevent a division by zero as F goes to zero. There are contrary opinions that the powerlaw relationship is better.

Temperature Dependence

It is useful to think of the coefficients a and b as simple constants evaluated at a particular set of conditions (μ_0 , ρ_0 and $v_0 = \mu_0/\rho_0$) multiplied by correction factors to account for actual air properties as was done for the powerlaw equations. That is

$$A = \left(\frac{v}{v_0}\right)A_0 \quad \text{and} \quad a = \left(\frac{v}{v_0}\right)a_0 \quad (41)$$

and

$$B = \left(\frac{\rho_0}{\rho}\right)B \quad \text{and} \quad b = \left(\frac{\rho_0}{\rho}\right)b_0 \quad (42)$$

Note that $\rho_0/\rho = T/T_0$ for a perfect gas of constant composition.

Fitting Quadratic Coefficients

The quadratic coefficients can also be computed from measured flow and pressure data. Given two points $(F_1, \Delta P_1)$ and $(F_2, \Delta P_2)$, the values of a_0 and b_0 are:

$$b_0 = \frac{\left(\frac{\Delta P_1}{F_1} - \frac{\Delta P_2}{F_2} \right)}{F_1 - F_2} \quad (43)$$

and

$$a_0 = \frac{\Delta P_1}{F_1} - b_0 F_1 \quad (44)$$

The main advantage of the quadratic model over the powerlaw model is computation speed achieved by avoiding the slow power function. (Tests have shown $\text{pow}(x)$ four to eight times slower than $\text{sqrt}(x)$. This performance is hardware and software dependent.) It must still be determined which model is the most accurate representation for a particular airflow element. For example, it has been found that the powerlaw model is a better approximation for smooth ducts while the quadratic model is a better approximation for rough ducts. However, for very large openings the derivatives (40) can become quite large leading to slow convergence of the simultaneous mass balance equations.

Crack Description

Theoretical relationships have been developed between the coefficients A and B of the quadratic airflow element model and the physical characteristics of the openings [Baker 1987]. These are

$$A_0 = 12\mu_0 z / Ld^3 \quad \text{and} \quad B_0 = \rho_0 C / 2d^2 L^2 \quad (45)$$

where

- μ = viscosity,
- ρ = density,
- z = distance along the direction of flow,
- d = crack width,
- L = crack length, and
- C = 1.5 + number of bends in the flow path.

For the mass flow form of the equations (38) the coefficients are

$$a_0 = 12\mu_0 z / \rho_0 Ld^3 \quad \text{and} \quad b_0 = C / 2\rho_0 d^2 L^2 \quad (46)$$

5.3.3.3 Ducts

The theory of flows in ducts (and pipes) is well established and summarized in Chapter 35 of the *2005 ASHRAE Fundamentals Handbook* [ASHRAE 2005] and treated more extensively in [Blevins 1984] in a chapter on pipe and duct flow. See the *Duct Fitting Database* [ASHRAE 2002] for extensive data on dynamic losses in duct fittings. Analysis is based on Bernoulli's equation and its assumptions. The friction losses in a section of duct or pipe are given by

$$\Delta P_f = f \frac{L}{D} \frac{\rho V^2}{2} \quad (47)$$

where

f = friction factor,
 L = duct length, and
 D = hydraulic diameter.

The dynamic losses due to fittings and so forth are given by

$$\Delta P_d = C_d \frac{\rho V^2}{2} \quad (48)$$

where C_d = dynamic loss coefficient. Total pressure losses are given by

$$\Delta P = \Delta P_f + \sum \Delta P_d \quad (49)$$

Since $F = \rho VA$, where A is the cross section (or flow) area,

$$F = \sqrt{\frac{2\rho A^2 \Delta P}{fL/D + \sum C_d}} \quad (50)$$

CONTAM calculates the friction factor using the nonlinear Colebrook equation [ASHRAE 2001, p 2.9, eqn. 29b]:

$$\frac{1}{\sqrt{f}} = 1.44 + 2 \cdot \log(D/\varepsilon) - 2 \cdot \log\left(1 + \frac{9.3}{\text{Re} \cdot \varepsilon/D \cdot \sqrt{f}}\right) \quad (51)$$

where

ε = roughness dimension, and
 Re = Reynolds number = $\rho VD/\mu = FD/\mu A$.

This nonlinear equation may be readily solved using the following iterative expression derived from equation (51) by Newton's method:

$$g^* = g - \frac{g - \alpha + \gamma \ln(1 + g\beta)}{1 + \gamma\beta/(1 + g\beta)} \quad (52)$$

where

$g = f^{-1/2}$,
 $\alpha = 1.14 - \gamma \ln(\varepsilon/D)$,
 $\beta = 9.3/(\text{Re} \cdot \varepsilon/D)$, and
 $\gamma = 2 \cdot \log(e) = 0.868589$.

The convergent solution is achieved in 2 or 3 iterations of equation (52) using $g = \alpha$ as a starting value. If the value of g has been saved from the previous time it was computed for a particular duct element, and the flow rate has not changed greatly, only one iteration of equation (52) will be needed to compute the friction factor.

The exact derivatives of equation (50) are difficult to compute, so CONTAM uses a secant approximation. The derivatives suffer the standard problem of powerlaw equations, i.e., they go undefined as ΔP approaches zero. This is solved in CONTAM by the linear approximation (22) with the coefficient computed to give the same flow as equation (50) at the user specified

transition Reynolds number (default = 2000). A more detailed description of the flow in the laminar region could be developed, but that would probably exceed the level of detail with which the rest of the problem is described in CONTAM.

□ **Terminal Loss Calculation**

In CONTAM, duct terminals are essentially zero-length ducts characterized by an equivalent loss coefficient that is determined during simulation from the properties of the terminal, C_t and C_b , and the schedule or control signal (S) acting upon the terminal.

$$C_e = C_t + C_b + (1-S)/S$$

where

C_t = terminal loss coefficient

C_b = terminal balance coefficient

S_c = schedule or control signal acting upon the terminal

$S = \max(1 \times 10^{-6}, S_{cl})$

$S_{cl} = \min(1, S_c)$

Using this technique, an input signal S_c of 0.0 will effect a very high loss coefficient, that of 1.0 will simply provide the loss determined by the combination of C_t and C_b , S_c between 1×10^{-6} and 1 will be used to increase the value of $(C_t + C_b)$ according to the equation above, and S_c less than 0.0 will equate to 1.0, i.e., the loss can not be reduced below $(C_t + C_b)$. The value C_e will then be used as the dynamic loss coefficient in the duct flow equations presented above.

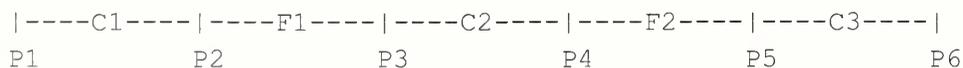
5.3.3.4 Fans

This section describes the theory of how forced flow elements are modeled with CONTAM. Forced flow elements include constant flow fans and variable flow fans. Constant flow fans include both constant volume and constant mass flow. Variable flow fans are modeled based on the input of a fan performance curve that relates pressure drop to airflow through the fan flow element.

Constant Flow Fans

One particularly simple but useful airflow element sets a constant flow between two nodes. Since the flow is constant, the partial derivatives of flow with respect to the node pressures must be zero. The constant flow element does not contribute to the Jacobian, $[A]$, but it does add to the right side vector, $\{B\}$.

Constant flow elements do not mathematically link the pressures of the adjacent nodes. It is necessary that all node in the network be linked to constant pressure nodes in order to have a unique solution. Violation of this restriction will produce a division by zero somewhere in the solution of the equations. Consider the following simple network:



where P_1 and P_6 are known pressures, and F_1 and F_2 are known flows. Since the flow through C_2 is determined by $P_3 - P_4$, and no other flow is related to those pressures, there are not enough

equations to determine P_3 and P_4 uniquely. This is a mathematical expression of the fact that it is possible for F_1 and F_2 to be assigned different flows, which produces a physically impossible condition.

CONTAM provides two constant flow elements: one for constant mass flow and one for constant volumetric flow.

Variable Flow Fans

The theory of flows induced by fans is summarized in Chapter 18 of the *2000 ASHRAE HVAC Systems and Equipment Handbook* [ASHRAE 2000]. More extensive treatment is given in [Osborne 1977]. Fan performance is normally characterized by a performance curve which relates the total pressure *rise* to the flow rate for a given fan speed and air density. Conversion to another fan speed or density is done with the fan laws.

$$Q / Q_0 = N / N_0 \quad (53)$$

or

$$F / F_0 = N\rho / N_0\rho_0 \quad (54)$$

and

$$P / P_0 = N^2\rho / N_0^2\rho_0 \quad (55)$$

where

- Q = volume flow rate,
- F = mass flow rate,
- P = total pressure rise,
- ρ = density,
- N = rotational speed, and
- subscript 0 indicates values at the rating conditions for the fan.

These laws are valid if all flow conditions at the two speeds are similar. In particular, they will not apply at very low flows where fully turbulent conditions have not been developed.

In CONTAM the fan performance curve is represented by a cubic polynomial:

$$P = a_0 + a_1F + a_2F^2 + a_3F^3 \quad (56)$$

with

$$P' = \frac{dP}{dF} = a_1 + 2a_2F + 3a_3F^2 \quad (57)$$

The polynomial coefficients are developed with air density and fan speed at the rating conditions. Therefore, when N and ρ are not at the rating conditions, convert the actual pressure rise P to P_0 .

$$P_0 = PN_0^2\rho_0 / N^2\rho \quad (58)$$

Solve (56) for F_0 using iterative or analytic solution and (57) for P'_0 . Convert these values to current conditions.

$$F = F_0 N \rho / N_0 \rho_0 \tag{59}$$

and

$$P' = P_0 N / N_0 \tag{60}$$

The derivative for flow with respect to pressure drop is given by

$$\frac{dF}{dP} = -\frac{1}{P'} \tag{61}$$

There are two important factors to note on the shape of the fan performance curve. First, it is described by a relationship of the form $P(F)$ instead of $F(P)$ which would be more appropriate for the calculation of flow and partial derivatives for the Jacobian. The basic shape of the performance curve cannot be well represented by a simple polynomial with P as the independent variable. CONTAM uses an analytic solution of the cubic polynomial (56) to determine F as a function of P .

Second, it is common for the performance curve to contain points of contraflexure (where $P' = 0$) creating up to three different flow rates at certain values of fan pressure rise. This causes difficulty in solving for the flow rate and has points where dF/dP goes to infinity. However, it is usually not recommended that the fan operate in the region of the contraflexure points. Therefore, the fan can be modeled with a performance curve that does include the contraflexure so long as you make sure that the air distribution system has not operated in that region.

You can identify the points of contraflexure from the coefficients of the polynomial. Solving equation (56) for F gives:

$$F = \frac{-2a_2 \pm \sqrt{4a_2^2 - 12a_1a_3}}{6a_3} \tag{62}$$

$4a_2^2 > 12a_1a_3$ There are two real points of contraflexure

$4a_2^2 = 12a_1a_3$ There is a point of inflection which still has the derivative problem

$4a_2^2 < 12a_1a_3$ The fan performance curve is monotonic and well-behaved for computing the derivatives for the Jacobian.

5.3.3.5 Doorways (Large Openings)

The powerlaw and quadratic models allow flow in only one direction at a time. Flows through large openings (e.g. doorways) tend to be more complex with the possibility of flows in opposite directions in different parts of the opening. The temperature and resulting density differences between two rooms may mean that the stack effect causes a positive pressure difference at the top of the doorway and a negative pressure difference at the bottom (or vice versa) allowing a two-way flow in the opening. A summary of research on heat transfer through doorways is presented in [Barakat 1987]. Most research has attempted to develop dimensionless correlations (using Nusselt, Prandlt, and Grashoff numbers) of the form

$$N_d / \text{Pr} = C \cdot Gr_D^b \quad (63)$$

where b is approximately 0.5 and C lies between 0.22 and 0.33 depending on the temperature difference used for the correlation. It has been shown that such a heat transfer is equivalent to an airflow which can be modeled by powerlaw elements by dividing the total opening into several smaller openings having the same total area but configured to properly account for the magnitude and direction of airflows at different heights in the opening.

An alternative approach is to create a single airflow element which accounts for the flow over the entire opening. A simple theory which estimates the stack induced air flow through a large opening in a vertical partition is given in [Brown and Solvason 1962]. This model of a doorway tends to be faster than the multiple opening approach. However, it also complicates the assembly process for the Jacobian matrix because one or two flows may exist. More importantly, development of the doorway element model requires knowledge of the vertical temperature profile used in the node model (here assumed to be constant) in order to compute the pressure difference as a function of height across the opening. This requirement compromises the independence of the modularity of airflow network program.

Multiple Opening Model

By assuming that the air density in each room is constant, the hydrostatic equation is used to relate pressures at various heights in each room:

$$P_{0j} = P_j + \rho_j g (h_j - h_0) \quad \text{and} \quad P_{0i} = P_i + \rho_i g (h_i - h_0) \quad (64)$$

$$P_j(y) = P_{0j} - \rho_j g y \quad \text{and} \quad P_i(y) = P_{0i} - \rho_i g y \quad (65)$$

where

- P_{0j}, P_{0i} pressure in zones j and i at $y = 0$, the reference elevation of the opening,
- ρ_j, ρ_i air densities of zones j and i ,
- P_j, P_i reference pressures of zones j and i ,
- h_j, h_i reference elevations of zones j and i , and
- h_0 elevation of the center of the opening.

Following [Brown and Solvason 1962] it is assumed that the velocity of the airflow as a function of height is given by the orifice equation:

$$V(y) = C_d \left(\frac{2(P_j(y) - P_i(y))}{\rho} \right)^{1/2} \quad (66)$$

where

- C_d = discharge coefficient, and
- ρ = density of the air going through the opening.

The opening, H high by W wide, may be divided into multiple openings each representing a thin horizontal strip Δy high and W wide. The mass flow through each strip would be

$$F_y = C_d \sqrt{2\rho\Delta P} W \Delta y \quad (67)$$

The coefficient for the powerlaw model (20) for each sub-opening is thus given by

$$C_0 = \sqrt{2} C_d W \Delta y \quad (68)$$

Two Opening Model

An approximate model can be constructed by considering an opening between two zones which have no flow paths to any other zones. Then the flow in each direction must be equal and $P_{0j} = P_{0i}$. Integration of equation (66) from $y=0$ to $y=H/2$ using (65) to compute ΔP gives the flow through the top half of the opening.

$$w = \int_{y=0}^{y=H/2} \rho V W dy = \frac{C_d}{3} W \sqrt{\rho g \Delta \rho} H^3 \quad (69)$$

This is equivalent to a simple powerlaw model with

$$C_0 = C_d \sqrt{2} \left(\frac{W H}{2} \right) \quad (70)$$

(a simple orifice with one half the total opening area) placed at an elevation of $y = 2H/9$. The two opening model is completed with an identical opening at $y = -2H/9$ elevation allowing an equal flow in the opposite direction. Since the orifice is being modeled with simple orifice openings, any additional pressure drop between the zones due to other forces is included in the two opening model. The model becomes less accurate as the neutral plane shifts from the center of the opening.

Single Opening Model

A third approach is to create a single airflow element which accounts for the flow over the entire opening. Begin by defining the neutral height, Y , where the velocity of the air is zero. From equation (66) this must occur when $P_j(y) = P_i(y)$. From equations (65) this must be

$$Y = \frac{P_{0j} - P_{oi}}{g(\rho_j - \rho_i)} \left(\frac{P_{0i} - P_{oj}}{g(\rho_i - \rho_j)} \right) \quad (71)$$

If $|Y| < H/2$, there is two-way airflow through the opening. If $\rho_j = \rho_i$, the neutral height cannot be computed, but, since there is no possibility of two-way flow, the opening can be considered a simple orifice opening.

Define $\Delta \rho \equiv \rho_j - \rho_i$ and a transformed height coordinate $z \equiv y - Y$. Then the pressure difference across the opening is given by

$$P_j(z) - P_i(z) = -gz\Delta \rho \quad (72)$$

The mass flow through the opening above the neutral height is given by

$$F_a = W \int_{z=0}^{z=\frac{H}{2}-Y} \rho_m V dz \quad m = j \text{ or } i \quad (73)$$

and the mass flow below the neutral height by

$$F_b = W \int_{x=\frac{H}{2}-Y}^{x=-} \rho_m V dz \quad m = j \text{ or } i \quad (74)$$

Whether the subscript m should be j or i depends on the direction of flow. Integration of equations (73) and (74) gives several different solutions for the airflow depending on the value of Y and the sign of $\Delta\rho$. Defining:

$$G \equiv \frac{2}{3} W C_d (2g|\Delta P|)^{1/2},$$

$$f_a \equiv \left| \frac{H}{2} - Y \right|^{3/2},$$

$$f_b \equiv \left| \frac{H}{2} + Y \right|^{3/2} \quad (75)$$

gives the following equations for flows.

Case	$\Delta\rho > 0$	$\Delta\rho < 0$	
$Y \leq -H/2$	$F = -G\sqrt{\rho_i} f_z - f_b $	$F = G\sqrt{\rho_j} f_z - f_b $	(76)
$Y \geq H/2$	$F = G\sqrt{\rho_j} f_z - f_b $	$F = -G\sqrt{\rho_i} f_z - f_b $	(77)
$-H/2 < Y < H/2$	$F_a = -G\sqrt{\rho_j} f_a$	$F_a = G\sqrt{\rho_j} f_a$	(78)
	$F_b = G\sqrt{\rho_j} f_b$	$F_b = -G\sqrt{\rho_j} f_b$	(79)

6 References

- ACM 1982. ACM-Transactions. Mathematical Software, Vol.8, No. 2, June 1982, p. 190.
- Achakji, G.Y. & G.T. Tamura, 1988. "Pressure Drop Characteristics of Typical Stairshafts in High-rise Buildings", *ASHRAE Transactions*, 94(1): 1223-1236.
- AIVC 1991. *Technical Note AIVC 34 – Airflow Patterns within Buildings: Measurement Techniques*, Coventry UK.
- ASHRAE 1993. *ASHRAE Handbook - 1993 Fundamentals*, Atlanta GA.
- ASHRAE 1999a. Method of Testing General Ventilation Air-Cleaning Devices for Removal Efficiency by Particle Size (ANSI approved). *Standard 52.2-1999*. Atlanta, GA.
- ASHRAE 1999b. *ASHRAE Handbook - HVAC Applications*, Atlanta GA.
- ASHRAE 2001. *ASHRAE Handbook - 2001 Fundamentals*, Atlanta GA.
- ASHRAE 2002. *Duct Fitting Database*, v. 2.2.5, Atlanta GA.
- ASHRAE 2004. *ASHRAE Handbook - 2004 HVAC Systems and Equipment*, Atlanta GA.
- ASHRAE 2005. *ASHRAE Handbook - 2005 Fundamentals*, Atlanta GA.
- ASTM 1999. Standard Test Method for Determining Air Leakage Rate by Fan Pressurization. *Standard E779-99*. American Society for Testing and Materials, West Conshohocken, PA.
- Axley, J.W. 1987. "Indoor Air Quality Modeling Phase II Report", NBSIR 87-3661, National Bureau of Standards (U.S.).
- Axley, J.W. 1988. "Progress Toward a General Analytical Method for Predicting Indoor Air Pollution in Buildings, Indoor Air Quality Modeling Phase III Report", NBSIR 88-3814, National Bureau of Standards (U.S.).
- Axley, J.W. 1991. "Adsorption Modeling for Building Contaminant Dispersal Analysis", *Indoor Air*, 1:147-171.
- Axley, J.W. 1995. "New Mass Transport Elements and Components for the NIST IAQ Model", NIST GCR 95-676, National Institute of Standards and Technology(U.S.).
- Baker, P.H., S. Sharples, & I.C. Ward. 1987. "Air Flow through Cracks," *Building and Environment*, Pergamon, 22(4): 293-304.
- Barakat, S.A. 1987. "Inter-zone Convective Heat Transfer in Buildings: a Review," *ASME Journal of Solar Engineering*, Vol. 109, May.
- Blevins, R.D. 1984. *Applied Fluid Dynamics Handbook*, New York: Van Nostrand Reinhold.
- Brown, W.G., & K.R. Solvason. 1962. "Natural Convection through Rectangular Openings in Partitions - 1: Vertical Partitions," *International Journal of Heat and Mass Transfer*, Vol. 5, pp. 859-867.
- Clarke, J.A. 1985. *Energy Simulation in Building Design*, Adam Hilger Ltd., Briston and Boston, p 204.
- Conte, S.D. and C. de Boor. 1972. *Elementary Numerical Analysis*, McGraw- Hill, New York NY.

- Dennis, J.E., Jr. and R.B. Schnabel. 1996. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Society for Industrial and Applied Mathematics, Philadelphia.
- Dhatt, G., G. Touzot, & G. Catin. 1984. *The Finite Element Method Displayed*, John Wiley & Sons, New York.
- Dols, W.S., Walton G.N., & Denton, K.R. 2000. "CONTAMW 1.0 User Manual", NISTIR 6476, National Institute of Standards and Technology.
- Emmerich, S.J. and A.K. Persily. 1996. "Multizone Modeling of Three Residential Indoor Air Quality Control Options", NISTIR 5801, National Institute of Standards and Technology.
- Emmerich, S.J. and A.K. Persily. 1998. "Energy Impacts of Infiltration and Ventilation in U.S. Office Buildings Using Multizone Airflow Simulation" Proceeding of *IAQ and Energy 98*. New Orleans, Louisiana: ASHRAE.
- Emmerich, S.J. and S.J. Nabinger. 2000. "Measurement and Simulation of the IAQ Impact of Particle Air Cleaners in a Single-Zone Building", NISTIR 6461, National Institute of Standards and Technology.
- Emmerich, S.J., JE Gorfain, M. Huang, C. Howard-Reed. 2003. "Air and Pollutant Transport from Attached Garages to Residential Living Spaces," NISTIR 7072, National Institute of Standards and Technology, Gaithersburg, MD. December 2003.
- Fang, J.B. and A.K. Persily. 1995. "Computer Simulations of Airflow and Radon Transport in Four Large Buildings", NISTIR 5611, National Institute of Standards and Technology.
- EPA 1989. *Exposure Factors Handbook*. Publication number EPA/600/8- 89/043.
- Feustel, H.E. & A. Rayner-Hooson (Eds). 1990. "COMIS Fundamentals", Lawrence Berkeley Laboratory, report LBL-28560.
- Feustel, H.E. & J. Dieris. 1992. "A survey of airflow models for multizone structures", *Energy and Buildings*, Vol 18, pp 79-100.
- Isaacs, L.T. & K.G. Mills. 1980. "Linear Theory Methods for Pipe Network Analysis", *Journal of the Hydraulics Division*, Proceedings ASCE, Vol. 106, pp. 1191-1201. (See also author's closure, 1982, Vol. 108, p. 153).
- Klote, J.H. 1982. "A Computer Program for Analysis of Smoke Control Systems", NBSIR 82-2512, National Bureau of Standards (U.S.).
- Klote, J.H. & J.A. Milke. 2002. *Principles of Smoke Management*, ASHRAE, Atlanta GA.
- Musser, A. and G. Yuill. 1999. "Comparison of Residential Air Infiltration Rates Predicted by Single-Zone and Multizone Models", *ASHRAE Transactions*, 1999. Vol. 105(Part 1).
- Musser, A. 2000. "Multizone Modeling as an Indoor Air Quality Design Tool", *Proceedings of Healthy Buildings 2000*. Espoo, Finland.
- Osborne, W.C., 1977. *Fans*, 2nd edition, Pergamon Press.
- Patankar, S.V., 1980. *Numerical Heat Transfer and Fluid Flow*, Hemisphere Publishing. ISBN: 0-89116-522-3.
- Persily, A.K. 1998. "A Modeling Study of Ventilation, IAQ and Energy Impacts of Residential Mechanical Ventilation", NISTIR 6162, National Institute of Standards and Technology.

- Persily, A.K. and S.R. Martin 2000. "A Modeling Study of Ventilation in Manufactured Houses", NISTIR 6455, National Institute of Standards and Technology.
- Persily, A.K. and E.M. Ivy. 2001. "Input Data for Multizone Airflow and IAQ Analysis", NISTIR 6585, National Institute of Standards and Technology.
- Press, W.H., S.A. Teukolsky, W.T. Vetterling & B.P. Flannery. 1992. *Numerical Recipes in C: the Art of Scientific Computing, Second Edition*, Cambridge University Press.
- Roulet, C-A, and L. Vandaele, *Airflow Patterns Within Buildings Measurement Techniques*, The Air Infiltration and Ventilation Centre, Technical Note 34, 1991, pp. iii.10 and iv.3.
- Stoecker, W.F. and P.A. Stoecker. 1989. *Microcomputer Control of Thermal and Mechanical Systems*. Van Nostrand Reinhold, New York.
- Versteeg, H.K., and W. Malalasekera, 1995. *An Introduction to Computational Fluid Dynamics : The Finite Volume Method*, Pearson Education Ltd. ISBN: 0-582-21884-5.
- Walker, I.S., and D.J. Wilson 1994. "Practical Methods for Improving Estimates of Natural Ventilation Rates", Proceedings 15th AIVC Conference, Vol. 2, pp. 517-525.
- Walton, G.N. 1989a. "AIRNET - A Computer Program for Building Airflow Network Modeling", NISTIR 89-4072, National Institute of Standards and Technology.
- Walton, G.N., 1989b. "Airflow Network Models for Element-Based Building Airflow Modeling", *ASHRAE Transactions*, Vol. 95, Pt. 2.
- Walton, G.N. 1997. "CONTAM96 User Manual", NISTIR 6056, National Institute of Standards and Technology.
- Walton, G.N. and W.S. Dols. 2003. "CONTAM 2.1 Supplemental User Guide and Program Documentation," NISTIR 7049, National Institute of Standards and Technology.
- Weber, D.D, and R.J. Kearney, 1980. *Natural Convective Heat Transfer Through an Aperture in Passive Solar Heated Buildings*, 5th National Passive Solar Conference, pp. 1037-1041.
- Wen, C.Y. and L.T. Fan, 1975. *Models for Flow Systems and Chemical Reactors*, Marcel Dekker, Inc. New York. ISBN: 0-8247-6346-7.
- Wray, C.P. and G.K. Yuill. 1993. "An Evaluation of Algorithms for Analyzing Smoke Control Systems", *ASHRAE Transactions*, Vol. 99, Pt 1.

APPENDICES

Appendix A PRJ File Format

CONTAM consists of two separate programs: ContamW provides the graphic user interface for describing the building and viewing simulation results; ContamX performs the simulation. "Project" files (.PRJ extension) provide input to both ContamW and ContamX. Project files are created by the functions in file prjsave.c in ContamW and read by the functions in file prjread.c in ContamW and ContamX. The project file contains some information that is used only by ContamW and not by ContamX, e.g., input units and the SketchPad display data. These data are read when the macro name CTMW is defined to be 1, and they are not read when it is defined to be 0.

Most of the data are read into structures that are defined in the files contam.h and celmts.h for ContamW and files simdat.h and selmts.h for ContamX. Some data are read into global variables that are defined in cglob.h and cextrn.h for ContamW and sglob.h and sextrn.h for ContamX. These variables are indicated by a leading underscore, `_`, in their name.

One important feature of the project files is the tremendous variety of data stored. It has been divided into sections grouping similar kinds of data, or objects, together. Each section is terminated with the special value `-999` which serves as a check for some reading errors. Within each section there are usually multiple objects, each stored into an individual structure. Such a section begins with the number of objects and starts the data for each object with a sequence number that serves to check for reading errors. These objects may be ordered as arrays when the total number is constant, or as linked lists to allow the number to vary.

In the following material individual variables are briefly described. Units are indicated within brackets, []. The type of variable is in parentheses (matching the definitions in the include file types.h).

- I1 – a one-byte signed integer (C type "char")
- I2 – a two-byte signed integer (C type "short")
- I4 – a four-byte signed integer (C type "long")
- IX – a default length signed integer (C type "int")
- UX – a default length unsigned integer (C type "unsigned")
- R4 – a four-byte real (C type "float")

Variables used only by ContamW are indicated by {W}. Items that appear in red are the most recently modified items including those whose data size has been increased from 2-bytes to 4-bytes to allow for an increased number of building components. This change took place between versions 2.3 and 2.4.

Comments are included in the project file to make it somewhat human-readable. The exclamation mark, "!", serves as a comment character. Anything after an exclamation mark to the end of the line is not read by the input processor. Any user-defined comments within a PRJ file will not be maintained when the file is saved by ContamW.

Sections of the PRJ File

Section 1: Project, Weather, Simulation, and Output Controls

Section 2: Species and Contaminants

Section 3: Level and Icon Data

Section 4: Day Schedules

Section 5: Week Schedules

Section 6: Wind Pressure Profiles

Section 7: Kinetic Reactions

Section 8a: Filter Elements

Section 8b: Filters

Section 9: Source/Sink Elements

Section 10: Airflow Elements

Section 11: Duct Elements

Section 12a: Control Super Elements

Section 12b: Control Nodes

Section 13: Simple Air Handling System (AHS)

Section 14: Zones

Section 15: Initial Zone Concentrations

Section 16: Airflow Paths

Section 17: Duct Junctions

Section 18: Initial Junction Concentrations

Section 19: Duct Segments

Section 20: Source/Sinks

Section 21: Occupancy Schedules

Section 22: Exposures

Section 23: Annotations

Section 1: Project, Weather, Simulation, and Output Controls

The initial section of the project file is read by the `run_read()` function and saved by the `run_save()` function.

The first line of the project file identifies the source program:

```
_string[] // program name "ContamW" (I1)
_string[] // program version "2.4a" (I1)
_echo     // (0/1) echo input files to the log file (I2)
```

The second line is a user-defined description of the project:

```
_prjdesc[] // (unused) project description (I1) {W}
```

The next line sets the following miscellaneous values:

```
_skwidth // total SketchPad width [cells] (I2) {W}
_skheight // total SketchPad height [cells] (I2) {W}
_def_units // default units: 0 = SI, 1 = US (I2) {W}
_def_flows // default flows: 0 = mass, 1 = volume (I2) {W}
_def_T // default temperature for zones [K] (R4) {W}
_udeft // units for temperature (I2) {W}
_rel_N // angle to true north [degrees] (R4) {W}
_wind_H // elevation for reference wind speed [m] (R4) {W}
_uwH // units for _wind_H (I2) {W}
_wind_Ao // local terrain constant (R4) {W}
_wind_a // velocity profile exponent (R4) {W}
```

The next line defines the weather (WTHDAT) for steady-state simulation in ContamX:

```
Tambt // ambient temperature [K] (R4)
barpres // barometric pressure [Pa] NOT corrected to sea level (R4)
windspd // wind speed [m/s] (R4)
winddir // wind direction: 0 = N, 90 = E, 180 = S, ...; (R4)
relhum // relative humidity: 0.0 to 1.0 (R4)
daytyp // type of day (1-12) (I2)
uTa // units for Tambt (I2) {W}
ubP // units for barpres (I2) {W}
uws // units for windspd (I2) {W}
uwd // units for winddir (I2) {W}
```

The next line defines weather data (WTHDAT) for the wind pressure test in ContamW:

```
Tambt // ambient temperature [K] (R4) {W}
barpres // barometric pressure [Pa] NOT corrected to sea level (R4) {W}
windspd // wind speed [m/s] (R4) {W}
winddir // wind direction: 0 = N, 90 = E, 180 = S, ...; (R4) {W}
relhum // relative humidity: 0.0 to 1.0 (R4) {W}
daytyp // type of day (1-12) (I2) {W}
uTa // units for Tambt (I2) {W}
ubP // units for barpres (I2) {W}
uws // units for windspd (I2) {W}
uwd // units for winddir (I2) {W}
```

The next four lines define the weather, contaminant and controls files, respectively:

```

_WTHpath[_MAX_PATH] // full name of weather file (I1)
_CTMpath[_MAX_PATH] // full name of contaminant file (I1)
_CVFpath[_MAX_PATH] // full name of continuous values file (I1)
_DVFpath[_MAX_PATH] // full name of discrete values file (I1)

```

The following lines define path location data (PLDDAT) for the creating the WPC file:

three lines for file paths and WPC description:

```

WPCfile[_MAX_PATH] // full name of WPC file (I1)
EWCfile[_MAX_PATH] // full name of EWC data source file (I1) {W}
WPCdesc[] // WPC description (I1) {W}

```

next a line describing reference coordinates:

```

X0 // X-value of ContamW origin in EWC coordinates [m] (R4) {W}
Y0 // Y-value of ContamW origin in EWC coordinates [m] (R4) {W}
Z0 // Z-value of ContamW origin in EWC coordinates [m] (R4) {W}
angle // Rotation of ContamW relative to EWC coordinates (R4) {W}
u_XYZ // units of coordinates (I2) {W}

```

then some EWC to WPC conversion parameters:

```

epsPath // tolerance for matching path locations [-] (R4) {W}
epsSpcs // tolerance for matching species [-] (R4) {W}
tShift // time shift of EWC data {W} [s] (hh:mm:ss → IX) {W}
dStart // date WPC data starts (I2) {W}
dEnd // date WPC data ends (I2) {W}
_useWPCwp // if true, use WPC file wind pressures (I2)
_useWPCmf // if true, use WPC file mass fractions (I2)

```

The next line defines the location (LOC DAT) (for future use with thermal simulation):

```

latd // latitude (degrees: north +, south -) (R4)
lgt d // longitude (degrees: east +, west -) (R4)
Tznr // time zone (Greenwich = 0, Eastern = -5, etc.) (R4)
altd // elevation above sea level [m] (R4)
Tgrnd // ground temperature [K] (R4)
utg // units for ground temperatures (I2)
u_a // units for elevation (I2)

```

The remaining data is stored in the run control (RCDAT) structure. In ContamX some values may be transferred to other variables before being used. The next two lines control the airflow simulation – first the nonlinear part:

```

sim_af // airflow simulation: 0 = steady, 1 = dynamic (I2)
afcalc // N-R method for non-linear eqns: 0 = SUR, 1 = STR (I2)
afmaxi // maximum number of N-R iterations (I2)
afrcnvg // relative airflow convergence factor (R4)
afacnvg // absolute airflow convergence factor [1/s] (R4)
afrelax // flow under-relaxation coefficient (for SUR) (R4)
uac2 // units for afacnvg (I2)
Pres // pressure test pressure (R4) {Contam 2.4}
uPres // units of Pres (I2) {Contam 2.4}

```

and then the linear part:

```

afslae // method for linear equations: 0 = SKY, 1 = PCG (I2)

```

```

afrseq      // if true, resequence the linear equations (I2)
aflmaxi     // maximum number of iterations (PCG) (I2)
aflcnvg     // relative convergence factor for (PCG) (R4)
aflinit     // if true, do linear airflow initialization (I2)
Tadj        // if true, use temperature adjustment (I2)

```

The next three lines control the mass fraction calculation – first for cyclic simulation:

```

sim_mf      // mass fraction (contaminant) simulation:
             // 0 = none, 1 = steady, 2 = transient, 3 = cyclic (I2)
ccmaxi     // simulation: maximum number of cyclic iterations (I2)
ccrcnvg    // relative convergence factor (R4)
ccacnvg    // absolute convergence factor [kg/kg] (R4)
ccrelax    // (unused) over-relaxation coefficient (R4)
uccc       // units for ccacnvg (I2)

```

then for non-trace contaminants:

```

mfnmthd    // simulation: 0 = SKY, 1 = BCG, 2 = SOR (I2)
mfnrseq    // if true, resequence the linear equations (I2)
mfnmaxi    // maximum iterations (I2)
mfncnvg    // desired relative convergence (R4)
mfncnvg    // desired absolute convergence (R4)
mfrelax    // relaxation coefficient (R4)
mfngamma   // trapezoidal integration factor (R4)
uccn       // units for mfncnvg (I2)

```

and then for trace contaminants:

```

mftmthd    // 0 = SKY, 1 = BCG, 2 = SOR (I2)
mfttrseq   // if true, resequence the linear equations (I2)
mftmaxi    // maximum iterations (I2)
mftcnvg    // desired relative convergence (R4)
mftcnvg    // desired absolute convergence (R4)
mftrelax   // relaxation coefficient (R4)
mftgamma   // trapezoidal integration factor (R4)
ucct       // units for mftcnvg (I2)

```

and then for transient contaminants by the short time step method:

```

sim_sts    // if true, transient m.f. by short time step method (I2)
sim_ldz    // if true, use 1D zones (I2)
sim_ldd    // if true, use 1D ducts (I2)
celldx     // default length of duct cells for C-D model [m] (R4)
sim_vjt    // if true, compute variable junction temperatures (I2)
udx        // units of celldx (I2)

```

The next line has four parameters; only the ones relating to density changes are active:

```

tsdens     // (0/1) vary density during time step (I2)
tsrelax    // (inactive) under-relaxation factor for calculating dM/dt (R4)
tsmaxi     // maximum number of iterations for density changes (I2)

```

The next line sets the dates, times, and time steps for simulation:

```

date_st    // day-of-year to start steady simulation (mmmdd → IX)
time_st    // time-of-day to start steady simulation (hh:mm:ss → I4)
date_0     // day-of-year to start transient simulation (mmmdd → IX)
time_0     // time-of-day to start transient simulation (hh:mm:ss → I4)

```

```

date_1      // day-of-year to end transient simulation (mmdd → IX)
time_1      // time-of-day to end transient simulation (hh:mm:ss → I4)
time_step   // simulation time step [s] (hh:mm:ss → IX)
time_list   // simulation output (results) time step [s] (hh:mm:ss → IX)
time_scrn   // simulation status time step [s] (up to 1 day) (hh:mm:ss → I4)

```

The next line controls the use of the restart file:

```

restart     // use restart file (I2)
rstdate    // restart date (mmdd → IX)
rsttime    // restart time (hh:mm:ss → I4)

```

The remaining parameters control the simulation outputs. They are in global variables in ContamX and the save[] vector in ContamW for flexibility. They may be spread over several lines:

```

_list      // data dump parameter (I2)
           // > 0  dump matrix analysis,
           // = 2  dump SIM file output,
           // > 2  dump lognotes.

_doDlg     // (0/1) ContamX will display dialog box (I1)
_pfsave   / save[0] // (0/1) save path flow results to SIM file (I1)
_zfsave   / save[1] // (0/1) save zone flow results to SIM file (I1)
_zcsave   / save[2] // (0/1) save mass fraction results to SIM file (I1)
_achvol   / save[3] // (0/1) ACH based on true volumes instead of std volumes
_achsave  / save[4] // (0/1) save building air exchange rate transient data
_abwsave  / save[5] // (0/1) save air exchange rate box-whisker data
_cbwsave  / save[6] // (0/1) save contaminant box-whisker data
_expsave  / save[7] // (0/1) save exposure transient data
_ebwsave  / save[8] // (0/1) save exposure box-whisker data
_zasave   / save[9] // (0/1) save zones age-of-air transient data
_zbwsave  / save[10] // (0/1) save zones age-of-air box-whisker data
save[11-31] // (unused; subject to change without notice) (I1)

```

The next line controls which sections are output to the .val file:

```

BldgFlowZ // output building airflow test (zones) (IX)
BldgFlowD // output building airflow test (ducts) (IX)
BldgFlowC // output building airflow test (classified flows) (IX)

```

The run control section is terminated with:

```

-999      // used to check for a read error in the above data

```

Note on outputs: `_pfsave` and `_zfsave` always have the same value – 0 or 1.

Example:

```

ContamW 2.4a 1
Test CONTAMW for proper display of all PLR_ type af elements.
! rows cols ud uf  T  uT  N   wH  u  Ao  a
    58  66  0  0 296.150 2   0.00 10.00 0 0.600 0.280
! Ta      Pb      Ws      Wd      rh  day u..
293.150 101325.0 0.000   0.0 0.500 1 2 0 0 1 ! steady simulation
293.150 101325.0 1.000 270.0 0.000 1 2 0 0 1 ! wind pressure test
null ! no weather file
null ! no contaminant file
null ! no continuous values file

```



```

mdiam      // mean diameter - particle [m] (R4)
edens      // effective density - particle [kg/m^3] (R4)
decay      // decay constant [1/s] (R4) {W}
Dm         // molecular diffusion coefficient [m^2/s] (R4)
ccdef      // default concentration [kg/kg air] (R4)
Cp         // (unused) specific heat at constant pressure [J/kgK] (R4)
ucc        // units to display concentration (I2) {W}
umd        // units to display mean diameter (I2) {W}
ued        // units to display effective density (I2) {W}
udm        // units to display diffusion coefficient (I2) {W}
ucp        // units to display specific heat (I2) {W}
name[]     // species name (I1)

```

The second line is:

```
desc[]     // species description (I1) {W}
```

The section is terminated with:

```
-999      // used to check for a read error in the above data (I2)
```

Example:

```

2 ! contaminants:
  1 2
4 ! species:
! # s t molwt mdiam edens decay Dm CCdef Cp u... name
1 1 0 44.0000 0.000e+000 0.000e+000 0.000e+000 2.000e-005 5.3184e-004 1000.00 1 0 0 0 0 C1
2 1 0 44.0000 0.000e+000 0.000e+000 0.000e+000 2.000e-005 6.8379e-004 1000.00 1 0 0 0 0 C2
3 0 1 28.9645 0.000e+000 0.000e+000 0.000e+000 2.000e-005 1.0000e+000 0.00 0 0 0 0 0 DryAir
Dry Air non-trace
4 0 1 18.0153 0.000e+000 0.000e+000 0.000e+000 2.000e-005 5.0000e-003 0.00 0 0 0 0 0 H2O
Water Vapor non-trace
-999

```

Section 3: Level and Icon Data

The data for recreating the SketchPad is stored in this section. The data are read by the `level_read()` function and saved by the `level_save()` function. The level data are stored in `LEV_DATA` structures that are defined in `contam.h` for `ContamW` and `simdat.h` for `ContamX`.

The icon data are stored in `ICON_DAT` structures in `ContamW` and are not saved in `ContamX`.

The level/icons section starts with:

```
nlev      // number of levels (IX)
```

This is followed by a data header comment line and then data for all nlev levels.

Each level has a data line that includes:

```

nr        // level number (IX), in order from 1 to nlev
refht     // reference elevation of level [m] (R4)
delht     // delta elevation to next level [m] (R4) {W}
nicon     // number of icons on this level (IX)
u_rfht    // units of reference elevation (I2) {W}
u_dlht    // units of delta elevation (I2) {W}

```

```
name[] // level name (I1)
```

This line is followed by a comment line and then data for nicon icons.

Each icon has a data line consisting of:

```
icon // icon type - see 'special symbols' in contam.h (I2) {W}
row // row position on the SketchPad (I2) {W}
col // column position on the SketchPad (I2) {W}
nr // zone, path, duct, etc., number (I2) {W}
```

The section is terminated with:

```
-999 // used to check for a read error in the above data (I2)
```

Example:

```
2 ! levels plus icon data:
! # refHt delHt ni u name
 1  0.000  3.000 44 0 0 one
!icn col row #
 14 15 17 0
 23 19 17 5
 23 21 17 6
 23 23 17 7
 23 25 17 8
 23 27 17 9
...
 16 49 33 0
 42 19 34 6
 2  3.000  3.000 78 0 0 <3>
!icn col row #
162 10 14 1
145 11 14 0
...
185 32 38 7
185 33 38 8
-999
```

Section 4: Day Schedules

Day schedule data are read by the `dschd_read()` function and saved by the `dschd_save()` function. The data are stored in the `DY_SCHD` structures that are defined in `contam.h` for `ContamW` and `simdat.h` for `ContamX`.

The day schedules section starts with:

```
_ndsch // number of day schedules (IX)
```

This is followed by a data header comment line and then data for all `_ndsch` schedules.

For each schedule the first data line includes:

```
nr // schedule number (IX); in order from 1 to _ndsch
npts // number of data points (I2)
shape // 0 = rectangular; 1 = trapezoidal (I2)
utyp // type of units (I2) {W}
ucnv // units conversion (I2) {W}
name[] // schedule name (I1) {W}
```

and the second line has:

```
desc[] // schedule description (I1) {W} may be blank
```

This is followed by a line for each of the npts data points:

```
time // time-of-day [s] (hh:mm:ss converted to I4)
ctrl // corresponding control value (R4) [-]
```

The section is terminated with:

```
-999 // used to check for a read error in the above data
```

Example:

```
1 ! day-schedules:
! # npts shap utyp ucnv name
  1  2  0  1  0 DySchd1
On all day
  00:00:00 1
  24:00:00 1
-999
```

Section 5: Week Schedules

Week schedule data are read by the `wschd_read()` function and saved by the `wschd_save()` function. The data are stored in the `WK_SCHD` structures that are defined in `contam.h` for `ContamW` and `simdat.h` for `ContamX`.

The week schedules section starts with:

```
_nwsch // number of week schedules (IX)
```

This is followed by a data header comment line and then data for all `_nwsch` schedules.

For each week schedule the first data line includes:

```
nr // schedule number (IX); in order from 1 to _nwsch
utyp // type of units (I2) {W}
ucnv // units conversion (I2) {W}
name[] // schedule name (I1) {W}
```

and the second line has:

```
desc[] // schedule description (I1) {W} may be blank
```

and the third line has:

```
j1 ... j12 // 12 day schedule indices (IX) - converted to pointers
```

The section is terminated with:

```
-999 // used to check for a read error in the above data
```

Example:

```
1 ! week-schedules:
! # utyp ucnv name
  1  1  0 WkSchd1
On all week
  1 1 1 1 1 1 1 1 1 1 1 1
-999
```

Section 6: Wind Pressure Profiles

Wind pressure profile data are read by the `wind_read()` function and saved by the `wind_save()` function. The data are stored in the `WIND_PF` structures that are defined in `contam.h` for ContamW and `simdat.h` for ContamX.

The wind pressure profiles section starts with:

```
_nwpf      // number of wind pressure profiles (IX)
```

This is followed by a data header comment line and then data for all _nwpf profiles.

For each wind pressure profile the first data line includes:

```
nr          // profile number (IX); in order from 1 to _nwpf
npts        // number of data points (I2)
type        // 1 = linear; 2 = cubic spline; 3 = trigonometric (I2)
name[]      // schedule name (I1) {W}
```

and the second line has:

```
desc[]      // profile description (I1) {W} may be blank
```

This is followed by a line for each of the npts data points that are used to compute the cubic spline or trigonometric coefficients for wind pressure profiles:

```
azm[]       // wind azimuth value {R4} [degrees]
coef[]      // normalized wind pressure coefficients {R4} [-]
```

The section is terminated with:

```
-999        // used to check for a read error in the above data
```

Example:

```
2 ! wind pressure profiles:
```

```
1 5 1 WindProf1
```

```
Wind pressure profile one.
```

```
0.0 1.000
90.0 0.500
180.0 1.000
270.0 0.500
360.0 1.000
```

```
2 9 1 WindProf2
```

```
Wind pressure profile two.
```

```
0.0 1.000
45.0 0.800
90.0 0.200
135.0 0.800
180.0 1.000
225.0 0.800
270.0 0.200
315.0 0.800
360.0 1.000
```

```
-999
```

Section 7: Kinetic Reactions

Kinetic reaction data are read by the `kinetic_read()` function and saved by the `kinetic_save()` function. The data are stored in the `KNR_DSC` structures that are defined in `contam.h` for ContamW and `simdat.h` for ContamX.

The kinetic reactions section starts with:

`_nkinr` // number of kinetic reactions (IX)
 This is followed by a data header comment line and then data for all `_nkinr` reactions.

For each reaction the first data line includes:

```
nr          // reaction number (IX); in order from 1 to _nkinr
nkrd       // number of reactions (I2)
name[]     // reaction name (I1) {W}
```

and the second line has:

```
desc[]     // reaction description (I1) {W} may be blank
```

This is followed by `nkrd` lines of reaction data:

```
prod[]     // product species name (I1)
src[]      // source species name (I1)
coef       // reaction coefficient (R4)
```

The kinetic reactions section is terminated with:

```
-999      // used to check for a read error in the above data
```

Example:

```
1 ! kinetic reactions:
1 2 Kinr1
Kinetic reaction one.
  C1 C1 0.5
  C2 C2 0.5
-999
```

Section 8a: Filter Elements

Filter element data are read by the `flte_read()` function and saved by the `flte_save()` function. The data are stored in the `FLT_ELT` structures that are defined in `contam.h` for `ContamW` and `simdat.h` for `ContamX`.

The filter element section starts with:

```
_nflte     // number of filter elements (IX)
```

This is followed by data for all `_nflte` filters.

For each element the first data line includes:

```
nr          // element number (IX); in order from 1 to _nflte
ftype       // element data type (string ? I2)
            // type names are stored in _flte_names in ftype order.
area        // face area of filter [m] (R4)
depth       // depth of filter [m] (R4)
dens        // density of filter material [kg/m^3] (R4)
            // note: area * depth * dens = filter mass [kg]
ual         // units of area and depth (I2) {W}
ud          // units of density (I2) {W}
name[]     // element name (I1) {W}
```

and the second line has:

```
desc[]     // filter description (I1) {W} may be blank
```

This is followed by lines of data that depend on the element data type.

The introductory lines that follow give the type number, the program macro defined as that number, the type name from `_flte_names`, and the data structures to hold the data. The data

structures are defined in *celmts.h* in *ContamW* and *selmts.h* in *ContamX*. Type numbers can change as long as the ordering in *_flte_names* and related arrays reflect that order.

Element type 0 [FL_CEF] "cef" stored in structure CEF_DAT.

The constant efficiency filter data consist of:

```
nspcs      // number of species (IX)
```

This is followed by *nspcs* lines of filter efficiency data:

```
spcs[]     // species name (I1)
eff        // filter efficiency [-] (R4)
```

Element type 1 [FL_PF0] "pf0" stored in structure PF0_DAT.

The simple particle filter data consist of:

```
npts       // number of data points (I2)
usz        // units of particle size (I2) {W}
```

This is followed by *npts* lines of filter efficiency data:

```
size       // particle size [m] (R4)
eff        // filter efficiency [-] (R4)
```

Element type 2 [FL_GF0] "gf0" stored in structure GF0_DAT.

The simple gas filter data consist of:

```
nspcs      // number of species (IX)
```

This is followed by *nspcs* sections consisting of:

```
spcs[]     // species name (I1)
bthru[]    // break through eff (R4) {Contam 2.4}
npts       // number of data points (I2)
```

Each of which is followed by *npts* lines of filter efficiency data:

```
load       // relative total loading [-] (R4)
eff        // filter efficiency for this species[-] (R4)
```

Element type 3 [FL_SPF] "spf" stored in structure SPF_DAT.

The super filter data consist of:

```
nselmt     // number of sub-elements(I2)
```

Then a line of data containing a list of the number of each sub-element in flow order:

```
elmt[]     // sub-element numbers (IX) - converted to pointers
```

The filter element section is terminated with:

```
-999      // used to check for a read error in the above data
```

Example:

```
4 ! filter elements:
1 cef 1 0.1 100 0 0 CEF
Constant efficiency filter
1
  ctm1 0.5
2 gf0 1 0.1 100 0 0 GF0
Simple gaseous filter
1
  ctm1 12
  0 1
  0.01 1
```

```

. . .
0.028 0
0.03 0
3 pf0 1 0.1 100 0 0 MERV_10
MERV-10 filter
16 0
1e-009 0.999
1.5e-009 0.999
. . .
6e-006 0.918
1e-005 0.92
4 spf 1 0.1 100 0 0 SF1
Super filter - MERV10 and Gaseous
2
3 2
-999

```

Section 8b: Filters

Filter data are read by the `filter_read()` function and saved by the `filter_save()` function. The data are stored in the `FLT_DSC` structures that are defined in `contam.h` for ContamW and `simdat.h` for ContamX.

The filter section starts with:

```
_nfilt // number of filters (IX)
```

This is followed by data for all `_nfilt` filters.

For each filter the first data line includes:

```
nr // filter number (IX); in order from 1 to _nfilt
fe // filter element number (IX)
nsub // number of sub-elements, 1 if not super filter (IX)
```

This is followed by `nsub` lines of initial total relative loadings:

```
load // initial relative loading [-] (R4)
nload // number of species loadings (I2) [0]
```

This is followed by `nload` lines of species loading data: [not yet used; `nload = 0`]

```
spcs[] // species name (I1)
load // initial relative loading [-] (R4)
```

Example:

```
2 ! filters:
1 4 2
0.11 0
0.22 0
2 2 1
0.00 0
-999
```

Section 9: Source/Sink Elements

Source/sink elements are read by the `cselmt_read()` function and saved by the `cselmt_save()` function. The data are stored in the `CSE_DAT` structures that are defined in `contam.h` for ContamW and `simdat.h` for ContamX. `CSE_DAT` includes a pointer to the element specific data structure.

The source/sink elements section starts with:

`_ncse` // number of source/sink elements (IX)

This is followed by a data header comment line and then data for all `_ncse` elements.

For each element the first data line includes:

```
nr          // element number (IX); in order from 1 to _ncse
spcs[]     // species name (I1)
ctype      // element data type (string ? I2)
            // element type names are stored in _cse_dnames in ctype
```

order.

```
name[]     // element name (I1) {W}
```

and the second line has:

```
desc[]    // element description (I1) {W} may be blank
```

This is followed by one or more lines of data that depend on the element data type.

The introductory lines that follow give the type number, the program macro defined as that number, the type name from `_cse_dnames`, and the data structures to hold the data. The data structures are defined in `celmts.h` in `ContamW` and `selmts.h` in `ContamX`. Type numbers can change as long as the ordering in `_csse_names` and related arrays reflect that order.

Element type 0 [CS_CCF] "ccf" stored in structure CSE_CCF.

The constant coefficient source model data:

```
G          // generation rate [kg/s] (R4)
D          // deposition rate [kg/s] (R4)
u_G       // units of generation (I2) {W}
u_D       // units of deposition (I2) {W}
```

Element type 1 [CS_PRS] "prs" stored in structure CSE_PRS.

The pressure driven source data:

```
G          // generation rate [kg/s] (R4)
x          // pressure exponent [-] (R4)
u_G       // units of generation (I2) {W}
```

Element type 2 [CS_CUT] "cut" stored in structure CSE_CUT.

The concentration cutoff model data:

```
G          // generation rate [kg/s] (R4)
Co         // cutoff concentration [kg/kg] (R4)
u_G       // units of generation (I2) {W}
u_C       // units of concentration (I2) {W}
```

Element type 3 [CS_EDS] "eds" stored in structure CSE_EDS.

The exponential decay model data:

```
G0         // initial generation rate [kg/s] (R4)
k          // decay constant [1/s] (R4)
u_G       // units of generation (I2) {W}
```

u_k // units of time (I2) {W}

Element type 4 [CS_BLS] "bls" stored in structure CSE_BLS.

The boundary layer diffusion model data:

hm // average film mass transfer coefficient [m/s] (R4)
 rho // film density of air [kg/m³] (R4)
 Kp // partition coefficient (R4)
 M // adsorbent mass/unit area [kg/m²] (R4)
 u_h // units of hm (I2) {W}
 u_r // units of rho (I2) {W}
 u_M // units of M (I2) {W}

Element type 5 [CS_BRS] "brs" stored in structure CSE_BRS.

The burst source data:

M // mass added to zone in one time step [kg] (R4)
 u_M // units of mass (I2) {W}

Element type 6 [CS_DVS] "dvs" stored in structure CSE_DVS.

The deposition velocity sink model data:

dV // deposition velocity [m/s] (R4)
 dA // deposition area [m²] (R4)
 u_V // units of velocity (I2) {W}
 u_A // units of area (I2) {W}

Element type 7 [CS_DRS] "drs" stored in structure CSE_DRS.

The deposition rate sink model data:

kd // deposition rate [1/s] (R4)
 u_k // units of deposition rate (I2) {W}

The source/sink element section is terminated with:

-999 // used to check for a read error in the above data

Example:

```
8 ! source/sink elements:
1 C1 ccf s1
Constant coefficient
  1 1 0 0
2 C1 prs s2
Pressure driven
  2 2 0
3 C1 cut s3
Cutoff concentration
  3 3 0 0
4 C1 eds s4
Decaying source
  4 4 0 0
5 C1 bls s5
Boundary layer diffusion
  5 5 5 0 0 0
```

```

6 C2 brs s6
Burst
 6 0
7 c1 dvs s7
Deposition velocity
 0.00194444 7 4 0
8 c1 drs s8
Deposition rate
 8 0
-999

```

Section 10: Airflow Elements

Airflow elements are read by the `afelmt_read()` function and saved by the `afelmt_save()` function. The data are stored in the `AFE_DAT` structures that are defined in `contam.h` for `ContamW` and `simdat.h` for `ContamX`. `AFE_DAT` includes a pointer to the element specific data structure.

The airflow elements section starts with:

```
_nafe // number of airflow elements (IX)
```

This is followed by a data header comment line and then data for all `_nafe` elements.

For each element the first data line includes:

```

nr // element number (IX); in order from 1 to _nafe
icon // icon used to represent flow path (R4) {W}
dtype // element data type (string ? I2)
// element type names are stored in _afe_dnames in dtype order.
name[] // element name (I1) {W}

```

and the second line has:

```
desc[] // element description (I1) {W} may be blank
```

This is followed by one or more lines of data that depend on the element data type.

The introductory lines that follow give the type number, the program macro defined as that number, the type name from `_afe_dnames`, and the data structures to hold the data. The data structures are defined in `celmts.h` in `ContamW` and `selmts.h` in `ContamX`. Type numbers can change as long as the ordering in `_afe_dnames` and related arrays reflect that order.

Element type 0 [PL_ORFC] "plr_orfc" stored in structure `PLR_ORF`.

The orifice data consist of:

```

lam // laminar flow coefficient (R4)
turb // turbulent flow coefficient (R4)
expt // pressure exponent (R4)
area // actual area [m^2] (R4) {X}
dia // hydraulic diameter [m] (R4) {X}
coef // flow coefficient (R4) {X}
Re // laminar/turbulent transition Reynolds number [-] (R4) {X}
u_A // units of area (I2) {X}
u_D // units of diameter (I2) {X}

```

Element type 1 [PL_LEAK1] "plr_leak1" stored in structure `PLR_LEAK`.

Element type 2 [PL_LEAK2] "plr_leak2" stored in structure PLR_LEAK.

Element type 3 [PL_LEAK3] "plr_leak3" stored in structure PLR_LEAK.

The leakage area data consist of:

```
lam      // laminar flow coefficient (R4)
turb     // turbulent flow coefficient (R4)
expt     // pressure exponent (R4)
coef     // flow coefficient (R4) {W}
pres     // reference pressure drop [Pa] (R4) {W}
area1    // leakage area per item [m^2] (R4) {W}
area2    // leakage area per unit length [m^2/m] (R4) {W}
area3    // leakage area per unit area [m^2/m^2] (R4) {W}
u_A1     // units of area1 [m^2] (I2) {W}
u_A2     // units of area2 [m^2/m] (I2) {W}
u_A3     // units of area3 [m^2/m^2] (I2) {W}
u_dP     // units of pressure (I2) {W}
```

Element type 4 [PL_CONN] "plr_conn" stored in structure PLR_CONN.

The (ASCOS compatible) connection data consist of:

```
lam      // laminar flow coefficient (R4)
turb     // turbulent flow coefficient (R4)
expt     // pressure exponent - 0.5 (R4)
area     // actual area [m^2] (R4) {W}
coef     // flow coefficient (R4) {W}
u_A      // units of area (I2) {W}
```

Element type 5 [PL_QCN] "plr_qcn" stored in structure PLR_QCN.

The volume flow powerlaw data consist of:

```
lam      // laminar flow coefficient (R4)
turb     // turbulent flow coefficient (R4)
expt     // pressure exponent (R4)
```

Element type 6 [PL_FCN] "plr_fcn" stored in structure PLR_FCN.

The mass flow powerlaw data consist of:

```
lam      // laminar flow coefficient (R4)
turb     // turbulent flow coefficient (R4)
expt     // pressure exponent (R4)
```

Element type 7 [PL_TEST1] "plr_test1" stored in structure PLR_TEST1.

The single test point powerlaw data consist of:

```
lam      // laminar flow coefficient (R4)
turb     // turbulent flow coefficient (R4)
expt     // pressure exponent (R4)
dP       // pressure drop [Pa] (R4) {W}
```

```

Flow      // flow rate [kg/s] (R4) {W}
u_P       // units of pressure drop {W}
u_F       // units of flow (I2) {W}

```

Element type 8 [PL_TEST2] "plr_test2" stored in structure PLR_TEST2.

The two test points powerlaw data consist of:

```

lam       // laminar flow coefficient (R4)
turb      // turbulent flow coefficient (R4)
expt      // pressure exponent (R4)
dP1       // point 1 pressure drop [Pa] (R4) {W}
F1        // point 1 flow rate [kg/s] (R4) {W}
dP2       // point 2 pressure drop [Pa] (R4) {W}
F2        // point 2 flow rate [kg/s] (R4) {W}
u_P1      // units of pressure drop (I2) {W}
u_F1      // units of flow (I2) {W}
u_P2      // units of pressure drop (I2) {W}
u_F2      // units of flow (I2) {W}

```

Element type 9 [PL_CRACK] "plr_crack" stored in structure PLR_CRACK.

The crack powerlaw data consist of:

```

lam       // laminar flow coefficient (R4)
turb      // turbulent flow coefficient (R4)
expt      // pressure exponent (R4)
length    // crack length [m] (R4) {W}
width     // crack width [m] (R4) {W}
u_L       // units of length (I2) {W}
u_W       // units of width (I2) {W}

```

Element type 10 [PL_STAIR] "plr_stair" stored in structure PLR_STAIR.

The stairwell powerlaw data consist of:

```

lam       // laminar flow coefficient (R4)
turb      // turbulent flow coefficient (R4)
expt      // pressure exponent (R4)
Ht        // distance between levels [m] (R4) {W}
Area      // cross-sectional area [m^2] (R4) {W}
peo       // density of people [pers/m^2] (R4) {W}
tread     // 1 = open tread 0 = closed {W}
u_A       // units of area (I2) {W}
u_D       // units of distance (I2) {W}

```

Element type 11 [PL_SHAFT] "plr_shaft" stored in structure PLR_SHAFT.

The shaft powerlaw data consist of:

```

lam       // laminar flow coefficient (R4)
turb      // turbulent flow coefficient (R4)

```

```

expt      // pressure exponent (R4)
Ht        // distance between levels [m] (R4) {W}
area      // cross-sectional area [m^2] (R4) {W}
perim     // perimeter [m] (R4) {W}
rough     // roughness [m] (R4) {W}
u_A       // units of area (I2) {W}
u_D       // units of distance (I2) {W}
u_P       // units of perimeter (I2) {W}
u_R       // units of roughness (I2) {W}

```

Element type 12 [PL_BDQ] "plr_bdq" stored in structure PLR_BDQ.

The volume flow powerlaw backdraft damper data consist of:

```

lam       // laminar flow coefficient {R4}
Cp        // turbulent flow coefficient ( dP > 0 ) {R4}
xp        // pressure exponent ( dP > 0 ) {R4}
Cn        // turbulent flow coefficient ( dP < 0 ) {R4}
xn        // pressure exponent ( dP < 0 ) {R4}

```

Element type 13 [PL_BDF] "plr_bdf" stored in structure PLR_BDF.

The mass flow powerlaw backdraft damper data consist of:

```

lam       // laminar flow coefficient {R4}
Cp        // turbulent flow coefficient ( dP > 0 ) {R4}
xp        // pressure exponent ( dP > 0 ) {R4}
Cn        // turbulent flow coefficient ( dP < 0 ) {R4}
xn        // pressure exponent ( dP < 0 ) {R4}

```

Element type 14 [QFR_QAB] "qfr_qab" stored in structure QFR_QAB.

The volume flow quadratic data consist of:

```

a         // dP = a*Q + b*Q*Q {R4}
b         // {R4}

```

Element type 15 [QFR_QAF] "qfr_fab" stored in structure QFR_FAB.

The mass flow quadratic data consist of:

```

a         // dP = a*F + b*F*F {R4}
b         // {R4}

```

Element type 16 [QFR_CRACK] "qfr_crack" stored in structure QFR_CRACK.

The crack mass flow quadratic data consist of:

```

a         // dP = a*F + b*F*F {R4}
b         // {R4}
length    // crack length [m] {R4}
width     // crack width [m] {R4}
depth     // crack depth [m] {R4}

```

```

nB      // number of bends
u_L     // units of length
u_W     // units of width
u_D     // units of depth

```

Element type 17 [QFR_TEST2] "qfr_test2" stored in structure QFR_TEST2.

The two test points mass flow quadratic data consist of:

```

a       // dP = a*F + b*F*F {R4}
b       // {R4}
dP1     // point 1 pressure drop [Pa] (R4) {W}
F1      // point 1 flow rate [kg/s] (R4) {W}
dP2     // point 2 pressure drop [Pa] (R4) {W}
F2      // point 2 flow rate [kg/s] (R4) {W}
u_P1    // units of pressure drop (I2) {W}
u_F1    // units of flow (I2) {W}
u_P2    // units of pressure drop (I2) {W}
u_F2    // units of flow (I2) {W}

```

Element type 18 [DR_DOOR] "dor_door" stored in structure AFE_DOR.

The single opening doorway data consist of:

```

lam     // laminar flow coefficient (R4)
turb    // turbulent flow coefficient (R4)
expt    // pressure exponent (R4)
dTmin   // minimum temperature difference for two-way flow [C] (R4)
ht      // height of doorway [m] (R4)
wd      // width of doorway [m] (R4)
cd      // discharge coefficient
u_T     // units of temperature (I2) {W}
u_H     // units of height (I2) {W}
u_W     // units of width (I2) {W}

```

Element type 19 [DR_PL2] "dor_pl2" stored in structure DR_PL2.

The double opening doorway data consist of:

```

lam     // laminar flow coefficient (R4)
turb    // turbulent flow coefficient (R4)
expt    // pressure exponent (R4)
dH      // distance above | below midpoint [m] {R4}
ht      // height of doorway [m] (R4) {W in v. 2.0}
wd      // width of doorway [m] (R4) {W}
cd      // discharge coefficient [-] (R4) {W}
u_H     // units of height (I2) {W}
u_W     // units of width (I2) {W}

```

Element type 20 [FN_CMF] "fan_cmf" stored in structure AFE_CMF.

The constant mass flow fan data consist of:

```
Flow      // design flow rate [kg/s] (R4)
u_F       // units of flow (I2) {W}
```

Element type 21 [FN_CVF] "fan_cvf" stored in structure AFE_CVF.

The constant volume flow fan data consist of:

```
Flow      // design flow rate [m^3/s] (R4)
u_F       // units of flow (I2) {W}
```

Element type 22 [FN_FAN] "fan_fan" stored in structure AFE_FAN.

The first line of performance curve fan data consists of:

```
lam       // laminar flow coefficient (R4)
turb      // turbulent flow coefficient (R4)
expt      // pressure exponent (R4)
rdens     // reference fluid density [kg/m^3] {R4}
fdf       // free delivery flow (prise = 0) [kg/s] {R4}
sop       // shut-off pressure (flow = 0) [Pa] {R4}
off       // fan is off if (RPM/rated RPM) < off {R4}
```

The second line consists of:

```
fpc[4]    // fan performance polynomial coefficients {R4}
npts      // number of mesasured data points (I2) {W}
Sarea     // shut-off orifice area [m^2] {R4} {W}
u_Sa     // units of shut-off area (I2) {W}
```

The next npts lines consists of:

```
mF        // measured flow rates [kg/s] (R4) {W}
u_mF     // units of measured flows (I2) {W}
dP        // measured pressure rises [Pa] (R4) {W}
u_dP     // units of pressure rises (I2) {W}
rP        // revised pressure rises [Pa] (R4) {W}
u_rP     // units of revised pressures (I2) {W}
```

Element type 23 [CS_FSP] "csf_fsp" stored in structure AFE_CSF.

Element type 24 [CS_QSP] "csf_qsp" stored in structure AFE_CSF.

Element type 25 [CS_PSF] "csf_psf" stored in structure AFE_CSF.

Element type 26 [CS_PSQ] "csf_psq" stored in structure AFE_CSF.

The first line of cubic spline fit element data consists of:

```
npts // number of data points (I2)
u_x  // units for x (I2)
u_y  // units for y (I2)
```

followed by npts lines of x, y pairs of data

```
x // value of independent variable (R4)
y // value of dependent variable (R4)
```

Element type 27 [AF_SUP] "sup_afe" stored in structure AFE_SUP.

The first line of super airflow element data consists of:

```
nse // number of sub-elements (I2)
sched // scheduled sub-element number (IX) (only one)
u_H // units of relative height (I2)
    followed by nse lines of sub-element data
    nr // sub-element number (IX)
    relHt // relative height of sub-element (R4)
    filt // 1= filtered, 0= not (I2)
```

The airflow element section is terminated with:

```
-999 // used to check for a read error in the above data
```

Example:

```
27 ! flow elements:
1 24 dor_door DR_DOOR
DR_DOOR Single opening w/ 2-way flow
0.0741669 1.76494 0.5 0.01 2 0.8 0.78 0 0 0
2 24 dor_pl2 DR_PL2
DR_PL2 Two-opening model
0.0185417 0.882469 0.5 0.444444 2 0.8 0.78 0 0
3 30 fan_cmf FN_CMF
FN_CMF Constant mass flow fan model
1 0
4 29 fan_cvf FN_CVF
FN_CVF Constant volume flow fan model
1 0
5 30 fan_fan FN_FAN
FN_FAN Cubic polynomial fan model
7.2e-006 0.00848528 0.5 1.2041 -0.377789 0.0286444 0.1
0.0286444 0.0745766 -0.00327397 5.29862e-005 4 0.01 0
1 0 0.1 0 0.1 0
10 0 0.5 0 0.5 0
30 0 0.75 0 0.75 0
35 0 0.9 0 0.9 0
6 25 plr_conn PLR_CONN
ASCOS Connection element. Analysis of Smoke Control of Systems ...
8.38053e-008 0.000188562 0.5 0.0002 0.666667 3
7 23 plr_crack PLR_CRACK
Crack element using powerlaw relationship.
1.84e-008 0.000861959 0.68394 2 0.002 0 4
8 23 plr_fcn PLR_FCN
Power law for mass flow.
3.52946e-005 0.01 0.5
9 23 plr_leak1 PLR_LEAK1
Leakage element (per item).
2.4e-008 8.48528e-005 0.5 0.6 10 0.0001 0 0 2 2 2 0
10 23 plr_leak2 PLR_LEAK2
Leakage element (per unit length).
3.8063e-009 6.00712e-005 0.65 0.6 10 0 0.0001 0 2 2 2 0
11 23 plr_leak3 PLR_LEAK3
Leakage element (per unit area).
3.8063e-009 6.00712e-005 0.65 0.6 10 0 0 0.0001 2 2 2 1
12 23 plr_orfc PLR_ORFC
Orifice element.
2.4e-005 0.00848528 0.5 0.01 0.1 0.6 30 0 0
13 23 plr_qcn PLR_QCN
Power law for volume flow.
3.52946e-005 0.01 0.5
```

APPENDIX A – PRJ FILE FORMAT

```
14 23 plr_shaft PLR_SHAFT
Shaft powerlaw model.
  0.306565 30.9005 0.508447 3 6 10 0.1 0 0 0 0
15 23 plr_stair PLR_STAIR
Stairwell powerlaw model.
  0.146333 2.83196 0.5 3 12.5 0 1 0 0
16 23 plr_test1 PLR_TEST1
Single point test data element.
  2.47343e-005 0.0086575 0.5 4 0.019 0 0
17 23 plr_test2 PLR_TEST2
Two point test data element.
  6.35906e-005 0.00913228 0.461488 4 0.019 10 0.029 0 0 0 0
18 23 plr_bdf PL_BDF
PL_BDF Backdraft Damper based on mass flow rate
  3.52946e-005 0.01 0.5 0.0001 0.5
19 23 plr_bdq PL_BDQ
PL_BDQ Backdraft Damper based on volume flow rate
  3.52946e-005 0.01 0.5 0.0001 0.5
20 23 qfr_crack QF_CRACK
This description is the maximum allowable length that CONTAMW allo
  565.645 90835.5 2 0.002 0.05 2 0 4 2
21 23 qfr_fab QF_FAB
QF_FAB Quadratic mass flow model
  1234.12 123456
22 23 qfr_qab QF_QAB
QF_QAB Quadratic volume flow model
  1 1234.12
23 23 qfr_test2 QF_TEST2
QF_TEST2 Quadratic test data (2 points)
  -44.6461 13430.1 4 0.019 10 0.029 0 0 0 0
24 23 csf_fsp SR1_TN54_FvP
Mass flow version of JAW Self reg vent TN54 (+dP only)
  13 0 2
  0 0
  0.1 0.00437088
  0.25 0.00947627
  . . .
  4.5 0.024082
25 23 csf_qsp SR2_TN54_QvP
Volume flow version of JAW Self reg vent TN54 (+dP only)
  13 0 2
  0 0
  0.1 0.00437088
  0.25 0.00947627
  . . .
  4.5 0.024082
26 23 csf_psf SR3_TN54_PvF
Mass flow version of JAW Self reg vent TN54 (-dP only)
  6 2 0
  -0.0153861 -10
  -0.00951973 -8
  . . .
  -0.000533175 -1
  0 0
27 23 csf_psq SR4_TN54_PvQ
Volume flow version of JAW Self reg vent TN54 (-dP only)
  6 2 0
  -0.0153861 -10
  -0.00951973 -8
  . . .
  -0.000533175 -1
```

```
0 0
-999
```

Section 11: Duct Elements

Duct elements are read by the `afelmt_read()` function with the duct flag set to 1 and saved by the `afelmt_save()` function. The data are stored in the `AFE_DAT` structures that are defined in `contam.h` for ContamW and `simdat.h` for ContamX. `AFE_DAT` includes a pointer to the element specific data structure.

NOTE: Some of the elements in this section were incorrectly documented in the CONTAM 2.1 program documentation. However, the data structures have not changed for this version, only the documentation provided herein.

The duct elements section starts with:

```
_ndfe      // number of duct flow elements (IX)
```

This is followed by a data header comment line and then data for all `_ndfe` elements.

For each element the first data line includes:

```
nr          // element number (IX); in order from 1 to _nafe
icon        // icon used to represent flow path ((R4) {W})
dtype       // element data type (string ? I2)
            // element type names are stored in _afe_dnames in dtype order.
name[]      // element name (I1) {W}
```

and the second line has:

```
desc[]      // element description (I1) {W} may be blank
```

This is followed by two or more lines of data that depend on the element data type.

The introductory lines that follow give the type number, the program macro defined as that number, the type name from `_afe_dnames`, and the data structures to hold the data. The data structures are defined in `celmts.h` in ContamW and `selmts.h` in ContamX. Type numbers can change as long as the ordering in `_afe_dnames` and related arrays reflect that order.

Element type 23 [DD_DWC] "dct_dwc" stored in structure DEF_DWC.

The Darcy-Colebrook data consist of:

```
rough       // roughness dimension [m] (R4)
lam         // laminar total loss coefficient per [m] of duct (R4)
u_R         // units for roughness (I2) {W}
```

Element type 24 [DD_PLR] "dct_plr" stored in structure DFE_ORF.

The orifice data consist of:

```
lam         // laminar flow coefficient (R4)
turb        // turbulent flow coefficient (R4)
expt        // pressure exponent (R4)
area        // actual area [m^2] (R4) {X}
dia         // hydraulic diameter [m] (R4) {X}
coef        // flow coefficient (R4) {X}
u_A         // units of area (I2) {X}
u_D         // units of diameter (I2) {X}
```

Element type 26 [DD_QCN] "dct_qcn" stored in structure DFE_QCN.

The volume flow powerlaw data consist of:

```
lam      // laminar flow coefficient (R4)
turb     // turbulent flow coefficient (R4)
expt     // pressure exponent (R4)
```

Element type 25 [DD_FCN] "dct_fcn" stored in structure DFE_FCN.

The mass flow powerlaw data consist of:

```
lam      // laminar flow coefficient (R4)
turb     // turbulent flow coefficient (R4)
expt     // pressure exponent (R4)
```

Element type 28 [DD_CMF] "dct_cmf" stored in structure DFE_CMF.

The constant mass flow fan data consist of:

```
Flow     // design flow rate [kg/s] (R4)
u_F      // units of flow (I2) {W}
```

Element type 29 [DD_CVF] "dct_cvf" stored in structure DFE_CVF.

The constant volume flow fan data consist of:

```
Flow     // design flow rate [m^3/s] (R4)
u_F      // units of flow (I2) {W}
```

Element type 27 [DD_FAN] "dct_fan" stored in structure DFE_FAN.

The first line of performance curve fan data consists of:

```
lam      // laminar flow coefficient (R4)
turb     // turbulent flow coefficient (R4)
expt     // pressure exponent (R4)
rdens    // reference fluid density [kg/m^3] {R4}
fdf      // free delivery flow (prise = 0) [kg/s] {R4}
sop      // shut-off pressure (flow = 0) [Pa] {R4}
off      // fan is off if (RPM/rated RPM) < off {R4}
```

The second line consists of:

```
fpc[4]   // fan performance polynomial coefficients {R4}
npts     // number of mesasured data points (I2) {W}
Sarea    // shut-off orifice area [m^2] {R4} {W}
u_Sa     // units of shut-off area (I2) {W}
```

The next npts lines consists of:

```
mF       // measured flow rates [kg/s] (R4) {W}
u_mF     // units of measured flows (I2) {W}
dP       // measured pressure rises [Pa] (R4) {W}
u_dP     // units of pressure rises (I2) {W}
rP       // revised pressure rises [Pa] (R4) {W}
```

```
u_rP      // units of revised pressures (I2) {W}
```

Element type 30 [DD_BDQ] "dct_bdq" stored in structure DFE_BDQ.

The volume flow powerlaw backdraft damper data consist of:

```
lam      // laminar flow coefficient {R4}
Cp       // turbulent flow coefficient ( dP > 0 ) {R4}
xp       // pressure exponent ( dP > 0 ) {R4}
Cn       // turbulent flow coefficient ( dP < 0 ) {R4}
xn       // pressure exponent ( dP < 0 ) {R4}
```

Element type 31 [DD_BDF] "dct_bdf" stored in structure DFE_BDF.

The mass flow powerlaw backdraft damper data consist of:

```
lam      // laminar flow coefficient {R4}
Cp       // turbulent flow coefficient ( dP > 0 ) {R4}
xp       // pressure exponent ( dP > 0 ) {R4}
Cn       // turbulent flow coefficient ( dP < 0 ) {R4}
xn       // pressure exponent ( dP < 0 ) {R4}
```

After each element comes the following geometry and leakage data stored in structure DUCT.

The first line of duct geometry data consists of:

```
hdia     // hydraulic diameter [m] {R4}
perim    // perimeter [m] {R4}
area     // cross sectional area [m^2] {R4}
major    // major dimension of rectangular or oval duct [m] {R4}
minor    // minor dimension of rectangular or oval duct [m] {R4}
As       // duct segment surface area [m^2] {R4}
Qr       // duct leakage rate at Pr [m^3/m^2] {R4}
Pr       // dPstatic for leakage rate [Pa] {R4}
```

The second line of duct geometry data consists of:

```
shape    // 0 = circle, 1 = rectangle, 2 = oval, 3 = other (I2) {W}
u_D      // units for diameter (I2) {W}
u_P      // units for perimeter (I2) {W}
u_A      // units for area (I2) {W}
u_mj     // units for major dimension (I2) {W}
u_mn     // units for minor dimension (I2) {W}
u_Qr     // units for leakage rate (I2) {W}
u_Pr     // units for standard dP (I2) {W}
```

The duct element section is terminated with:

```
-999     // used to check for a read error in the above data
```

Example:

```
9 ! duct elements:
1 23 dct_dwc Dct1
Duct one.
9e-005 0 4
0.2 0.628319 0.0314159 0.2 0.2 0 0 250
0 4 4 2 4 4 0 0
2 23 dct_plr Dct2
Duct two
2.4e-005 0.00848528 0.5 0.01 0.1 0.6 0 0
```

```

0.2 0.628319 0.0314159 0.2 0.2 0 0 250
0 4 4 2 4 4 0 0
3 23 dct_fcn Dct3
Duct three.
3.52946e-005 0.01 0.5
0.2 0.628319 0.0314159 0.2 0.2 0 0 250
0 4 4 2 4 4 0 0
4 23 dct_qcn Dct4
Duct four
3.52946e-005 0.01 0.5
0.2 0.628319 0.0314159 0.2 0.2 0 0 250
0 4 4 2 4 4 0 0
5 23 dct_fan Dct5
Duct five
7.2e-006 0.00848528 0.5 1.2041 4.53711 764.429 0.1
764.429 -22.0238 28.2143 -13.3333 5 0.01 0
0 0 765 0 765 0
1 0 755 0 755 0
2 0 730 0 730 0
3 0 590 0 590 0
4 0 275 0 275 0
0.2 0.628319 0.0314159 0.2 0.2 0 0 250
0 4 4 2 4 4 0 0
6 23 dct_cmf Dct6
Duct 6
1 0
0.2 0.628319 0.0314159 0.2 0.2 0 0 250
0 4 4 2 4 4 0 0
7 23 dct_cvf Dct7
Duct seven
1 0
0.2 0.628319 0.0314159 0.2 0.2 0 0 250
0 4 4 2 4 4 0 0
8 23 dct_bdq Dct8
Duct eight
3.52946e-005 0.01 0.5 0.0001 0.5
0.2 0.628319 0.0314159 0.2 0.2 0 0 250
0 4 4 2 4 4 0 0
9 23 dct_bdf Dct9
Duct nine
3.52946e-005 0.01 0.5 0.0001 0.5
0.2 0.628319 0.0314159 0.2 0.2 0 0 250
0 4 4 2 4 4 0 0
-999

```

Section 12a: Control Super Elements

Control Super Element data are read by the `ctrlse_read()` function and saved by the `ctrlse_save()` function in ContamW. The data are stored in the `CTRLSE_DAT` structures that are defined in `contam.h` for ContamW. Super Element data are not read in by ContamX, only the nodes which are instantiated from Control Super Elements are read in by ContamX as described in the following Section 12b.

The Super Elements section starts with:

```
_nse1mt // number of super elements (IX)
```

This is followed by data for all `_nse1mt` super elements.

For each super element the first data line includes:

```
nr                // element number (IX); in order from 1 to _nselemt
flags            // if( flags & CF_DEF ) super element is defined (U2)
                // if( flags & CF_SNS ) super element is a sensor
in              // input node number (IX)
out            // output node number (IX)
nn             // number of sub-nodes (IX)
ni            // number of sketch icons (I2)
name[]         // element name (I1)
```

and the second line has:

```
desc[]          // element description (I1) may be blank
```

This is followed by a section that contains sub-node data of the Super Element. This will be very similar to the "control nodes:" section of the project file (see Section 12b below) and repeated within section 12b for each super node that is instantiated from a super element. However, the actual parameters of the instantiated super nodes may vary from those of the super element sub-nodes.

This is followed by lines of data that contain the SketchPad icon parameters of the Super Element sketch and will follow the "levels plus icon data:" section of the project file but only for a single "level" of data that represents the Super Element sub-node icons (see Section 3).

The Super Elements section is terminated with:

```
-999
```

Example:

```
2 ! control super elements:
! #  f  in out  nn  ni name
  1  0  0  0  0  0 SE_1

0 ! sub-nodes:
  2  1  1  2  3  8 SE_2
Add 100 to Input
3 ! sub-nodes:
! # typ seq f n  c1  c2 name
  1 add  1 0 2  0  3 InputNode
Add two inputs

  2 pas  3 0 1  1  0 OutputNode
Split (or pass) input

  3 set  2 0 0  0  0 <none>
K = 100
100
189 19  8  1
170 20  8  1
178 25  8  1
188 26  8  2
177 19  9  3
185 15 11  3
170 16 11  3
174 19 11  3
-999
```

Section 12b: Control Nodes

Control node data are read by the `ctrl_read()` function and saved by the `ctrl_save()` function. The data are stored in the `CTRL_DSC` structures that are defined in `contam.h` for ContamW and the `CT_NODE` structures defined `simdat.h` for ContamX.

The control nodes section starts with:

```
_nctrl // number of control nodes (IX); in order from 1 to _nctrl
```

This is followed by a data header comment line and then data for all _nctrl control nodes.

For each control node the first data line includes:

```
nr // node (SketchPad) number (IX); in order from 1 to _nctrl
type // node data type (string → I2)
      Node type names are stored in _ctrl_names in type order.
seqnr // computation sequence number (IX); set in ContamW
flags // flags for offset & scale, time constant, and 1D sensor (U2)
inreq // number of required inputs (I2) {W}
n1 // SketchPad number of input node #1 (IX)
n2 // SketchPad number of input node #2 (IX)
name[] // element name (I1) {W}
```

and the second line has:

```
desc[] // control node description (I1) {W} may be blank
```

This may be followed by one more line of data that depends on the node type.

Node type 0 [CT_SNS] "sns" stored in structure SENSOR {W} or SNSDAT {X}.

```
offset // offset value (R4)
scale // scale value (R4)
tau // time constant (R4)
oldsig // signal at last time step - for restart (R4)
source // index of source (source not defined at read time) (IX)
type // type of source: 1=zone, 2=path, 3=junction, 4=duct... (I2)
measure // 0=contaminant, 1=temperature, 2=flow rate, 3=dP ... (I2)
X // X-coordinate of sensor [m] (R4)
Y // Y-coordinate of sensor [m] (R4)
relHt // relative height of sensor [m] (R4)
units[] // units of coordinates {W} (I1)
species[] // species name [I1]; convert to pointer
```

Node type 1 [CT_SCH] "sch" stored in structure SCHDAT.

```
ps // week schedule index (I2); converted to pointer
```

Node type 2 [CT_SET] "set" stored in structure SETDAT.

```
value // constant value (R4)
```

Node type 3 [CT_CVFF] "cvf" stored in structure CDVDAT. {Contam 2.1}

```
name[] // name of the value read from the Continuous Values file (I1)
```

Node type 4 [CT_DVFF] "dvf" stored in structure CDVDAT. {Contam 2.1}

```
name[] // name of the value read from the Discrete Values file (I1)
```

Node type 5 [CT_LOG] "log" stored in structure LOGDAT.

```
offset // offset value (R4)
scale // scale value (R4)
undef // true if using default units (I2) {W}
header[] // header string (I1)
units[] // units string (I1)
```

Node type 6 [CT_PAS] "pas" has no additional data.

Node type 7 [CT_MOD] "mod" stored in structure MOD.

```
offset    // offset value (R4)
scale     // scale value (R4)
```

Node type 8 [CT_HYS] "hys" stored in structure HYSDAT.

```
slack;    // hysteresis parameter (R4)
slope     // 1.0 / (1.0 - slack) (R4)
oldsig    // prior output signal (R4)
```

Node type 9 [CT_ABS] "abs" has no additional data.

Node type 10 [CT_BIN] "bin" has no additional data.

Node type 11 [CT_DLS] "dls" stored in structure CDVDAT. {Contam 2.1}

```
dsincr    // day schedule number for increasing signal (I2)
dsdecr    // day schedule number for decreasing signal (I2)
```

Node type 12 [CT_DLX] "dlx" stored in structure CDVDAT. {Contam 2.1}

```
tauincr   // time constant for increasing signal [s] (I4)
taudecr   // time constant for decreasing signal [s] (I4)
```

Node type 13 [CT_INT] "int" has no additional data.

Node type 14 [CT_RAV] "rav" stored in structure CDVDAT. {Contam 2.1}

```
tspan     // time span for the running average [s] (I4)
```

Node type 15 [CT_INV] "inv" has no additional data.

Node type 16 [CT_AND] "and" has no additional data.

Node type 17 [CT_OR] "od" has no additional data.

Node type 18 [CT_XOR] "xor" has no additional data.

Node type 19 [CT_ADD] "add" has no additional data.

Node type 20 [CT_SUB] "sub" has no additional data.

Node type 21 [CT_MUL] "mul" has no additional data.

Node type 22 [CT_DIV] "div" has no additional data.

Node type 23 [CT_SUM] "sum" stored in structure SUMAVG.

Node type 24 [CT_AVG] "avg" stored in structure SUMAVG.

Node type 25 [CT_MAX] "max" stored in structure SUMAVG.

Node type 26 [CT_MIN] "min" stored in structure SUMAVG.

```
npcs      // number of controls to be processed (IX)
pc[npcs]  // indices of npcs control nodes (IX)
```

Node type 27 [CT_LLS] "lls" has no additional data.

Node type 28 [CT_ULS] "uls" has no additional data.

Node type 29 [CT_LBS] "lbs" stored in structure BANDAT.

Node type 30 [CT_UBS] "ubs" stored in structure BANDAT.

```
band      // width of band (R4)
```

Node type 31 [CT_LLC] "llc" has no additional data.

Node type 32 [CT_ULC] "ulc" has no additional data.

Node type 33 [CT_PCI] "pci" stored in structure PCDAT.

```
kp // proportional gain factor (R4)
```

Node type 34 [CT_PII] "pii" stored in structure PICDAT.

```
kp // proportional gain factor (R4)
ki // integral gain factor (R4)
oldsig // prior output signal - for restart (R4)
olderr // prior error value - for restart (R4)
```

Node type 35 [CT_SUP] "sup" stored in structure SUPDAT.

```
def // 0:super element undefined 1:defined (I2)
se // super element number (IX)
in // input sub-node number (IX)
out // output sub-node number (IX)
```

Node type 36 [CT_SEN] this is only a placeholder value for use with the type dialog.

Node type 37 [CT_SPH] "sph" has no additional data.

The control nodes section is terminated with:

```
-999 // used to check for a read error in the above data
```

Example:

```
9 ! control nodes:
! # typ seq f n c1 c2 name
1 sns 2 0 0 0 0 <none>
zone sensor
0 1 0 0 1 1 0 0.000 0.000 0.000 0 C2
2 log 5 0 1 1 0 Z5C2
report
0 1 1 MassFraction kg/kg
3 set 0 0 0 0 0 <none>
K = 1
1
4 sup 0 0 1 3 0 SN_2
Super node based on SE_2
0 2 5 6
5 add 1 0 2 3 7 <none>
Add two inputs

6 pas 3 0 1 5 0 SN_2
Split (or pass) input

7 set 2 0 0 0 0 <none>
K = 100
100
8 sns 3 0 0 0 0 <none>
path sensor
0 1 0 0 3 2 2 0.000 0.000 0.000 0 none
9 log 7 0 1 8 0 Path-L
report
0 1 1 FlowRate kg/s
-999
```

Section 13: Simple Air Handling System (AHS)

AHS data are read by the `system_read()` function and saved by the `system_save()` function. The data are stored in the `AHS_DSC` structures that are defined in `contam.h` for `ContamW` and `simdat.h` for `ContamX`.

The AHS section starts with:

```
_nahs      // number of AHS (IX)
```

This is followed by a data header comment line and then data for all `_nahs` systems.

For each AHS the first data line includes:

```
nr          // AHS number (IX); in order from 1 to _nahs
zone_r      // return zone number (IX)
zone_s      // supply zone number (IX)
path_r      // recirculation air path number (IX)
path_s      // outdoor air path number (IX)
path_x      // exhaust path number (IX)
```

and the second line has:

```
desc[]      // AHS description (I1) {W} may be blank
```

The AHS section is terminated with:

```
-999        // used to check for a read error in the above data
```

Example:

```
1 ! simple AHS:
! # zr# zs# pr# ps# px# name
  1  2  3 29 30 31 Ahs1
Simple Air Handling System #1
-999
```

Section 14: Zones

Zone data are read by the `zone_read()` function and saved by the `zone_save()` function. The data are stored in the `ZONE_DSC` structures that are defined in `contam.h` for `ContamW` and `simdat.h` for `ContamX`.

The zone section starts with:

```
_nzone      // number of zones (IX)
```

This is followed by a data header comment line and then data for all `_nzone` zones.

For each zone the data line includes:

```
nr          // zone number (IX); in order from 1 to _nzone
flags       // zone flags - bits defined in contam.h (U2)
ps          // week schedule index (IX); converted to pointer
pc          // control node index (IX); converted to pointer
pk          // kinetic reaction index (IX); converted to pointer
pl          // building level index (IX); converted to pointer
relHt      // zone height [m] (R4)
Vol         // zone volume [m^3] (R4)
T0          // initial zone temperature [K] (R4)
P0          // initial zone pressure [Pa] (R4)
name[]      // zone name (I1) {W}
color       // zone fill color (I2) {W} {Contam 2.4}
u_Ht        // units of height (I2) {W}
u_V         // units of volume (I2) {W}
u_T         // units of temperature (I2) {W}
u_P         // units of pressure (I2) {W}
```

```

cdaxis // conv/diff axis - 0:no, >0:yes (I2)
If cdaxis > 0 then the line continues to include:
"1D:" // string indicating 1D data follows
X1 // X coordinate of one end of cdaxis
Y1 // Y " "
H1 // Relative Height "
X2 // X coordinate of other end of cdaxis
Y2 // Y " "
H2 // Relative Height "
celldx // length of c/d cell [m] (R4)
axialD // axial diffusion coeff [m^2/s] (R4)
u_aD // units of axial diffusion (I2)
u_L // units of c/d axis limits (I2)

```

The zones section is terminated with:

```
-999 // used to check for a read error in the above data
```

Example:

```

7 ! zones:
! Z# f s# c# k# l# relHt Vol T0 P0 name clr u.. axz 1-D data:
  1 3 0 0 0 2 0.000 5 296.15 0 z5 -1 0 2 0 0 1 1D: 0 10 0 0 0 0 0.1 0.1 0 0
  2 10 0 0 0 2 0.000 2 296.15 0 ahsR -1 0 2 0 0 0
  3 10 0 0 1 2 0.000 1 296.15 0 ahsS -1 0 2 0 0 0
  4 3 0 0 0 1 0.000 1 296.15 0 z1 -1 0 2 0 0 0
  5 3 0 0 0 1 0.000 2 293.15 0 z2 -1 0 2 0 0 0
  6 3 0 0 0 1 0.000 3 296.15 0 z3 -1 0 2 0 0 0
  7 3 0 0 0 1 0.000 4 296.15 0 z4 -1 0 2 0 0 0
-999

```

Section 15: Initial Zone Concentrations

Initial zone contaminant mass fractions are read by the zone_read() function and saved by the zone_save() function. The mass fractions are stored in the ZONE_DSC structures that are defined in contam.h for ContamW and they are stored in an array in ContamX.

The zone contaminants section starts with:

```

nn // number of mass fractions that follow (IX)
nn should equal _nzone * _nctm.

```

This is followed by _nzone lines in order from 1 to _nzone.

Each data line contains _nctm mass fractions (R4) in contaminant order.

```

nr // zone number (IX)
CC0[i][1] // initial mass fraction (R4) of zone i, contaminant 1
...
CC0[i][_nctm] // initial mass fraction (R4) of zone i, contaminant _nctm

```

The initial zone concentrations section is terminated with:

```
-999 // used to check for a read error in the above data
```

Example:

```

14 ! initial concentrations:
! Z# C1 C2
  1 1.000e+000 2.000e+000
  2 1.000e+000 2.000e+000
  3 1.000e+000 2.000e+000

```

```

4 1.000e+000 2.000e+000
5 1.000e+000 2.000e+000
6 1.000e+000 2.000e+000
7 1.000e+000 2.000e+000

```

-999

Section 16: Airflow Paths

Path data are read by the `path_read()` function and saved by the `path_save()` function. The data are stored in the `PATH_DSC` structures that are defined in `contam.h` for `ContamW` and `simdat.h` for `ContamX`.

The path section starts with:

```
_npath // number of paths (IX)
```

This is followed by a data header comment line and then data for all `_npath` paths.

For each path the data line includes:

```

nr // path number (IX); in order from 1 to _npath
flags // airflow path flag values (I2)
pzn // zone N index (IX); converted to pointer
pzm // zone M index (IX); converted to pointer
pe // flow element index (IX); converted to pointer
pf // filter index (IX); converted to pointer
pw // wind coefficients index (IX); converted to pointer
pa // AHS index (IX); converted to pointer
ps // schedule index (IX); converted to pointer
pc // control node index (IX); converted to pointer
pld // level index (IX); converted to pointer
X // X-coordinate of envelope path [m] (R4) {Contam 2.1}
Y // Y-coordinate of envelope path [m] (R4) {Contam 2.1}
relHt // height relative to current level [m] (R4)
mult // element multiplier (R4)
wPset // constant wind pressure [Pa] (pw==NULL) (R4)
wPmod // wind speed(?) modifier (pw!=NULL) (R4)
wazm // wall azimuth angle (pw!=NULL) (R4)
Fahs // AHS path flow rate [kg/s] (pw==NULL) (R4)
Xmax // flow or pressure limit - maximum (R4) {W}
Xmin // flow or pressure limit - minimum (R4) {W}
icon // icon used to represent flow path (U1) {W}
dir // positive flow direction on sketchpad (U1) {W}
u_Ht // units of height (I2) {W}
u_XY // units of X and Y (I2) {W}
u_dP // units of pressure difference (I2) {W}
u_F // units of flow (I2) {W}

```

The paths section is terminated with:

```
-999 // used to check for a read error in the above data
```

Example:

```

31 ! flow paths:
! P# f n# m# e# f# w# a# s# c# l# X Y relHt mult wPset wPmod wazm Fahs..
1 8 1 2 0 0 0 1 0 0 2 0.000 0.000 0.000 1 0 0 0 2 0 0 129 5 0 0 0 0
2 8 3 1 0 0 0 1 0 0 2 0.000 0.000 0.000 1 0 0 0 1 0 0 128 2 0 0 0 0

```

```

3 7 -1 1 12 0 1 0 0 0 2 0.000 0.000 1.500 1 0 0.36 180 0 0 0 30 1 0 0 0 0
4 7 -1 1 12 0 2 0 0 0 2 0.000 0.000 1.500 2 0 0.36 180 0 0 0 23 1 0 0 0 0
5 7 -1 4 13 0 2 0 0 0 1 19.000 17.000 1.500 1 0 0.36 0 0 0 0 23 4 0 0 0 0
6 6 -1 4 8 0 0 0 0 0 1 21.000 17.000 1.500 1 0 0 0 -1 0 0 23 4 0 0 0 0
. . .
30 38 -1 3 0 1 0 0 0 0 2 17.000 5.000 1.500 1 0 0 0 -1 0 0 0 6 0 0 0 0
31 64 2 -1 0 0 0 0 0 0 2 0.000 0.000 1.500 1 0 0 0 -1 0 0 0 6 0 0 0 0
-999

```

Section 17: Duct Junctions

Junction data are read by the `jct_read()` function and saved by the `jct_save()` function. The data are stored in the `JCT_DSC` structures that are defined in `contam.h` for `ContamW` and `simdat.h` for `ContamX`.

The junction section starts with:

```
_njct // number of junctions (IX)
```

This is followed by a data header comment line and then data for all `_njct` junctions.

For each junction the data line includes:

```

nr // junction number (IX); in order from 1 to _njct
flags // junction flags - bits defined in contam.h (U2)
jtype // junction type flag: 0=jct, 1=term, 3=term w/ wind {Contam 2.1}
pzn // surrounding zone index (IX); converted to pointer
dnr // duct segment number - use during read (IX)
pk // kinetic reaction index (IX); converted to pointer
ps // schedule index (IX); converted to pointer
pc // control node index (IX); converted to pointer
pld // level index (IX); converted to pointer
X // X-coordinate of ambient terminal [m] (R4) {Contam 2.1}
Y // Y-coordinate of ambient terminal [m] (R4) {Contam 2.1}
relHt // height relative to current level [m] (R4)
T0 // initial junction temperature [K] (R4)
P0 // initial junction pressure [Pa] (R4)
icon // icon used to represent junction (U1) {W}
color // icon color (I2) {W} {Contam 2.4}
u_Ht // units of height (I2) {W}
u_XY // units of X and Y (I2) {W}
u_T // units of temperature (I2) {W}
u_dP // units of pressure difference (I2) {W}

```

If `jtype > 0` then the line continues with:

```

"T:" // string indicates terminal data follows {Contam 2.3}
pf // filter index (IX); converted to pointer {Contam 2.3}
pw // wind coefficients index (IX); converted to pointer
wPset // constant wind pressure [Pa] (pw==NULL) (R4)
wPmod // wind speed(?) modifier (pw!=NULL) (R4)
wazm // wall azimuth angle (pw!=NULL) (R4)
bal // balance terminal flag (I2) {Contam 2.3}
Ad // area of duct connected to terminal [m^2] (R4) {Contam 2.3}
Af // free area of terminal [m^2] (R4) {Contam 2.3}
Fdes // design flow rate [kg/s] (R4) {Contam 2.3}
Ct // terminal loss coefficient (R4) {Contam 2.3}

```

```

Cb          // balance loss coefficient (R4) {Contam 2.3}
Cbmax      // maximum Cb desired (R4) {Contam 2.3}
u_A        // units of area (I2) {Contam 2.3}
u_F        // units of mass flow (I2) {Contam 2.3}
ddir       // direction of terminal duct - to show wind pressure (U1) {W}
fdir       // positive flow direction - to show flow (U1) {W}

```

The junctions section is terminated with:

```
-999      // used to check for a read error in the above data
```

Example:

```

23 ! duct junctions:
! J#  f t  z#  dct  k#  s#  c#  l#  X      Y      relHt  T0  P0  icon  clr  u..  Terminal:
  1  0 3  -1  0  0  0  0  2  0.000  0.000  0.000 293.15 0 162  -1 0 0 2 0 T:
    <continue previous line> 0 0 0 0 -1 0 0.06 0.06 0 1 0 0.8 0 0 2 5
  2  0 0  -1  0  0  0  0  2  0.000  0.000  0.000 293.15 0 158  -1 0 0 2 0
  3  0 3  -1  0  0  0  0  2  0.000  0.000  0.000 293.15 0 162  -1 0 0 2 0 T:
    <continue previous line> 0 0 0 0 -1 1 0.03 0.015 0.2 1 0 0.8 0 0 5 2
. . .
23  0 1  1  0  0  0  0  2  0.000  0.000  0.000 293.15 0 162  -1 0 0 2 0 T:
    <continue previous line> 0 0 0 0 -1 0 0.15708 0.15708 0 0.125 0 0.8 0 0 1 1
-999

```

Section 18: Initial Junction Concentrations

Initial junction contaminant mass fractions are read by the `zone_read()` function and saved by the `zone_save()` function. The mass fractions are stored in the `ZONE_DSC` structures that are defined in `contam.h` for `ContamW` and they are stored in an array in `ContamX`.

The initial junction contaminants section starts with:

```

mn          // number of mass fractions that follow (IX)
nn should equal _njct * _nctm. This is followed by _njct lines in order from junction 1 to
junction _njct.

```

Each data line contains _nctm mass fractions (R4) in contaminant order.

The initial junction concentrations section is terminated with:

```
-999      // used to check for a read error in the above data
```

Example:

```

46 ! initial junction concentrations:
! J#      C1      C2
  1 5.318e-004 6.838e-004
  2 5.318e-004 6.838e-004
. . .
22 1.000e+000 2.000e+000
23 1.000e+000 2.000e+000
-999

```

Section 19: Duct Segments

Duct data are read by the `duct_read()` function and saved by the `duct_save()` function. The data are stored in the `DCT_DSC` structures that are defined in `contam.h` for `ContamW` and `simdat.h` for `ContamX`.

The duct segment section starts with:

```
_ndct      // number of ducts (IX)
```

This is followed by a data header comment line and then data for all `_ndct` ducts.

For each duct the data line includes:

```
nr          // duct number (IX); in order from 1 to _ndct
flags       // duct flag values (I2)
pjn        // junction n index (IX); converted to pointer
pjm        // junction m index (IX); converted to pointer
pe         // duct flow element index (IX); converted to pointer
pf         // filter index (IX); converted to pointer
ps         // schedule index (IX); converted to pointer
pc         // control node index (IX); converted to pointer
dir        // positive flow direction on sketchpad (U1) {W}
length     // length of the duct segment [m] (R4)
Ain        // flow area at inlet end [m^2] - future (R4)
Aout       // flow area at outlet end [m^2] - future (R4)
sllc       // sum of local loss coefficients (R4)
color      // icon color (I2) {W} {Contam 2.4}
u_L        // units for length (I2) {W}
u_A        // units for flow area (I2) {W}
```

The duct segment section is terminated with:

```
-999
```

Example:

```
! D#  f  n#  m#  e#  f#  s#  c#  dir  len  Ain  Aout  sllc  clr  u..
   1  0  1  2  1  0  0  0  2  5  0.15708  0.15708  0  -1  0  0
   2  0  2  3  2  0  0  0  2  5  0.15708  0.15708  0  -1  0  0
   3  0  2  4  2  0  0  0  2  5  0.15708  0.15708  0  -1  0  0
   .  .  .
  20  0  22  13  3  0  0  0  1  5  0.15708  0.15708  0  -1  0  0
  21  0  23  14  2  0  0  0  1  5  0.15708  0.15708  0  -1  0  0
-999
```

Section 20: Source/Sinks

Contaminant sources/sinks are read by the `css_read()` function and saved by the `css_save()` function. The data are stored in the `CSS_DSC` structures that are defined in `contam.h` for `ContamW` and `simdat.h` for `ContamX`.

The source/sink section starts with:

```
_ncss      // number of source/sinks (IX)
```

This is followed by a data header comment line and then data for all `_ncss` source/sinks.

For each source/sink the data line includes:

```
nr          // source/sink number (IX); in order from 1 to _ncss
pz         // zone index (IX); converted to pointer
pe         // source/sink element index (IX); converted to pointer
ps         // schedule index (IX); converted to pointer
pc         // control node index (IX); converted to pointer
mult       // multiplier (R4)
CC0        // initial mass fraction (some types) [kg/kg] (R4)
Xmin       // minimum x coordinate of source/sink location [m] (R4)
Ymin       // minimum y coordinate of source/sink location [m] (R4)
Hmin       // minimum z coordinate relative to level [m] (R4)
```

```

Xmax      // maximum x coordinate of source/sink location [m] (R4)
Ymax      // maximum y coordinate of source/sink location [m] (R4)
Hmax      // maximum z coordinate relative to level [m] (R4)
u_XYZ     // units of location coordinates (I2)

```

The source/sink section is terminated with:

```
-999      // used to check for a read error in the above data
```

Example:

```

! #  z#  e#  s#  c#  mult  CC0  (X, Y, H)min  (X, Y, H)max  u
1  1  1  0  0  1  0  0.05 0 0  0.05 0 0 0
2  1  2  0  0  1  0  1.05 0 0  1.05 0 0 0
3  1  3  0  0  1  0  2.05 0 0  2.05 0 0 0
4  1  4  0  0  1  0  3.05 0 0  3.05 0 0 0
5  1  5  0  0  1  0  4.05 0 0  4.05 0 0 0
6  1  6  1  0  1  0  5.05 0 0  5.05 0 0 0
7  1  7  0  0  1  0  6.05 0 0  6.05 0 0 0
8  1  8  0  0  1  0  7.05 0 0  7.05 0 0 0
-999

```

Section 21: Occupancy Schedules

Occupancy schedules are read by the `oschd_read()` function and saved by the `oschd_save()` function. The data are stored in the `OD_SCHD` structures that are defined in `contam.h` for `ContamW` and `simdat.h` for `ContamX`.

The occupancy schedule section starts with:

```
_nosch    // number of schedules (IX)
```

This is followed by a data header comment line and then data for all `_nosch` schedules.

For each schedule the first data line includes:

```

nr        // schedule number (IX); in order from 1 to _nosch
npts      // number of points (I2)
u_XYZ     // units of location coordinates (I2)
name[]    // schedule name (I1) {W}

```

and the second line has:

```
desc[]    // schedule description (I1) {W} may be blank
```

This is followed by `npts` lines of schedule data:

```

time      // time-of-day [s] (hh:mm:ss converted to I4)
pz        // zone index (I2); converted to pointer
X         // X-coordinate of occupant [m] (R4)
Y         // Y-coordinate of occupant [m] (R4)
relHt     // height relative to current level [m] (R4)

```

The occupancy schedule section is terminated with:

```
-999      // used to check for a read error in the above data
```

Example:

```

1 ! occupancy schedules:
1 3 0 OccDySched1
Occupant day schedule one.
00:00:00 0 0.050 0.000 1.500
00:12:00 4 0.000 0.000 0.000
24:00:00 4 0.000 0.000 0.000
-999

```

Section 22: Exposures

Exposure data are read by the `pexp_read()` function and saved by the `pexp_save()` function. The data are stored in the `PEXP_DSC` structures that are defined in `contam.h` for ContamW and `simdat.h` for ContamX.

The exposure section starts with:

```
_npexp // number of exposures (IX)
```

This is followed by three lines of data for all `_npexp` exposures.

For each exposure the first data line includes:

```
nr // exposure number (IX); in order from 1 to _npexp
gen // = 1 if contaminants are generated (I2)
ncg // number of contaminant generations (I2)
cgmlt // contaminant generation multiplier [-] (R4)
inhmax // peak inhalation rate [m^3/s] (R4) {unused}
inhsch // weekly inhalation schedule index (I2)
bodywt // body weight [kg] (R4) {unused}
u_inh // units for inhalation rate (I2)
u_bw // units for body weight (I2)
```

The second line has the exposure description:

```
desc[] // exposure/person description (I1)
```

The third line has the indices of 12 occupancy schedules:

```
odsch[12] // vector of daily occupancy schedules - 12 indices
```

It is followed by `ncg` lines of contaminant generation data:

```
name // species name (I1)
ps // schedule index (I2); converted to pointer
cgmax // peak generation rate [kg/s] (R4)
u_cg // units of generation rate (I2) {W}
```

The exposure section is terminated with:

```
-999 // used to check for a read error in the above data
```

Example:

```
2 ! exposures:
1 1 2 1 0.000334444 1 70 4 0
Occupant one.
1 1 1 1 1 1 1 1 1 1 1 1 ! occ. schd
C1 1 1 0 ! occ. gen
C2 1 2 0 ! occ. gen
2 0 2 1 0.000668889 1 80 4 0
Occupant two.
1 1 1 1 1 1 1 1 1 1 1 1 ! occ. schd
C1 0 1 0 ! occ. gen
C2 1 2 0 ! occ. gen
-999
```

Section 23: Annotations

Annotations are read by the `note_read()` function and saved by the `note_save()` function. The data are stored in the `NOTE_DSC` structures that are defined in `contam.h`. Annotations are used only in ContamW; nothing is stored when this section is read in ContamX.

The annotations section starts with:

```
_nnode      // number of annotations (IX)
```

This is followed by data for all _nnode annotations:

For each annotation data line includes:

```
nr          // annotation number (IX) {W}; in order from 1 to _nnode  
note[]      // annotation (I1) {W}
```

The annotations section is terminated with:

```
-999        // used to check for a read error in the above data
```

Example:

```
6 ! annotations:  
1 PLR_type Airflow Elements  
2 to force flows through PLR paths  
.  
.  
.  
6 FN_type Airflow Elements  
-999
```

Appendix B ContamX Program Documentation

B.1 Introduction

This documentation is intended as an aid to understanding the ContamX 2.4 computer code and is meant primarily to address the logical structure of the program.

The overall logical structure of the program is presented below in [Overall Program Structure](#). This material is not necessary for the general user of the program, but is provided for those interested in understanding or perhaps enhancing its capabilities. You can use the hyperlinks to jump to detailed descriptions of the highlighted functions. After executing a link you may return to the spot where the link was called by clicking on the left arrow in the toolbar above the document. This process is designed to facilitate both the rapid review of program structure and quick access to the details of the calculations. If working from a hard copy simply keep in mind that functions are presented alphabetically. There are also many small functions performing simple operations that are called from many places in the program – see [Utility Functions](#). A special method is used to store the data for the solution of the simultaneous equations – see [Sparse Matrix Data Structure](#).

B.2 Development Environment

CONTAMX Version 2.1 was developed using Microsoft Visual C++ versions 6.0 and .NET. It is basically a console application written in the C programming language. However, the program does require Microsoft Windows in order to display simulation progress in a dialog box. Other than that, it could be compiled in a non-Windows environment with minor modifications to the source code.

B.3 ContamX Program Structure

This section presents the program structure of the solver, ContamX. Section 2.3.1 presents the overall program structure, 2.3.2 presents the main solver functions in alphabetical order, 2.3.3 presents the utility functions used by the solver, 2.3.4 presents the sparse matrix data structure and 2.3.5 presents the solution to the species differential equations.

B.3.1 Overall Program Structure

Program initialization [mostly from main() in contamx.c]

Determine path to CONTAMX2.EXE

Open CONTAMX2.LOG file for informative dumps

Determine path to project file – command line or interactive input

Determine if dialog box used

if so, open it and start ContamX thread,

otherwise, continue ContamX in DOS window

Determine if ACATS Bridge used

if so, connect to bridge – in bridge.c

Set output mode (dialog box or DOS window) for error messages [from contamx()]

Data initialization [from contamx() in contamx.c]

Read project file – see [prj_read\(\)](#)

Complete ContamX data structures – see [sim_data\(\)](#)

Create simulation arrays [from contamx(); see [mat_init\(\)](#)

Set up airflow matrices – see [af_mat_set\(\)](#)

Set up non-trace mass matrices – see [mf_mat_set\(\)](#)

Set up trace mass matrices – see [mf_mat_set\(\)](#)

Report equation numbers to LOG file

Initialize simulation [from contamx() in contamx.c]

Set start/end dates and times

Open/initialize weather file – see [weather_init\(\)](#)

Open/initialize contaminant file – see [ambtctm_init\(\)](#)

Initialize schedules and control values – see [ctrl_sim\(\)](#)

Open simulation output files

Perform duct balance [from contamx() in contamx.c] -- see [Balance\(\)](#)

or

Perform simulation [from contamx() in contamx.c]

Initialize flow and mass fraction values

from restart file – [resget\(\)](#)

or by steady-state calculation – see [sim_init\(\)](#)

If using STS method, set up STS structures – see [STSinit\(\)](#)

Perform transient or cyclic calculation – see [sim_loop\(\)](#)

Normal program termination [from contamx() in contamx.c]

Report simulation performance to LOG file

De-allocate all earlier heap allocations – see [mat_term\(\)](#)

Perform memory checks

Close LOG file

Return to main(); return to operating system

B.3.2 ContamX Solver Functions

□ **af_mat_alc()**

[function in file matset.c]

Allocate remaining airflow solution arrays.

Determine number of non-zero coefficients in symmetric matrix – in `ija_minset()`

Allocate and set the `ija` index vector – in `ija_minset()`

Allocate the `sa`, `b`, and `x` coefficients vectors

For symmetric skyline solution method allocate remaining arrays – in `sky_alc()`

For conjugate gradient method allocate `p`, `pp`, `r`, `rr`, `z`, `zz`, and preconditioning vectors

For Gauss elimination allocate `aa` full matrix

□ **af_mat_set()**

[function in file matset.c]

Set up airflow matrices.

Transfer control parameters from `_rcdat` to the structure for airflow calculations

Set up vector of pointers to nodes with constant pressure nodes last

Determine positions of all non-zero elements in flow solution matrix – in `acolset()`

Allocate and fill the `ijf` vector sparse data mapping – see [Sparse Matrix Data Structure](#)

For the skyline method for solving simultaneous algebraic equations

Report the initial matrix profile structure – in `ija_prfl()`

 Compute new sequence of equations to reduce average profile – in `ija_optord()`

 Reset the airflow equation numbers in the node data structures

 Report the final matrix profile structure

Allocate remaining arrays depending on solution method – see [af_mat_alc\(\)](#)

Set pointers for off-diagonal elements in the link structures

Report solution notes to the LOG file

□ **ambtctm_get()**

[function in file weather.c]

Get the weather file data for the current simulation time.

Note: the `_ambtctm` structure defined in `weather.c` holds data for computing the contaminant mass fractions at the current simulation time (`_sim_time`) from data in the contaminant file at the times (`time0` and `time1`) that bound the `_sim_time`.

Set the contaminant file pointer

Reset time at end-of-day

Read the contaminant file until `time0 < _sim_time <= time1`

 Reset `time0` and corresponding data as necessary

 Check the data date

 Read and store the time and corresponding contaminant data

Determine the mass fraction of the first non-trace contaminant

Set contaminant data for current simulation; use linear interpolation if not at `time1`

Transfer data to the structure for `_ambt` node global variable

□ **ambtctm_init()**

[function in file weather.c]

Initial processing of the contaminant file.

Open the contaminant file – using `nxtopen()`
 Confirm file type and version
 Read the contaminant file start and end dates
 Confirm simulation dates within contaminant file dates
 Read number of species in file and allocate vectors in `_ambtctm` structure
 Read species names and determine which match simulation contaminants
 Position contaminant file at the simulation start date
 Read and store contaminant data for time 00:00:00
 For cyclic simulation save position in contaminant file of first time step
 Determine the mass fraction of the first non-trace contaminant
 The weather and contaminant files are read using the `nxtword()` utility function. This requires that the `_unxt` file pointer be set to the weather file or the contaminant file before each is read.

□ **Balance()**

[functions in file balance.c]

Determine loss coefficients to produce design duct flows.

Reset ambient weather:

No wind
 $P = 101\,325$ Pa for 1.2041 kg/m³ air; other P for other density
 Reset zone temperatures to 20 °C. Compute steady state flows

In `SetBalData()`:

Check that flows go in same direction as design values
 If not, signal error in .BAL file, write SIM file, terminate.
 Find associations between fans and balancing terminals.
 Write BAL file to indicate errors for display in ContamW
 Write SIM file, terminate.
 Set up data structures for balancing each fan/terminals system.
 Convert balancing fans to mass flow fans - `afe_cmf()`.
 Set up control value arrays, and set links to fans and terminals.

In `Balance()`:

Control the iteration of `SolveDB()` to determine control values that produce the desired terminal flows.
 In `SolveDB()`:
 Sum the flows through the terminals; use to adjust fan flow.
 Adjust flow control at each terminal toward convergent value.
 Write .BAL file reporting results of balancing -
 Success/failure indicator and
 flow rate and pressure rise for each fan, and
 relative flow and balance loss coefficient for each terminal.
 Write SIM file to report flow rate and ΔP for each path and fan.
 Free all memory allocated for balancing.

□ **calc_SP()**

[function in file simulate.c]

Calculate stack pressures for each flow path.

This function computes the P_1 , P_2 , and $\rho g(z_1 - z_2)$ terms from the Bernoulli equation:

$$\Delta P = \left(P_1 + \frac{\rho V_1^2}{2} \right) - \left(P_2 + \frac{\rho V_2^2}{2} \right) + \rho g(z_1 - z_2)$$

P_1 and P_2 are computed by a hydrostatic adjustment to the reference pressures in nodes 1 and 2. z_1 and z_2 are the elevations of the ends of the link relative to nodes 1 and 2. For airflow paths the elevations are identical. They may differ for ducts. The algorithm includes hydrostatic equations for incompressible and compressible air.

The final result is an estimate for the pressure drop across each opening based on the reference pressures in the connected nodes, the wind pressure, and the ‘stack’ pressure.

□ **calc_WP()**

[function in file simulate.c]

Calculate wind pressures for each flow path.

The wind pressure for links not connecting to the ambient node is zero. It is also zero if no wind pressure model has been specified for the link. Wind pressure models include constant value, linear interpolation using `lintld()`, spline interpolation using `splint()`, and a trigonometric interpolation using `htrigf()`, between user specified wind pressure values as a function of relative azimuth angle.

□ **ctrl_links()**

[function in file prjdata.c]

Set control nodes and links for simulation.

Set pointers to incoming data according to data type and number

Create linked list of control nodes in calculation sequence

Here it is necessary to set the pointers in sensors to the proper node data (for zones and junctions) and the proper link data (for paths and ducts). Note that nodes and links can be sensed and controlled making it is impossible to place the control data in the project file so that the referenced items have been defined.

The control structures are then placed in a linked list (starting at `_Ctrl0`) so that when a node is called while sequentially processing the list in `ctrl_sim()` the input signals will be current. The sequence was determined in `ContamW` in function `ctrl_order()` in file `prjpack.c`.

□ **ctrl_sim()**

[function in ctrlsim.c]

Process day-schedules, exposure-schedules, and controls.

Set clock-time (incl. DST) and day type

Loop through the week-schedule list

Loop through the control nodes list

Set any scheduled zone temperatures

Loop through the exposure-schedule list

□ **de_mat_alc()**

[function in file matset.c]

Allocate differential equation solution arrays.

Determine number of non-zero coefficients in symmetric matrix – see [ija_minset\(\)](#)

Allocate and set the `ija` index vector – see [ija_minset\(\)](#)

Allocate the `sa`, `b`, and `x` coefficients vectors

For symmetric skyline solution method allocate remaining arrays – in sky_alc() in file matset.c

For conjugate gradient method allocate *p*, *pp*, *r*, *rr*, *z*, *zz*, and preconditioning vectors

For Gauss elimination allocate *aa* full matrix.

□ **FillAf()**

[function in afesim.c]

Fill the Jacobian matrix for air flows.

Clear the Jacobian coefficients vector

Loop through all air links

Initialize the coefficients on the diagonal to dM/dt for variable pressure nodes

Initialize the coefficients on the diagonal to 1.0 for variable pressure nodes

Initialize the $\sum F$ and $\sum |F|$ values for each node.

Loop through all air links

 If a numerical derivative is being computed

 Compute flows for adjusted pressure drop

 For each flow (1 or 2) through the link

 Compute dF/dP if a numerical derivative is used

 Add flows to the $\sum F$ and $\sum |F|$ values

 Add dF/dP value to the appropriate coefficients in the Jacobian vector

Check for zeroes on the diagonal (would cause divide-by-zero in solution)

 Set any such nodes to constant pressure and recompute all values

Note the use a linked list of link (path) structures and of C pointer to functions:

```
for( pp=_pafp0; pp; pp=pp->next) // loop through all flow paths
{
    IX (*pf)(AF_PATH ) = pp->pe->pfunc; // pointer to simulation function
    . . .
    pf( pp ); // compute the flow(s) through path pp
    . . .
}
```

The linked list provides a very simple structure to loop through the flow paths. The pointers to functions provide a way to call the different types of flow paths without creating a switch statement to access each different type of path. The functions to evaluate each different type of flow path or duct are included in file afesim.c.

□ **Fill_Mf()**

[function in solvmf.c]

□ **ija_minset()**

[function in file matset.c]

Minimize the sparse matrix index vector.

Reduce the "full" sparse matrix index vector, *ijf*, to *ija* which does not include diagonal elements or upper triangle, if symmetric. Call once with count = 0 to determine size of *ija*, then call with count = 1 to fill *ija*.

□ **ija_optord()**

[function in file matset.c]

Determine the optimum (minimum profile) ordering for a set of simultaneous equations.

From the old sparse matrix index vector *ija* (see [Sparse Matrix Data Structure](#)) use the ACM Transactions On Mathematical Software (TOMS) 582 algorithm to compute a new optimum ordering for the variable pressure/mass/... rows. The TOMS 582 algorithm has been converted from Fortran to C in file `gpskc.c`. Return the new ordering in the *new* vector. Return 1 if the ordering has changed, 0 if it has not.

❑ **init_Af()**

[function in solvaf.c]

Use linear relations to determine initial guess for pressures.

Zero the sparse matrix array

Loop through all links setting airflow matrix coefficients for linear flow elements models

Solve the simultaneous linear equations for node pressures – see [solve_slae\(\)](#)

Set up the coefficients of the airflow matrix so that solution of the simultaneous linear equations will give an initial guess of pressures to start the Newton-Raphson method for solving the non-linear airflow equations. This initialization tends to save a few N-R iterations and is useful for cases involving large numbers of airflow nodes.

❑ **init_Mfn()**

[function in solvmf.c]

Initialize arrays for non- trace mass fraction calculations.

Set the initial mass fractions in the zones and junctions.

Set the initial mass fractions in the boundary layer controlled sinks.

Set up the matrix for computing the mass fractions – in `Fill_Mf()`

❑ **init_Mft()**

[function in solvmf.c]

Initialize arrays for trace mass fraction calculations.

Set the initial mass fractions in the zones and junctions.

Set the initial mass fractions in the boundary layer controlled sinks.

Set up the matrix for computing the mass fractions – in `Fill_Mf()`

❑ **LagDuct()**

[function in file solvSTS.c]

Use Lagrangian or Eulerian model to compute concentrations in a 1-D duct.

Special processing for zero-length duct and return

Determine Reynolds number and computation mode:

Eulerian for slow flow; Lagrangian for fast flow

If necessary, switch from Lagrangian to Eulerian – in `EulerSet()`

Transfer contaminant flow at duct inlet to appropriate cell

Solve for concentrations throughout the duct segment

Note that Lagrangian method uses a circular linked list to avoid shifting data

❑ **LagSim()**

[function in file solvSTS.c]

Compute 1-D contaminant convection-diffusion in duct system.

Process each duct segment in sequence (when all inputs are known)

Compute 1-D convection diffusion – see [LagDuct\(\)](#)

Add contaminant flow at duct exit to destination junction

When all inputs to destination junction are known,

Compute destination junction concentrations

□ **LinkSeq()**

[function in file solvSTS.c]

Put duct segments in ordered list in computational sequence.

Start with with all duct segments in linked 'todo' list

Determine number of input flows needed by each junction

Compute total flow into each junction

Iterate until 'todo' list reduced to null:

Loop through all links in old 'todo' list:

if the number of flows still needed at the 'from' junction is zero,
transfer link to 'done'; decrease flows needed at 'to' junction;

else

transfer link to new 'todo' list.

Reverse the loop direction at each iteration.

Return pointer to start of 'done' list

□ **LinkSim()**

[function in file solvSTS.c]

Compute contaminant concentrations in duct system.

Process each duct segment in sequence (when all inputs are known)

Add contaminant flow through duct to destination junction

When all inputs to destination junction are known,

Compute destination junction concentrations

□ **mat_init()**

[function in file contamx.c]

Allocate the matrices for solving airflows and mass fractions.

Allocate _facBins to report airflow solution iterations

Set up airflow matrices – see [af_mat_set\(\)](#)

Set numerical derivative flags – in [setNmDrv\(\)](#)

If using non-trace contaminants

Copy control parameters from _rcdat to _MFn_mat

Set up non-trace mass matrices – see [mf_mat_set\(\)](#)

If using trace contaminants

Copy control parameters from _rcdat to _MFt_mat

Set up trace mass matrices – see [mf_mat_set\(\)](#)

□ **mat_term()**

[function in file contamx.c]

Free the allocated matrices for solving airflows and mass fractions.

All heap memory allocated in [mat_init\(\)](#) is freed in reverse order in [mat_term\(\)](#). Other memory allocated earlier in [prj_read\(\)](#), [contamx\(\)](#), and [main\(\)](#) is also freed.

□ **mf_mat_set()**

[function in file matset.c]

Set up contaminant matrices.

Transfer control parameters from _rcdat to the structure for mass fraction calculations

Set up vector of pointers to nodes with constant pressure nodes last
Determine positions of all non-zero elements in flow solution matrix – in `acolset()`
Allocate and fill the *ijf* vector sparse data mapping – see [Sparse Matrix Data Structure](#)
For the skyline method for solving simultaneous algebraic equations
 Report the initial matrix profile structure – in `ija_prfl()`
 Compute new sequence of equations to reduce average profile – see `ija_optord()`
 Reset the airflow equation numbers in the node data structures
 Report the final matrix profile structure – in `ija_prfl()`
Allocate remaining arrays depending on solution method – see `de_mat_alc()`
Set pointers for off-diagonal elements in the link structures
Report solution notes to the LOG file

□ **prj_read()**

[function in file prjread.c]

Read the project (PRJ) file.

This and related functions in the same file have been written to read the project file for ContamW or for ContamX depending on the definition of the macro CTMW being 1 or zero, respectively. ContamW uses data such as sketchpad data and units for displaying parameters that are not needed in ContamX.

The ContamW version of `prj_read()` includes code for updating previous versions of the project file. Sections of the project file are read by functions within the `prjread.c` file. Sections are ordered so that items referenced by pointers in the data structures are read before they are referenced. For example, a flow path must refer to a flow element, so flow elements are read before flow paths. If a significant error is encountered while reading any section, input processing is terminated and an error code is returned by `prj_read()`.

The ContamW version calls functions to check and process pointers in the controls because control nodes have pointers to other control nodes. The ContamX version includes related processing for controls and functions to create the airflow network by converting zones and junctions to network nodes and converting paths and ducts to network links – see [sim_data\(\)](#).

Several temporary data structures are allocated, used, and then freed. For example, most named elements such as schedules, filters, flow elements, etc., are stored in individually allocated structures with an allocated vector of pointers to those structures. When such an element is later referenced by sequence number, that number is converted to the pointer to the element according to the vector of pointers.

□ **sim_data()**

[function in file prjdata.c]

Convert project data to forms used for simulation.

Transfer elements from the `_rcdat` structure to individual global variables
Transfer steady-state weather data to ambient airflow node.
Transfer default contaminant concentrations to ambient airflow node.
Set ductwork junction volumes – in `jct_set()`
Set airflow node data – in `afnd_set()`
Set airflow link data – in `afpt_set()`
Define implicit flow elements: simple AHS and duct leaks
Set implicit flow paths
Check completeness of airflow network – in `airnet_check()`

Create linked list of day-schedules
 Create linked list of week-schedules
 Complete trace and non-trace source linked lists
 Copy boundary-layer non-trace sinks to separate list
 Copy boundary-layer trace sinks to separate list
 Convert occupancy zones to node pointers
 Create linked list of exposure structures
 Complete control links – see [ctrl_links\(\)](#)

□ **sim_init()**

[function in file simulate.c]

Initialize flows and compute steady-state mass fractions.

If using non-trace contaminants

 Compute gas constant for mixed air in each zone

 Compute mass of air in each zone

If using WPC File

 Get wind pressure from WPC File [in [WPC_get\(\)](#)]

Compute wind pressures – see [calc_WP\(\)](#)

Compute stack pressures – see [calc_SP\(\)](#)

Initial guess for pressures by linear flow approximation - see [init_Af\(\)](#)

Reset convergence and time step values

Iterate through density changes:

{

 Solve for airflows by Newton-Raphson iteration with

 simple trust region method – see [SolveAfstr\(\)](#)

 or simple under-relaxation – see [SolveAfsur\(\)](#)

 Break loop if densities have converged

 Tighten N-R convergence for next iteration

 If using non-trace contaminants

 Compute non-trace mass fractions [in [init_Mfn\(\)](#) and [solv_DE\(\)](#)]

 Compute gas constant for mixed air in each zone

 Compute mass of air in each zone

 Re-compute stack pressures - see [calc_SP\(\)](#)

}

Initialize trace and non-trace mass transfer data – see [init_Mfn\(\)](#) and [init_Mft\(\)](#)

Compute trace mass fractions [call [solv_DE\(\)](#)]

If not initializing for transient simulation, report results – in [simout\(\)](#)

□ **sim_loop()**

[function in file simulate.c]

Perform transient simulation.

Set parameters to control output and time step loop

Display information (for DOS window only)

While (not done):

{

 Check for/process user interruption

 Initialize daily summary results as needed

```

Increment time of day by one time step
Get contaminant data - see ambtctm\_get\(\)
Get weather data – see weather\_get\(\)
Compute airflows and mass fractions for current time – see sim\_step\(\)
Write results if time of day is on a listing time step – in simout\(\)
Compute flows for daily summary
Compute mass fractions for daily summary
Check for reaching non-cyclic end point – set done
If (time = 24:00:00)
    Write daily summary results
    Check for cyclic convergence – set done
    If (not done)
        Advance day count; reset time of day to 00:00:00
        Write results
        Reset read position in weather and contaminant files
    }

```

```

Write summary results – in simout\(\)
Close results files, weather file, and contaminant file

```

□ **sim_step()**

[function in file simulate.c]

Simulate a single time-step.

```

Save mass and mass fraction values at end of last time step
Get contaminant data - see ambtctm\_get\(\)
Get weather data – see weather\_get\(\)
Get data from CVF, DVF, and WPC files, if used
Set all schedule values and process controls – see ctrl\_sim\(\)
Compute scheduled mass gains – in cse\_sim\(\)
Compute wind pressures – see calc\_WP\(\)
If using STS method
    Compute stack pressures - see calc\_SP\(\)
    Compute airflows and then concentration by STS – see Solve\_STS\(\) @@@
    Compute exposures, filter loading, and ambient impact
else loop until zone masses converge
{
    Compute stack pressures - see calc\_SP\(\)
    Solve for airflows by Newton-Raphson iteration with
        simple trust region method – see SolveAfstr\(\)
        or simple under-relaxation – see SolveAfsur\(\)
    If using non-trace contaminants:
        Compute non-trace mass fractions – see solve\_Mfn\(\)
    Re-compute gas constant for mixed air in each zone
    Compute mass in each zone
        Check zone mass convergence
    }
Solve trace species mass fractions – see solve\_Mft\(\)

```

Compute exposure values

❑ **solve_DE()**

[function in file solvde.c]

Solve differential equations by trapezoidal integration and mixed direct/iterative solution of simultaneous equations.

Solve $[A] \{x\} = \{b\}$. Temporary $[A]$ and $\{b\}$ created so new $\{xp\}$ can be calculated.

❑ **Solve_Mfn()**

[function in solvmf.c]

Solve for the mass fractions of the non-trace contaminants.

This function uses the same process described in [solve_Mft\(\)](#) except that only the non-trace contaminants are considered.

❑ **Solve_Mft()**

[function in solvmf.c]

Solve for the mass fractions of the trace contaminants.

If on the first iteration during a time step:

 Compute the constant (predictor) portions of the difference equation

 Fill the $[A]$ matrix – in [Fill_Mf\(\)](#)

 Solve the difference equations for values at the end of the time step

 Transfer results to node and sink structures

❑ **Solve_STS()**

[function in file solvSTS.c]

Use STS method to compute contaminant concentrations.

Initialize contaminant summation variables to zero

Compute scheduled mass gains – in [cse_sim\(\)](#)

Compute airflows and node pressures

 for simple trust region method – see [SolveAfstr\(\)](#)

 for simple under-relaxation method -- see [SolveAfsur\(\)](#)

If variable density, recompute nodde air masses

Test stability limits; report fatal error if unstable.

Transfer computed flows to ducts, terminals and leaks.

Add zone-to-zone contaminant flows to zone cells – in [Flow_STSzz\(\)](#)

Add duct-to-zone contaminant flows to zone cells – in [Flow_STSdz\(\)](#)

Compute contaminant concentrations in zones

 for fully mixed zones – in [MixZsim\(\)](#)

 for 1-D convection-diffusion zones – in [CDZsim\(\)](#)

Add zone-to-duct contaminant flows to duct cells – in [Flow_STSzd\(\)](#)

Solve duct junction concentrations

 Determine calculation sequence – see [LinkSeq\(\)](#)

 For 1-D duct model – see [LagSim\(\)](#)

 For simple duct model – see [LinkSim\(\)](#)

 For duct junction temperatures – in [LinkTSim\(\)](#)

Test for valid (≥ 0) concentrations;; report fatal error if invalid

Adjust non-trace concentrations and reset air thermal properties

□ SolveAfstr()*[function in solvaf.c]*

Solve symmetric simultaneous non-linear algebraic equations by Newton-Raphson with simple trust region.

Dr. David Lorenzetti developed this function. It provides a more robust algorithm than the earlier under-relaxation method. It is the default method and should be used unless some particular problem is encountered.

□ SolveAfsur()*[function in solvaf.c]*

Solve symmetric simultaneous non-linear algebraic equations by Newton-Raphson with simple under-relaxation.

Loop until convergence or iteration limit

Fill the airflow solution matrix – see [FillAf\(\)](#)

Check convergence

Solve simultaneous linear equations for dP values - see [solve_slae\(\)](#)

Under-relax the pressure adjustments

Adjust the airflow node pressures

Clear any numerically insignificant airflows

This method was used in ContamW 1.0 and the earlier DOS versions of CONTAM.

□ solve_slae()*[function in solvse.c]*

Solve symmetric simultaneous linear algebraic equations.

Select solution method:

Skyline:

Transfer data from sparse to skyline arrays – in [fill_sky_s\(\)](#)

L-U factor skyline matrix – in [luf_sky_s\(\)](#)

Solve simultaneous equations directly – in [lus_sky_s\(\)](#)

Preconditioned conjugate gradient (PCG):

Transfer data from sparse to preconditioning array – in [fill_ssm\(\)](#)

Perform incomplete Cholesky decomposition – in [luf_chl_c\(\)](#)

Solve simultaneous equations iteratively – in [sa_pcg\(\)](#)

Gauss elimination:

Transfer data from sparse to full arrays – in [fill_ges\(\)](#)

L-U factor the full matrix – in [luf_ge\(\)](#)

Solve the simultaneous equations – in [lus_ge\(\)](#)

The skyline method does a direct solution using a reduced array. It is more reliable than PCG and usually faster for small problems. The PCG method is faster for large problems if it converges. The gauss elimination method is only to provide a well-established solution for comparison with the other methods. It is prohibitively slow for larger problems.

□ STSinit()*[function in file STSinit.c]*

Set up data structures for STS method

In STSinit()

Merge trace and non-trace source/sink lists

Create cells for 1-D duct model – in [LAGinit\(\)](#)

Create cells for mixed and 1-D zone models – in CDZinit()
 Convert links to airflow nodes to links to cells – in CellLink()
 paths, terminals, sensors, source/sinks, occupansy positions
 Allocate 1-D Convection-Diffusion tridiaagonal matrices – in alc_1DCD()

□ **weather_get()**

[function in file weather.c]

Get the weather file data for the current simulation time.

Note: the `_weather` structure defined in `weather.c` hold data for computing the weather data at the current simulation time (`_sim_time`) from data in the weather file at the times (`time0` and `time1`) that bound the `_sim_time`.

Set the weather file pointer

Reset time at end-of-day

Read the weather file until `time0 < _sim_time <= time1`

 Reset `time0` and corresponding data as necessary

 Check the data date

 Read and store the time and corresponding data

 Reset wind direction for interpolation

Set weather data for current simulation; use linear interpolation if not at `time1`

Convert ASHRAE humidity ration to Contam mass fraction of H₂O

Transfer data to global variables

□ **weather_init()**

[function in file weather.c]

Initial processing of the weather file.

Open the weather file – using `nxtopen()`

Confirm file type and version

Read the weather file start and end dates

Confirm simulation dates within weather file dates

Read and store day-of-week, day-type, DST, and Tg for all dates

Position weather file at the simulation start date

Set day-data globals: `_daytyp`, `_dayofw`, `_DSTind`

Read and store weather data for time 00:00:00

For cyclic simulation save position in weather file of first time step

The weather and contaminant files are read using the `nxtword()` utility function. This requires that the `_unxt` file pointer be set to the weather file or the contaminant file before each is read.

B.3.3 Utility Functions

Compiler dependent functions have been grouped in file **config.c**. Memory allocation and de-allocation functions are in file **heap.c**. Other utility functions are in file **utils.c**. Three different compilers have been used – two for the earlier DOS-based CONTAM93 and CONTAM96 programs, and Visual C++ versions 6.0 and .NET for the current MS Windows based program.

□ **File path processing**

[functions in file config.c]

These functions are used to create output file paths that will open in the proper directory.

pathsplit() converts a path into its component parts (drive, directory, name, and extension).

pathmerge() does the opposite.

❑ **Error handling**

[functions in file config.c]

Error messages are generated by a call to the **error()** function. Passed parameters include the *severity* of the error, the name and line number of the file from which **error()** is called, and an indefinite number of strings describing the error. The function maintains a count of ‘severe’ errors. The global variable `_emode` (in `setEmode()`) controls display of the error message.

If *severity* ≥ 0 , the function will merge the message strings into a single string and create another string reporting the file name and line number – i.e., the source of the error. The severity, source, and error message are then displayed in a dialog box or the DOS window depending on `_emode`, and they are written to the LOG file.

If *severity* > 2 , ContamX will be terminated by calling **finish()** which also closes all open files.

If *severity* = 2, the count of severe errors is incremented.

If *severity* < -1 , the count of severe errors is reset to zero.

If *severity* = -1, the only action is to return the current count of severe errors.

If the count of severe errors reaches 10 and the dialog box is in effect, the user will be asked if he wants to terminate the simulation. If he does not, the count of severe errors is reset to zero.

This prevents an endless display of error messages that could occur in some circumstances.

The **lognote()** function is used to write similar informative messages to the LOG file only. It is a debugging tool not intended to be informative to the general ContamX user.

❑ **Console input**

[functions in file config.c]

getkey() is used to read a single keystroke. The *wait* parameter tells **getkey()** to wait until a key is pressed or to return immediately if a key has not been pressed.

noyes() obtains a no or yes response to a query. It returns 0 for no or 1 for yes. The query is presented in a dialog box or the DOS window depending on `_emode`.

❑ **Heap tests**

[functions in file config.c]

memrem() reports the memory still available in the heap. It is useful for determining if heap memory has been properly freed. It works with the Borland and Watcom compilers but is not available in Visual C++.

memwalk() reports the status of the allocated heap memory and then displays the status of every heap allocation. This seems to not be working with the current Visual C++.

nptest() reports if a value has been written to address 0. This occurs when a value is written to a null pointer. The test is not applicable to Visual C++.

❑ **Math error processing**

[functions in file config.c]

fltterr() and **matherr()** trap various floating point errors, print error messages, and terminate the program.

□ Heap processing

[functions in file heap.c]

All memory allocations and de-allocations should go through `alc_e()` and `fre_e()` to allow some useful heap checking options based on the definition of the macro `MEMTEST`: 1 = test guard bytes; 2 = log actions; and 0 = no tests. When `MEMTEST > 0`, four guard bytes are added before and after the normal heap memory allocation. These guard bytes are used to test writes and reads beyond the ends of the allocated vector -- especially useful for off-by-one indexing. Based on an idea and code by Paul Anderson, "Dr. Dobb's C Sourcebook", Winter 1989/90, pp 62 - 66, 94. When `MEMTEST > 1`, every allocation and de-allocation is reported in the LOG file.

`alc_e()`, `fre_e()`, and `chk_e()` allocate, de-allocate, and check the guard bytes of a single block of memory. The check is automatically performed at de-allocation.

`alc_mc()`, `fre_mc()`, and `chk_mc()` allocate, de-allocate, and check a rectangular matrix (2-dimensional array) given minimum and maximum row and column indices. Memory is allocated contiguously.

`alc_mvc()`, `fre_mvc()`, and `chk_mvc()` allocate, de-allocate, and check a matrix with variable length rows (used to store skyline method coefficients).

`alc_v()`, `fre_v()`, and `chk_v()` allocate, de-allocate, and check a vector (2-dimensional array) given the minimum and maximum indices.

`alc_ec()`, `fre_ec()`, and `alc_eци()` speed and simplify allocation and de-allocation for many small structures such as the CONTAM zones, paths, schedules, flow elements, etc. Small structures are 'allocated' within a larger allocated block which is initially created with `alc_eци()`. A single call to `fre_ec()` will de-allocate the entire block; individual structures cannot be freed. This saves quite a bit of memory allocation overhead and is quite a bit faster than calling `alloc()` for each small structure. Based on an idea and code by Steve Weller, "The C Users Journal", April 1990, pp 103 - 107.

□ Text File Reading Functions

[functions in file utils.c]

`nxtopen()`, `nxtline()`, `nxtwrд()`, and `nxtclose()` are used to process the project, weather, and contaminant ASCII text data files. These functions process the file designated `FILE *_unxt`. A key feature of these functions is their handling of erroneous input values.

□ Format Conversion Functions

[functions in file utils.c]

`dblcon()` converts a string to an 8-byte real variable.

`fltcon()` converts a string to a 4-byte real variable.

`longcon()` converts a string to a 4-byte integer variable.

`intcon()` converts a string to a default size integer variable.

`timecon()` converts a string (hh:mm:ss) to seconds past midnight.

`datecon()` converts a string (dd/mm, mm/dd) to a day-of-year number (1 - 365).

`datxcon()` converts a string (mm/dd) to a day-of-year number (1 - 365) (for Excel compatibility).

`fltstr()` converts a 4-byte real variable to a string.

intstr() converts a 4-byte integer variable to a string.

timestr() converts a 4-byte integer (seconds past midnight) to a string of the form: hh:mm:ss.

datestr() converts a day-of-year integer (1 - 365) to a string (ddmm).

datxstr() converts a day-of-year (1 - 365) to a string (dd/mm) (for Excel compatibility).

□ **Input Conversion Functions**

[functions in file utils.c]

readR8() reads the next word from file `_unxt` and converts it to an 8-byte real variable.

readR4() reads the next word from file `_unxt` and converts it to a 4-byte real variable.

readI4() reads the next word from file `_unxt` and converts it to a 4-byte integer variable.

readI2() reads the next word from file `_unxt` and converts it to a default size integer variable.

readHMS() reads the next word from file `_unxt` and converts it to a 4-byte integer time value.

readMD() reads the next word (ddmm, mmdd) from file `_unxt` and converts it to a day-of-year number (1 - 365).

readMDx() reads the next word (dd/mm) from file `_unxt` and converts it to a day-of-year number (1 - 365).

The “default size” integer is 2-bytes for a 16 bit compiler and 4-bytes for a 32 bit compiler. CONTAM presently enforces the rule that such an integer must be in the range -32767 to +32767, i.e., fits in a 2-byte integer. This limits the maximum number of zones, paths, etc., to 32767. Relaxing this limit will also require the conversion of small integers in the data structures.

B.3.4 Sparse Matrix Data Structure

Calculation of both the airflows and mass fractions involve solving simultaneous linear algebraic equations. These simultaneous equations can be expressed in matrix form: $[A] \{x\} = \{b\}$ where $\{b\}$ is an N-element vector, $[A]$ is an N by N matrix, and the solution $\{x\}$ is an N-element vector. As N gets large (CONTAM projects with over 3000 nodes have been created) the size of **A** becomes excessive and the execution time using full matrix methods becomes even more excessive. When N is large only a small portion of elements of **A** are non-zero, therefore sparse matrix techniques should be used to solve such problems. There are also methods to store only the non-zero elements of **A**.

Press et al. [Press 2001, p78] present a good method for storing sparse matrices:

"To represent a matrix **A** of dimension $N \times N$, the row-indexed scheme sets up two one-dimensional arrays, call them *sa* and *ija*. The first of these stores matrix element values in single or double precision as desired; the second stores integer values. The storage rules are:

- The first N locations of *sa* store **A**'s diagonal matrix elements in order. (Note that diagonal elements are stored even if they are zero; this is at most a slight storage inefficiency, since diagonal elements are nonzero in most realistic applications.)
- Each of the first N locations of *ija* stores the index of the array *sa* that contains the first off-diagonal element of the corresponding row of the matrix. (If there are no off-diagonal elements for that row, it is one greater than the index in *sa* of the most recently stored element of a previous row.)
- Location 1 of *ija* is always equal to $N + 2$. (It can be read to determine N.)

- Location $N + 1$ of *ija* is one greater than the index in *sa* of the last off-diagonal element of the last row. (It can be read to determine the number of nonzero elements in the matrix, or the number of elements in the arrays *sa* and *ija*.) Location $N + 1$ of *sa* is not used and can be set arbitrarily.
- Entries in *sa* at locations $\geq N + 2$ contain **A**'s off-diagonal values, ordered by rows and, within each row, ordered by columns.
- Entries in *ija* at locations $\geq N + 2$ contain the column number of the corresponding element in *sa*.

In ContamX the coefficients for the airflow Jacobian and for the mass fraction calculations are first stored into this sparse structure. They may then be transferred to another data structure designed for the solution algorithm being employed. There is a small run-time overhead for this process, but it is very flexible when it is not known that a single best solution method exists for all cases.

Appendix C ContamW Program Documentation

C.1 Introduction

This part presents an overview of the program structure, data and logic of the ContamW, CONTAM's graphical user interface (GUI). It does not present the functions in as much detail as was done for the solver, because of the large number of files and functions associated with the GUI. The intent is to present enough information to provide the user of this document with a starting point to understanding the program structure. The GUI provides a means of creating and viewing the multi-zone model, and creates the PRJ file for use by CONTAM's simulation engine, ContamX. The GUI also serves as a means to view the simulation results that are output by ContamX.

C.2 Development Environment

ContamW is an event-driven program developed solely for the Windows platform using the WIN32 Application Programming Interface (API) and written in the C programming language. Version 2.1 was developed using Microsoft Visual C++ versions 6.0 and .NET. As of the release of this document, the source code consists of 87 c-code files (.c), 33 header files (.h), 8 context-sensitive help header files (.hh), and 16 resource-related files. The graphical charting routines were developed using a third-party charting tool OletraChart 6.0. This software is required in order to build the Visual Studio project.

C.3 Program Structure

The figure below shows the GUI which is made up of two windows – one “superimposed” on top of another. The *SketchPad* window is the large white portion that is “superimposed” on the larger *Main* window.

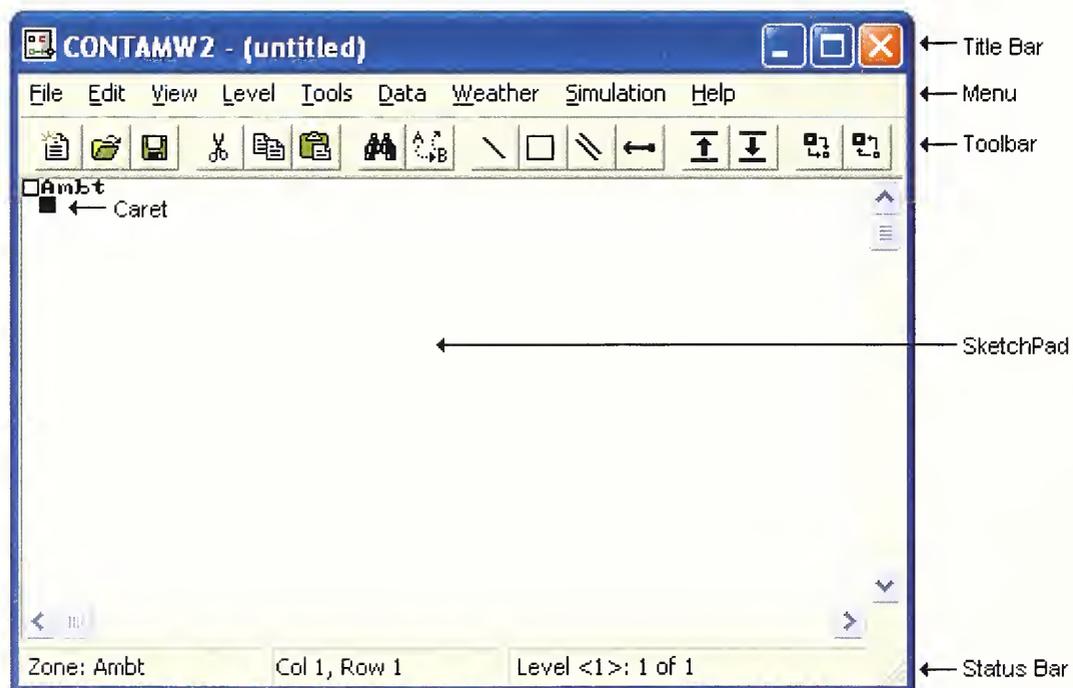


Figure – The ContamW GUI

C.3.1 Main Program and Message Loop

Every Windows program must have a *WinMain()* function which is the entry point of the program. This function is equivalent to the *main()* function in a standard C program. In ContamW, this function is located in the file *contam.c*. *WinMain()* performs program initialization and executes the main message loop of the program.

As this is an event-driven program, it contains a message loop that gets and dispatches messages from and to the operating system respectively. This loop gets messages that are directed to the ContamW program from the Windows message queue then dispatches the message back to the operating system which then calls the appropriate *window procedure* of ContamW for processing.

C.3.2 Window Procedures

Each window displayed by ContamW, including dialog boxes and property sheet pages, has an associated *window procedure*. Window procedures are called by Windows after ContamW dispatches messages associated with a particular window back to the operating system. These messages contain parameters that indicate to the operating system which window procedure to call, a message identifier, and message-specific parameters as required. Window procedures contain sections of code known as *message handlers* that are executed when the window procedure receives a particular message.

ContamW contains approximately 90 window procedures. Two of these procedures, *WndProc()* and *SketchPadWndProc()*, are associated with the two main windows of the program and the remaining procedures are associated with individual dialog boxes, property sheet pages and miscellaneous windows that display charts and simulation results. *WndProc()* handles the menu and associated toolbar commands and passes keyboard messages to the SketchPad window procedure *SketchPadWndProc()*. *SketchPadWndProc()* handles all SketchPad related commands including drawing walls, ducts and controls and placing icons onto the SketchPad.

C.4 Program Data

Before describing the ContamW data structures, it is important to understand the basic philosophy of the program. ContamW is not meant to depict an exact physical representation of a building, but to portray a building in a manner that is representative of the multizone modeling perspective. Therefore, the physical dimensions of the representations of building zones are not important, but the interconnectivity or topology of the zones is important. The ContamW SketchPad provides a means of describing a building in a manner that constrains the representation to the multizone modeling domain.

C.4.1 SketchPad Data

From the users perspective, the SketchPad consists of a two-dimensional grid of equal-sized cells in which icons can be placed. The SketchPad is actually represented as a set of global 2D arrays having equal dimensions indicating the width (columns) and height (rows) of the SketchPad. These arrays are used to indicate the symbols located on the SketchPad for each building level and to maintain references to lists of data elements that contain the detailed information describing the building elements that make up each building level. The following is a list of the SketchPad arrays along with a description of the values stored in each array. Values are defined in *contam.h*. In the list below, the first array in each set contains information related to the building level currently displayed on the SketchPad, and the second (preceded by *_SL*) contains information related to the level below the current level, i.e. the sublevel.

<p><code>_Sketch</code>[row][col] <code>_SLSketch</code>[row][col]</p>	<p>Contains identifiers indicating a symbol for <i>Zones, Flow path, Simple AHS, Supply, Return, Note, Source Sink and Exposure</i> (0 => empty cell)</p>
<p><code>_Walls</code>[row][col] <code>_SLWalls</code>[row][col]</p>	<p>Contains identifiers indicating the wall symbol. Symbol values from <i>WL_EW</i> to <i>WL_NESW</i>.</p>
<p><code>_Zones</code>[row][col] <code>_SLZones</code>[row][col]</p>	<p>Indicates zone number for every cell on the SketchPad. AMBT 32766 => ambient zone ZNDF -2 => undefined zone WALL 32767 => wall location</p>
<p><code>_Paths</code>[row][col] <code>_SLPaths</code>[row][col]</p>	<p>Contains the id number of each symbol that represents a building component. The following types of components warrant values in the <code>_Paths</code> array: <i>Airflow paths, Ducts, Junctions, Zones</i> and <i>Notes</i>.</p>
<p><code>_Ducts</code>[row][col] <code>_SLDucts</code>[row][col]</p>	<p>Contains identifiers indicating a symbol for all <i>Ducts, Junctions</i> and <i>Terminals</i>. Values are from <i>DCT_EW</i> to <i>IOJ_CB</i>.</p>
<p><code>_Links</code>[row][col] <code>_SLLinks</code>[row][col]</p>	<p>Contains identifiers indicating controls-related symbols. Values are from <i>CL_EW</i> to <i>CTRL_P</i>.</p>
<p><code>_Ctrls</code>[row][col] <code>_SLCtrls</code>[row][col]</p>	<p>Contains a ctrl/link number for each control symbol in the corresponding <code>_Links</code> array.</p>

Table – Global SketchPad arrays

An example of a project file and associated SketchPad arrays is presented in spreadsheet form in Appendix D.

C.4.2 Building Organization

ContamW buildings (projects) are organized by building levels that contain building component information. This organization is represented by a doubly linked list of level data structures (*LEV_DATA*) as illustrated in the figure below. This list is used to populate the SketchPad arrays whenever the user selects a different level to display. A global pointer is used to access the current level and the levels above and below the current level are accessed via each level's pointer to the level above and below. Each level contains a pointer to a list of icon data for all of the icons contained on the level. Icon data structures (*ICON_DAT*) contain information that includes the component number, icon identifier, and column and row in which the icon is located on the SketchPad, i.e., in the SketchPad arrays.

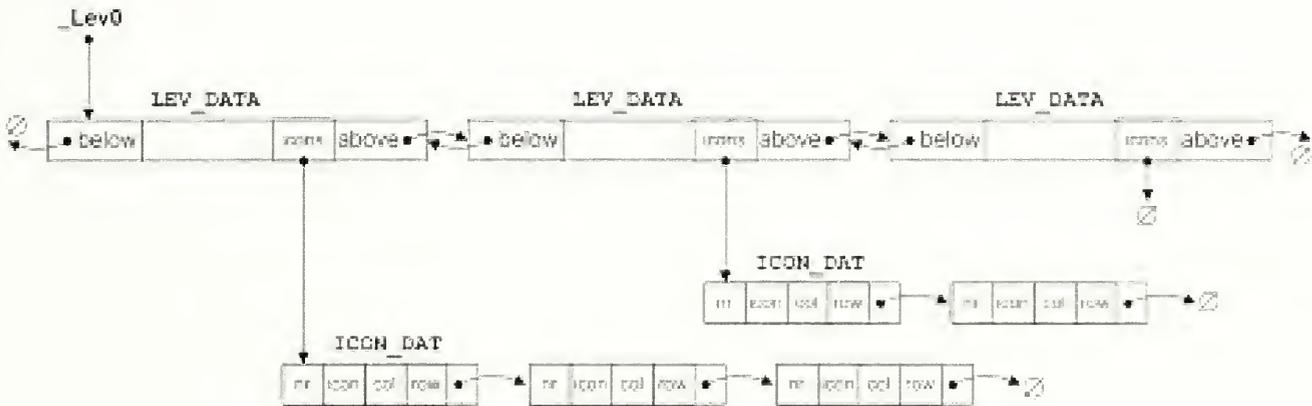


Figure – Schematic of Building Level and Icon Data Storage

C.4.3 Building Component and Element Data

The data used to describe a building in ContamW is basically divided into two types: building *component* and building *element* data. In terms of how the data is managed by ContamW – these data are referenced-by-number (RefByNum) and referenced-by-name (RefByName) respectively. RefByName data are referenced by a unique, user-defined name, and RefByNum data are referenced by a number assigned by ContamW.

RefByNum data are those that are considered specific to a building (project) and include *zones*; *airflow paths*; *simple air handling systems*; *duct segments*, *junctions* and *terminals*; *source-sinks*; *occupants* and *controls*. These components only exist in relation to a given building, i.e., they physically relate to other components of the building and are represented on the SketchPad by individual icons. RefByName data are those types of data that can be shared between building components and even between different projects via ContamW libraries. RefByName elements include *airflow elements*, *duct flow elements*, *wind pressure profiles*, *species*, *source-sink elements*, *filter elements*, *kinetic reaction elements*, *non-occupant schedules* and *annotations*.

RefByNum data can refer to RefByName data, and more than one RefByNum component can refer to the same RefByName element. For example an *airflow path* (e.g. a doorway) connects two zones of a building is defined in part by its location within the building. The airflow characteristics of this *airflow path* are represented by a pressure-flow model characterized by an *airflow element*. There may be several doors having the same airflow characteristics; therefore, each airflow path (e.g. door) can refer to the same airflow element.

Building component data are stored in linked-lists as shown in the figure below that illustrates the concept for airflow paths and elements. These lists are referred to by and accessed through arrays of pointers (e.g. *_PathList*) that are used to maintain the components in order according to

their location within the building. Building components are numbered starting at the top level in the upper left hand corner of the SketchPad moving left to right and down the SketchPad then proceeding down through each level in the same manner. Whenever a project file is saved, reordering of the array of pointers and renumbering of the building components will be performed as necessary.

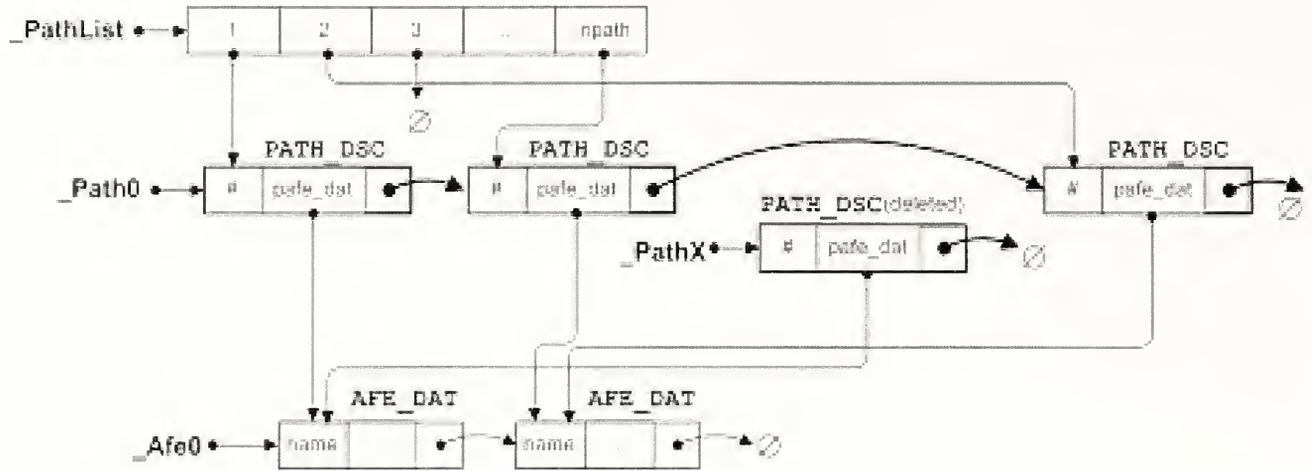


Figure – Schematic of Airflow Path Component (*PATH_DSC*) and Element (*AFE_DAT*) Data Storage

The figure above shows some of the global variables that are used to maintain the list of airflow paths (components) and airflow elements that are referenced by the paths. `_PathList` is the array of pointers used to maintain the SketchPad order of the paths, `_Paths0` is a pointer to the head of the linked list of `PATH_DSC` structures. `_PathX` is a pointer to the head of a list of paths structures that have been deleted, but could be reused when new paths are added.

The routines used to create new paths are contained in the file `paths.c`. These routines are typical of the building component management routines:

- Path creation functions: `path_new()`, `path_dflt()`, `path_add()`, `path_put()`
- Path deletion functions: `path_del()`, `path_delete()`
- Path editing functions: `path_old()`, `path_get()`

The routines for managing airflow element related data are found in `afedlg.c` (airflow element specific) and `cutils.c` (building elements in general).

- Airflow element creation functions: `afelmt_new()`, `elmt_insert()`
- Airflow element deletion functions: `elmt_delete()`
- Airflow element editing functions: `afelmt_old()`, `elmt_replace()`

These data are managed by a set of lower level routines that perform all memory allocation/deallocation for ContamW. These routines are located in the file `heap.c`. These memory management routines are used to minimize the number of actual calls to the standard-C `malloc()` and `free()` functions by maintaining memory used by ContamW in larger memory blocks with which the routines in `heap.c` operate.

C.5 Program Logic

As previously indicated, ContamW is an event-driven program, which basically means that the program functions that are executed depend on user-generated commands, i.e., user events. These events fall into the following basic categories: saving/retrieving project files, drawing

building components, editing building components, running simulations and working with simulation results. The following sections describe the program logic that handles these events.

C.5.1 Message (Event) Handlers

Selecting an item from the menu or double-clicking on a SketchPad icon are examples of events for which message handlers are required. In ContamW, all menu selection events are handled by the *WM_COMMAND* message handler *WndProc_OnCommand()* of *WndProc()*. Each menu selection is associated with a resource identifier that is used as a case selector of a switch statement in the *WM_COMMAND* message handler. For example, if the user selects the File-Open command, the *WM_COMMAND* message is sent with the identifier of the File-Open menu resource *IDM_FILE_OPEN*. In the case of a double-click mouse event on the SketchPad window, both the *WM_LBUTTONDOWN* and *WM_LBUTTONDBLCLK* commands are sent to *SPWndProc_OnLButtonDown()* message handler of *SketchPadWndProc()*.

C.5.2 Saving and Retrieving Project Files

CONTAM project files are ASCII files that maintain all information related to a CONTAM project. The project file format is described in detail in the section [Project File \(.PRJ\)](#). Project file related commands are executed via the File menu and are handled within the *WM_COMMAND* message handler of the main window procedure *WndProc()*. The File menu identifiers are all of the form *IDM_FILE_X* where *X* is one of the following *NEW*, *OPEN*, *SAVE*, *SAVE_AS*, or *EXIT*. File input (reading) and writing (saving) are handled by the *prj_read()* and *prj_save()* functions contained in the files *prjread.c* and *prjsave.c*. These two functions call lower level functions each of which handles a specific section of the project file as described in the project file documentation.

There are a few other features of note related to project file processing. If the user attempts to open a project file of an older version, conversion functions will be called to convert to a format compatible with the current version of the program. These routines are located in the file *c10toc20.c*. If the user clicks the OK button after viewing a building component data dialog box, a global flag *_saveprjf* is set to indicate that the file should be saved and a warning will be displayed if the user attempts to exit the program with this flag set. Also, the project file will be saved automatically when the user attempts to perform a simulation, because the simulation program reads the project file and it is a separate executable from the GUI.

C.5.3 SketchPad Drawing

Icons are placed upon the SketchPad using either one of the four drawing tools or the popup icon placement menu.

C.5.3.1 Drawing Tools

There are four drawing tools: line and box for walls, ducts and control links. Tool selection is performed via either the Tool menu or associated toolbar buttons. The message handlers contained in *WndProc()* associated with each tool set a global flag to indicate which tool is selected: *_bWall*, *_bBox*, *_bDuct* or *_bLink*.

The actual drawing process is activated once the user clicks the left mouse button (or hits the Enter key) at which point the drawing mode is activated and indicated by setting the flag *_bDraw*. When in the drawing mode, all mouse commands and keyboard arrow key commands are captured by the SketchPad window and converted to the ContamW-specific message *CT_DRAW* handled by the function *SPWndProc_OnCtDraw()*. *SPWndProc_OnCtDraw()* then calls specific drawing functions depending on the selected drawing tool as indicated by the

drawing tool flags. Drawing is finalized by clicking the left mouse button activating the message handler *SPWndProc_OnLButtonDown()*. This message handler will then call the drawing tool dependent routines to validate the drawing and place the proper icons into SketchPad arrays. If the drawing was valid then the screen can be updated to reflect the placement of the building components, e.g., walls, ducts or control links. This is performed via the *InvalidateRect()* command that causes the SketchPad window to redraw or repaint itself. SketchPad “painting” is performed via the *WM_PAINT* message handler *SPWndProc_OnPaint()*.

The following is a pseudo-code outline of the drawing process, and the following list indicates the functions and their locations within the source code files.

PseudoCode for the Wall Drawing Process**Select Tool**

```

WndProc_OnCommand ( IDM_TOOLS_DRAWWALL ) // User: selects drawing tool
{
  _bWall = true; // tool selection flag
  SPWndProc_OnCtSwitchCursor( 1 ); // display drawing cursor
}

```

Begin Drawing

```

SPWndProc_OnLButtonDown( ) // User: clicks LMB
{
  if( !_bDraw ) _bDraw = true; // set drawing flag
}

```

Drawing

```

SPWndProc_OnKey( left | right | up | down | enter | esc )
SPWndProc_OnMouseMove( x, y )
{
  SendMessage( SketchPad, CT_DRAW, dir, increment );
}

```

Finish Drawing

```

SPWndProc_OnLButtonDown( ) // User: clicks LMB
{
  if( _bDraw )
  {
    if( wall_check() == 0 ) // validate wall drawing
    {
      walls_set(0); // set icons in _Walls[][]
      zones_set(0); // set icons in _Zones[][]
      InvalidateRect( SketchPad ); // send WM_PAINT message
    }
  }
}

```

SketchPad Painting – Display of SketchPad arrays on the screen

```

SPWndProc_OnPaint( ) // SketchPad window WM_PAINT message handler
{
  sketch_redraw()
  {
    for( for each cell of SketchPad )
    {
      sk_draw_symbol()
      {
        grblank(); // set background color of icon
        sk_draw_icon() // get icons from SketchPad arrays
        {
          grputc(icon, color)
          {
            SetTextColor(color); // Windows API function call
            TextOut(icon); // Windows API function call
          }
        }
      }
    }
  }
}

```

TextOut()

This is a Windows Graphics Device Interface (GDI) function that outputs text to a given device context which in this case is the SketchPad window. The fonts for the device (*WALTON01-16.FON*) are initialized at program startup by the *InitFont()* function call from within *SPWndProc_OnCreate()*. The ContamW installation program installs these fonts into the system fonts folder.

Function	File Name
WndProc_OnCommand() IDM_TOOLS_DRAWWALL	WndProc.c
SPWndProc_OnCtSwitchCursor() SPWndProc_OnLButtonDown() SPWndProc_OnKey() SPWndProc_OnMouseMove() SPWndProc_OnCtDraw() SPWndProc_OnPaint()	SPWndPro.c
walls_set() zones_set() sketch_redraw() sk_draw_symbol() sk_draw_icon()	sketch.c
wall_check()	walldraw.c
grblank() grputc() InitFont()	wutils.c
SetTextColor() TextOut()	Windows API

Table – SketchPad Drawing Related Functions

C.5.3.2 Icon Placement

Some building components are drawn onto the SketchPad via the pop-up icon placement menu. This menu contains a list of building components that are represented by specific icons. The menu selections are enabled/disabled based on context-sensitive rules for icon placement.

The menu is activated when the user clicks the right mouse button (RMB) on the SketchPad. This causes the *WM_ONRBUTTONDOWN* message to be sent to the SketchPad window which is handled by the *SPWndProc_OnRButtonDown()* message handler. This in turn calls both *SetPopMenu()* and *TrackPopMenu()* functions. *SetPopMenu()* is a ContamW function that implements the context-sensitive rules of icon placement, and *TrackPopMenu()* is a Windows API function that displays the actual menu wherever the mouse pointer is when the RMB is clicked. When the user selects an item from the icon placement menu a *WM_COMMAND* message is sent to the SketchPad window along with a menu command identifier. The *SPWndProc_OnCommand()* message handler function contains a case for each icon placement menu selection of the form *IDM_POPI_X* where *X* stands for *ZONE*, *AMBIENT*, *PHANTOM*, *PATH*, *AHS*, *INLET*, *OUTLET*, *SS*, *EXPOSURE*, or *NOTE*. Each of these message handlers places the corresponding icon identifier in the appropriate SketchPad array then calls the *sk_draw_symbol()* function to display the new icon on the SketchPad window. The icon will be red in color to indicate that the icon is undefined, i.e., it is not yet associated with a building component. At this point a value indicating an undefined building component will also be placed into the appropriate SketchPad arrays if necessary, e.g., a value of *ZNDF* (-2) indicating an undefined zone will be placed in the *_Paths* array and in every cell of the *_Zones* array that is enclosed by the walls immediately surrounding the zone icon.

C.5.4 Creating and Editing Building Components

Once an icon is placed on the SketchPad a building component must be created and associated with it. This is done via dialog boxes and property sheets associated with the type of building component represented by the particular icon.

When the user double clicks on an undefined icon on the SketchPad the *WM_LBUTTONDOWN* message is sent to the SketchPad window with the *fDoubleClick* parameter set to true. The *SPWndProc_OnLButtonDown()* function checks the SketchPad arrays in a hierarchical fashion to determine the type of icon for which to display properties. Having determined the type of undefined icon, a temporary building component data structure is allocated with default values. This temporary structure is passed as a parameter to the associated component dialog box procedure (a type of window procedure) to be modified as required by the user. If the user selects “OK” on the dialog box, control returns to the *SPWndProc_OnLButtonDown()* message handler, and a permanent copy of the data structure representing the building component is made and added to the corresponding list of building components (see [Building Component and Element Data](#)). At this point a number value is assigned to the building component by ContamW and placed into the appropriate SketchPad array at the proper location.

Editing of existing building components is accomplished in much the same way as creating a new one. The user double clicks on the desired icon and instead of creating a new, default component, a copy is made of the existing component and sent to the appropriate dialog box procedure.

C.5.5 Running Simulations

ContamW is used to develop a set of equations that represents a building as a multizone airflow and contaminant transport network. The GUI provides the means to develop a schematic representation of the building in the level of detail required by the multizone modeling paradigm. The equation solver then takes this schematic representation and converts it into a set of airflow and contaminant transport equations.

The equation solver, *contamx2.exe*, is a stand-alone executable that operates directly on the project file created using the ContamW GUI. Simulation parameters are input by the user via the Simulation Parameters property sheet accessed from the Simulation->Set Simulation Parameters menu command. A simulation is run when the user selects Simulation->Run from the menu which causes the *IDM_SIMULATION_RUN* case of the *WndProc_OnCommand()* message handler function to be executed which calls the function *ContamSim()* located in the source file *SimDlgX2.c*. *ContamSim()* uses the *CreateProcess()* Windows API function to execute *contamx2.exe* in another thread. Complete documentation of the solver is provided in [ContamX Program Documentation](#).

C.5.6 Viewing Simulation Results

The equation solver creates an output file having the same name as the project file but with the *.sim* extension. The GUI can be used to display these results. Results can also be exported to tab-delimited text files for importing into spreadsheet programs for more detailed analysis or written to report files that are organized for legibility. There are several methods of viewing results via the GUI as described below.

C.5.6.1 SketchPad Results

SketchPad results – in the form of color-coded lines to indicate the relative magnitude and direction of pressure drop and airflow rate – can be displayed for each airflow path one level and one time step at a time (See figure below). If a matching *.sim* file is available, a global variable *_resready* is set to true, and the user can select to view/hide the results display and select different time steps of transient results to display via the set of View menu commands which are handled by the *IDM_VIEW_X* cases of the function *WndProc_OnCommand()*. The SketchPad results are read from the *.sim* file by the *age_of_air()* function for the current time step as

maintained in the global variable *_time_index*. The actual display of results is handled within the *SPWndProc_OnPaint()* message handler function which calls *res_dsp_level()* that performs scaling of the lines and calls lower level display functions *grlinevw()* and *grlinehw()* that in turn call Windows GDI functions to perform the actual graphical display. Along with the color-coded lines, the value and direction of the airflows and pressure drop of the individual airflow paths are displayed in the status bar for the currently highlighted airflow path icon on the SketchPad. This is done via the *sketch_status()* function which is called when several events occur including change of highlighted SketchPad cell via keyboard or mouse movement, changing the current level displayed or changing between SketchPad display modes (normal, results, wind and links).

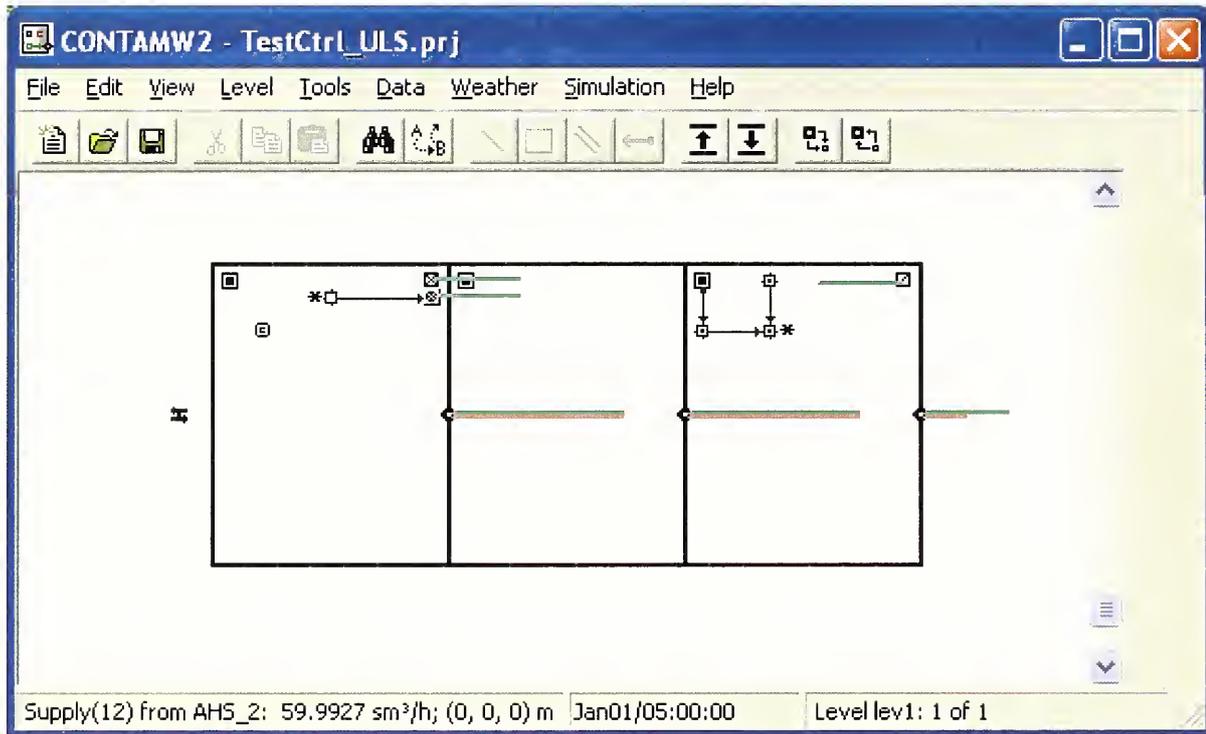


Figure – SketchPad Displaying Simulation Results of Airflow (blue lines) and Pressure Difference (red lines) for Airflow Paths

Function	File Name
WndProc_OnCommand() IDM_VIEW_X	WndProc.c
SPWndProc_OnPaint()	SPWndPro.c
age_of_air()	flows_out.c
res_dsp_level()	resultsw.c
grlinevw() grlinehw()	wutils.c
sketch_status()	bsketch.c

Table –SketchPad Results Display Functions

C.5.6.2 Results Display Window

Users may choose to display/hide a separate results display window (shown in the right side of the figure below). This window displays contaminant concentrations results and net airflow rates between adjacent zones of the currently highlighted zone icon on the SketchPad for steady state simulations and for the current display time step for transient simulations. Occupant related results can also be displayed in this window for highlighted exposure icons. This window is a modeless dialog box for which a global handle *g_hDlgResults* is maintained to which messages are dispatched from the main message loop. The dialog box procedure for this window *SSResultsDlgProc()* is located in the file *SSResDlg.c*. This window is updated when appropriate by *sketch_status()* which sends messages to this window via the *g_hDlgResults* handle.

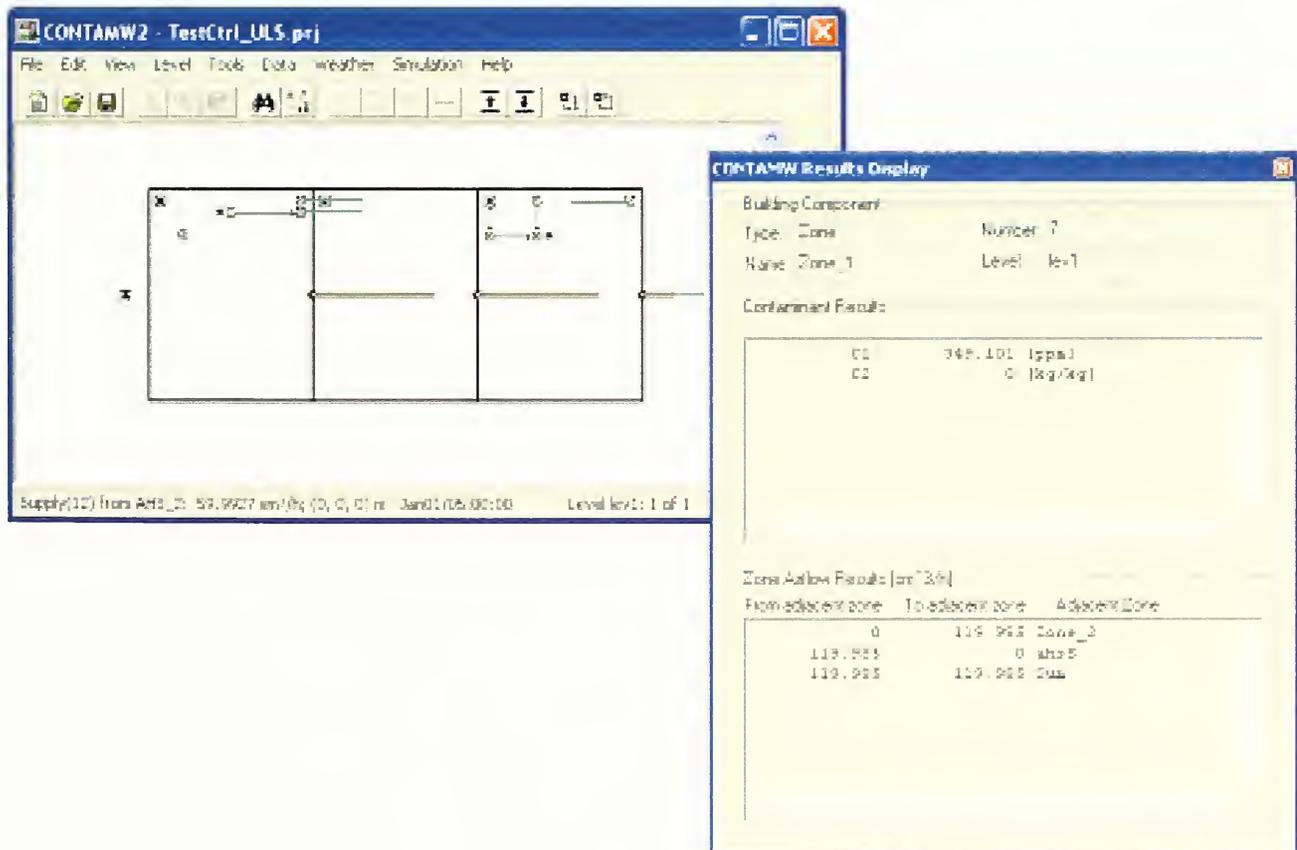


Figure –Results Display Window

C.5.6.3 Result Graphs

Users can use the GUI to display graphs of transient simulation results (see figure below) via the Simulation->Plot... menu commands. These commands are handled by the *IDM_SIMULATION_GRAPH_X* cases of the *WndProc_OnCommand()*, where *X* is either *CONTAM*, *AIRFLOW* or *EXPOSURE*. Each of these cases will in turn create a dialog box having associated dialog box procedures *GraphRsltContamDlgProc()*, *GraphRsltAirflowDlgProc()* and *GraphRsltExposDlgProc()* located in the files *GrphCtm.c*, *GrphFlow.c* and *GrphExp.c* respectively. These procedures and others within these files provide for the user input of chart options, allocated memory for data to be plotted and creation and display of the windows in which the charts are displayed. The charts are created and displayed using third party charting software known as Oletra Chart. Oletra Chart is implemented as a DLL, *olch2d32.dll*, whose import library *olch2d32.lib* is included in the project along with other related header files: *oc_color.h*, *olch2d.h*, *olch2dcm.h*.

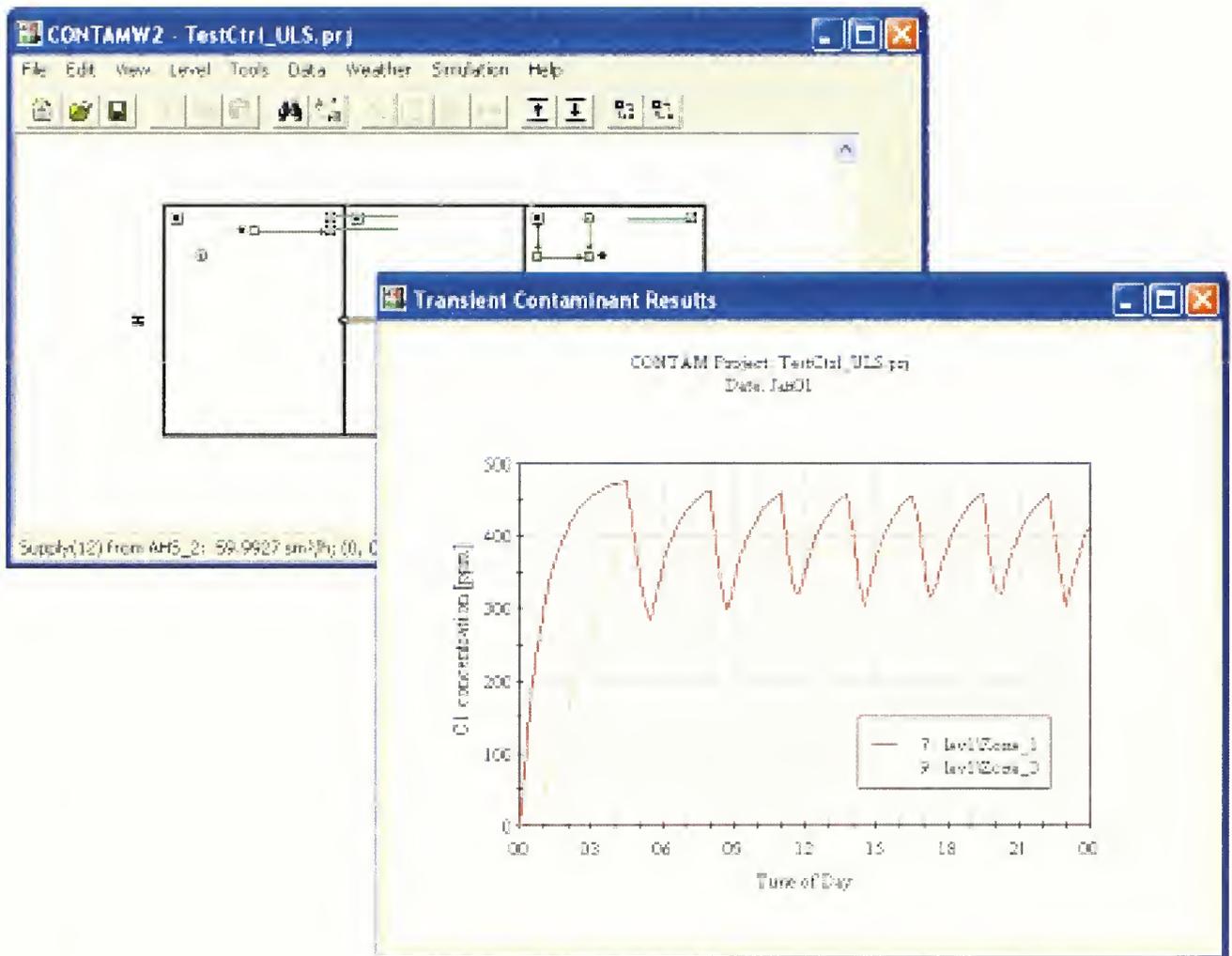
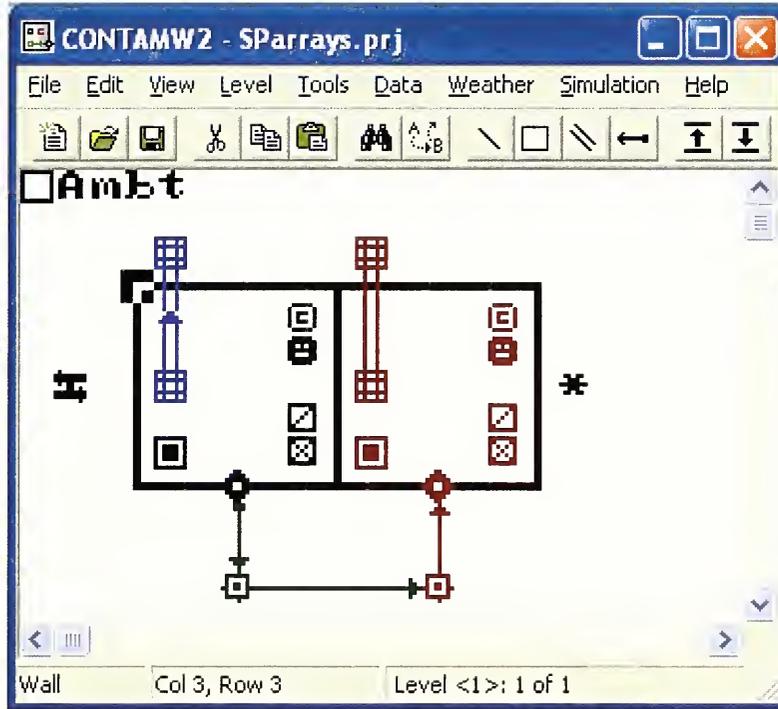


Figure – Transient Contaminant Results Graph

Appendix D SketchPad Arrays

This appendix displays the SketchPad arrays presented in section C.4.1. All of the spreadsheets presented in this appendix refer to the figure below, and each spreadsheet represents part of the arrays in which SketchPad data is stored. The associated project file, *SParrays.prj*, is installed in the samples subdirectory of the CONTAM 2.4 installation directory.



SketchPad used to demonstrate SketchPad array data.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	14	11	11	11	11	11	19	11	11	11	11	11	15	0
4	0	0	0	12	0	0	0	0	0	12	0	0	0	0	0	12	0
5	0	0	0	12	0	0	0	0	0	12	0	0	0	0	0	12	0
6	0	0	0	12	0	0	0	0	0	12	0	0	0	0	0	12	0
7	0	0	0	12	0	0	0	0	0	12	0	0	0	0	0	12	0
8	0	0	0	12	0	0	0	0	0	12	0	0	0	0	0	12	0
9	0	0	0	17	11	11	11	11	11	21	11	11	11	11	11	16	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

_Walls[][] array of SParrays.prj

APPENDIX D – SKETCHPAD ARRAYS

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	7	65	109	98	116	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	133	0	0	0	0	0	133	0	0
5	0	0	0	0	0	0	0	0	131	0	0	0	0	0	131	0	0
6	0	130	0	0	0	0	0	0	0	0	0	0	0	0	0	0	42
7	0	0	0	0	0	0	0	0	129	0	0	0	0	0	129	0	0
8	0	0	0	0	5	0	0	0	128	0	5	0	0	0	128	0	0
9	0	0	0	0	0	0	23	0	0	0	0	0	23	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

_Sketch[][] array of SParrays.prj

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a
1	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a
2	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a
3	a	a	a	W	W	W	W	W	W	W	W	W	W	W	W	W	a
4	a	a	a	W	1	1	1	1	1	W	-2	-2	-2	-2	-2	W	a
5	a	a	a	W	1	1	1	1	1	W	-2	-2	-2	-2	-2	W	a
6	a	a	a	W	1	1	1	1	1	W	-2	-2	-2	-2	-2	W	a
7	a	a	a	W	1	1	1	1	1	W	-2	-2	-2	-2	-2	W	a
8	a	a	a	W	1	1	1	1	1	W	-2	-2	-2	-2	-2	W	a
9	a	a	a	W	W	W	W	W	W	W	W	W	W	W	W	W	a
10	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a
11	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a
12	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a
13	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a
14	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a	a

a = AMBT = -1
 ZNDF = -2 (undefined zone)
 W = WALL = -3

_Zones[][] array of SParrays.prj

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0
5	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0
6	0	1	0	0	2	0	0	0	0	0	0	0	0	0	0	0	1
7	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
8	0	0	0	0	1	0	0	0	2	0	-2	0	0	0	0	0	0
9	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

_Paths[][] array of SParrays.prj

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	162	0	0	0	0	0	162	0	0	0	0	0	0
3	0	0	0	0	146	0	0	0	0	0	144	0	0	0	0	0	0
4	0	0	0	0	154	0	0	0	0	0	144	0	0	0	0	0	0
5	0	0	0	0	144	0	0	0	0	0	144	0	0	0	0	0	0
6	0	0	0	0	162	0	0	0	0	0	162	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

_Ducts[][] array of SParrays.prj

APPENDIX D – SKETCHPAD ARRAYS

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	2	2	2	2	2	2	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

_Ctrls[][] array of SParrays.prj

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	181	0	0	0	0	0	177	0	0	0	0
11	0	0	0	0	0	0	179	0	0	0	0	0	169	0	0	0	0
12	0	0	0	0	0	0	185	170	168	168	168	178	185	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

_Links[][] array of SParrays.prj

