ERDC/CERL TR-04-22

US Army Corps of Engineers®
Engineer Research and Development Center

Facility Composer Design Wizards

A Method for Extensible Codified Design Logic Based on Explicit Facility Criteria

Van J. Woods, Susan D. Nachtigall,
Beth A. Brucker, and Awilda Andrillion

November 2004

Construction Engineering Research Laboratory

ERDC/CERL TR-04-22
November 2004

Facility Composer Design Wizards: A Method for Extensible Codified Design Logic Based on Explicit Facility Criteria

Van J. Woods, Susan D. Nachtigall, Beth A. Brucker, and Awilda Andrillion

Construction Engineering Research Laboratory
PO Box 9005
Champaign, IL 61826-9005

Final Report

Prepared for U.S. Army Corps of Engineers
Washington, DC 20314-1000

Under Work Unit LK6K75-N, “Fort Future Facilities”

ABSTRACT: Government design criteria are commonly captured in the form of design guides, regulations, technical manuals, and web pages, but not in a computable format. Current design systems provide no way to directly interact with a specific criterion, or to efficiently extend the functionality of an application to directly support criteria usage. Consequently, the only two choices are that designers must either manually ensure that all applicable criteria are identified and satisfied, or that a large customized application is developed. Custom systems are slow to develop and change, and difficult to update. Such systems do allow data modularization, but do not provide modular functionality—the ability to support customized methods or algorithms that perform useful operations on the data. The *Facility Composer* suite of tools supports the capturing and tracking of facility criteria and requirements, planning and design charrettes, and associated planning and design analyses. *Facility Composer* addresses many of the problems associated with the decentralized, non-computationally explicit, ad-hoc definition, distribution, and utilization of design criteria. This report describes work undertaken to provide a set of the most commonly used tools as part of the core features in *Facility Composer*, and also to provide a means for modularized extensibility of design logic.

Contents

List of Figures and Tables

Figures

Tables

Conversion Factors

Non-SI* units of measurement used in this report can be converted to SI units as follows:

Multiply	By	To Obtain
acres	4,046.873	square meters
cubic feet	0.02831685	cubic meters
cubic inches	0.00001638706	cubic meters
degrees (angle)	0.01745329	radians
degrees Fahrenheit	(5/9) x (°F – 32)	degrees Celsius
degrees Fahrenheit	(5/9) x (°F – 32) + 273.15.	kelvins
feet	0.3048	meters
gallons (U.S. liquid)	0.003785412	cubic meters
horsepower (550 ft-lb force per second)	745.6999	watts
inches	0.0254	meters
kips per square foot	47.88026	kilopascals
kips per square inch	6.894757	megapascals
miles (U.S. statute)	1.609347	kilometers
pounds (force)	4.448222	newtons
pounds (force) per square inch	0.006894757	megapascals
pounds (mass)	0.4535924	kilograms
square feet	0.09290304	square meters
square miles	2,589,998	square meters
tons (force)	8,896.443	newtons
tons (2,000 pounds, mass)	907.1847	kilograms
yards	0.9144	meters

* *Système International d'Unités* ("International System of Measurement"), commonly known as the "metric system."

Preface

This study was conducted for Headquarters, U.S. Army Corps of Engineers (HQUSACE) under Project "Fort Future," Work Unit LK6K75-N, "Fort Future Facilities." The technical monitor was Michael P. Case, CEERD-CF-N, Special Projects Officer for Fort Future.

The work was performed by the Engineering Processes Branch (CF-N), of the Facilities Division (CF), Construction Engineering Research Laboratory (CERL). The CERL Principal Investigators were Beth Brucker, Susan Nachtigall, and Van Woods. Special recognition is given to William Zwicky and Awilda Andrillion for their help in developing the wizard examples contained within this report. Also special recognition is given to David Bailey for the development of the Design Logic Tree for the DoD Minimum Antiterrorism Standards for Buildings Wizard. The technical editor was William J. Wolfe, Information Technology Laboratory. Donald K. Hicks is Chief, CECER-CF-N, and L. Michael Golish is Operations Chief, CECER-CF. The associated Technical Director was Paul A. Howdyshell, CEERD-CV-ZT. The Director of CERL is Alan W. Moore.

CERL is an element of the U.S. Army Engineer Research and Development Center (ERDC), U.S. Army Corps of Engineers. The Commander of ERDC COL James R. Rowan, and the Director of ERDC is Dr. James R. Houston.

1 Introduction

Background

Government design criteria are contained in many volumes of design guides, regulations, technical manuals, and web pages, but few (if any) are expressed in a computable format. Moreover, current design systems provide no way to directly interact with a specific criterion, or to efficiently extend the functionality of an application to directly support criteria usage according to locally acceptable practices. Consequently, designers must either manually ensure that all applicable criteria are identified and satisfied, or a large customized application is developed to assist. Custom systems that support this process are costly, slow to develop and change, and difficult to update with the most current criteria data and design processes. Such systems do allow data modularization (which allows offices to have their own libraries of data), but do not provide modular functionality—the ability to support customized methods or algorithms that perform useful operations on the data.

The *Facility Composer* suite of tools supports the definition, use, and tracking of facility criteria and requirements during planning and design charrettes. *Facility Composer* addresses many of the problems associated with the commonly decentralized, non-computationally explicit, ad-hoc definition, distribution, and utilization of design criteria. While customers expressed their appreciation of these capabilities, they also described the need for more flexibility, as their design practices commonly varied by:

1. **Regional differences**, which are influenced by differences in building codes, weather, local available labor skills and materials, topography, and social and historical conventions.
2. **Organization-specific practices**, which reflect differences in company-specific institutionalized approaches to solving problems, company/client specific mission priorities (for example, State Department vs. Army Reserve and National Guard), and historical corporate knowledge.
3. **Facility type**, which is one of the most significant reasons for the need for specialized processes because the function of a particular facility type demands design approaches and algorithms that address their requirements in a specific way.

These factors can change the priority of certain criteria, or require that certain (possibly identical) criteria be treated differently. For example, the criteria for sustainable design may be defined to be generally applicable, whereas the specific design strategies (e.g., the decision to use straw bale construction) will likely be significantly influenced by specific region and facility type.

Objectives

The primary objective of this work was to implement an extensibility mechanism within the *Facility Composer* system capable of allowing for effective customization, maintenance, reuse, and incremental delivery of computable design logic.

Approach

The efforts of this work resulted in the creation of the concept of a design "Wizard." As a result, a software framework for creating wizards was developed as well as a diverse set of example wizards, which are now available as part of the *Facility Composer* system. This report includes examples and discussions illustrating the concepts and implementation issues surrounding the use and development of wizards.

Mode of Technology Transfer

It is planned that these technology concepts will be integrated into the current and future development of *Facility Composer*. More specifically, this technology will be targeted for development currently underway as part of the Totally Integrated Project Delivery (TIPD) program.

This report will be made accessible through the World Wide Web (WWW) at URL:

http://www.cecer.army.mil

2 *Facility Composer* "Wizard" Concept

Definition

"Wizards" are software components that operate on a discrete design task by taking criteria and user input in order to create or manipulate a building and criteria model rapidly, according to recognized practices. A Wizard is defined as:

> A module of software that represents a discreet design task within a particular context, typically characterized by a sequential series of questions and options from which codified design logic and criteria are used to create or modify a solution.

A Wizard is programmed to use the criteria data expressed in the *Facility Composer* system to create or analyze something in a useful way. For example, a simple wizard might determine the number of faucets required for a restroom within a certain building type with a particular building occupancy level, based on standard design criteria tables. This helps the designer ensure that the design solution meets design guide requirements, and that the customer's requirements are satisfied.

Wizard Types

Wizards are not limited to any particular phase of the design process; they are envisioned to be applicable from preliminary design, to design development, and to detailed design, and from generation, to analysis and review. Figure 1 shows the relationship between some Wizards that have already been prototyped or conceptualized, and the rest of the applications in the *Facility Composer* suite (specifically, *Planning Composer*, and *Layout Composer*). The different categories of Wizards envisioned include, but are not limited to: Criteria Wizards, Model Generation Wizards, and Analysis Wizards.

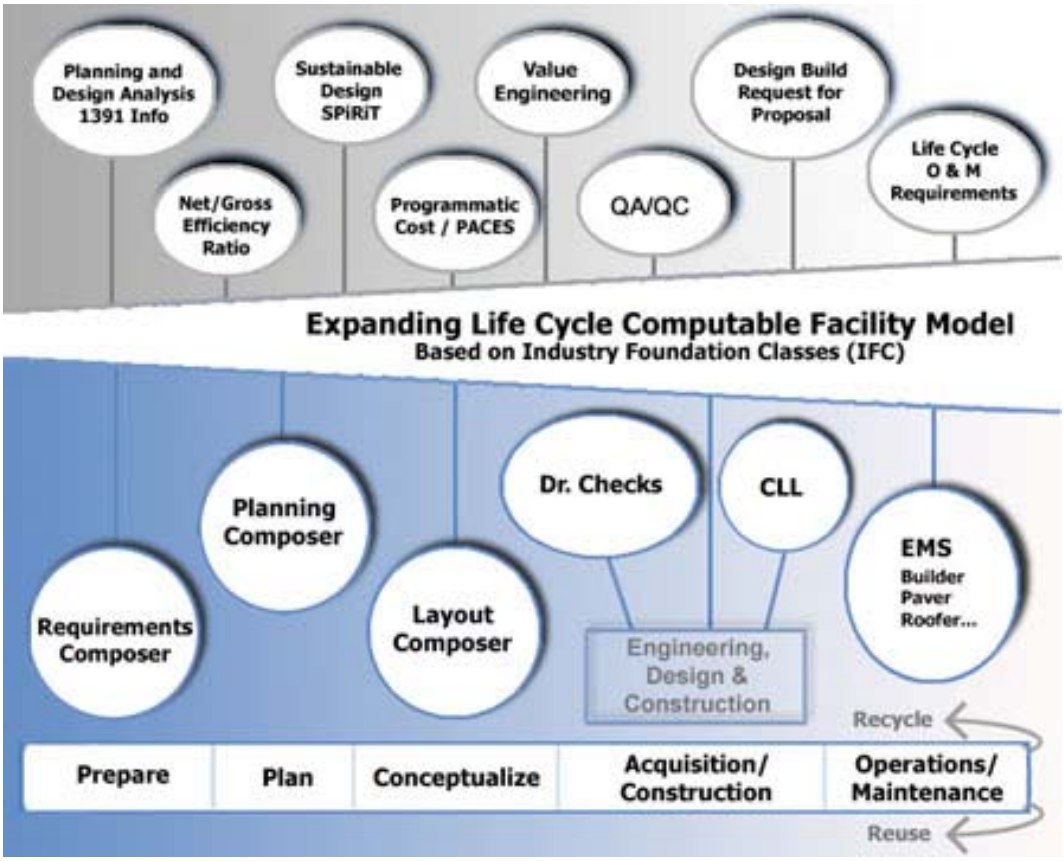


Figure 1. ***Facility Composer*** **diagram.**

Criteria Wizard

Criteria Wizards are wizards that assist a user primarily in Planning Composer by providing one or more worksheets consisting of questions and answers, selection options, and structured data entry from which an algorithm or calculation is performed to arrive at a value for a particular criterion. For example, in the Parking Allowance and Site Area Calculation Wizard (described in Chapter 4), Planning Composer demands criteria such as the number of parking stalls, handicap-accessible parking stalls, and site area while the wizard provides the means to determine the values for these criteria based on user input. Other Criteria Wizards currently under development include the Plumbing Fixture Calculation Wizard and the Sustainable Designer's Aid Wizard.

Model Generation Wizard

Model Generation Wizards are wizards that interact with commercial computer-aided design (CAD) software to generate model components and object configurations through parametric modeling formulas or manual specification. These wizards can rapidly generate configurations from which iterative refinements could

occur. Examples of these would be a Duct Layout Wizard based on supply and exhaust airflow, or a Lighting/Ceiling Grid Layout based on grid spacing, diffuser layouts, and lighting algorithms and requirements (foot-candle, lumens).

Analysis Wizard

Analysis Wizards interact with third-party analysis tools in addition to custom analysis tools written within *Facility Composer*. Examples of third-party tools might include: energy analysis, security analysis, and force protection analysis. Analysis Wizards currently being considered for *Facility Composer* are Net To Gross Area Calculation and Preliminary Egress Analysis.

One item to note is that the system should, whenever possible, suggest answers from which professional judgment and expertise should be able to choose to override. Rather than letting the computer have the last say, we recommend an approach of enforcing accountability that is supported by recording when criteria modifications are made, by storing the user name, Wizard version, time and date stamp of modification time, default value, and overridden value. Criteria analysis should be provided to easily compare the specified desired criteria values against the values the designer actually chose. This approach respects the fact that while a wizard can efficiently support common practices, a design professional's expert judgment is still required and is the ultimate final determinant.

Future Wizards

The following list of ideas for wizards (which we may consider implementing based on customer feedback) suggests the type of capabilities that may be incorporated into a wizard:

- Design-Build Request for Proposal (RFP) Wizard
- Preliminary Design Building Code Analysis Wizard
- Project Engineering Report (PER)
- DD1391 Planning and Design Reporting Wizard
- Net / Gross Area Wizard
- New Project Wizard (Project Templates)
- Lighting Grid & HVAC diffuser layout
- Electrical Outlet Wizard
- Criteria Conflict Analysis and Resolution Analysis Wizard
- Create New Building Wizard (specify number of stories desired and standard floor-to-floor height)
- Spatial Conflict Analysis Wizard
- Energy Analysis Wizard.

Software Architecture

From a software architecture and design perspective, adding modular functionality to *Facility Composer* presents other desirable characteristics. Modular functionality allows for incremental development, which means as customers decide they would stand to gain from the development of a particular Wizard that helps them codify some part of a design process, we will address those requirements and the software development at that point. This will ensure a process that provides greater responsiveness to designer's needs and that is flexible enough to evolve over time. This is consistent with the currently popularized *agile* software development methodology (Cockburn 2002, Highsmith 2002).

From a software perspective, there is a strong parallel between the concept of a Wizard and the notion of object-oriented (OO) software principles. OO is based on the premise of recognizing the intrinsic coupled nature of data and the legal operations that are allowed to operate on that data. In OO lingo, Wizards "encapsulate" the allowable "methods" of setting and modifying "attributes" of criteria data. For instance, Planning Composer requests specific criteria that in turn, can be modified through the collection of methods within the corresponding Wizard (Figure 2).

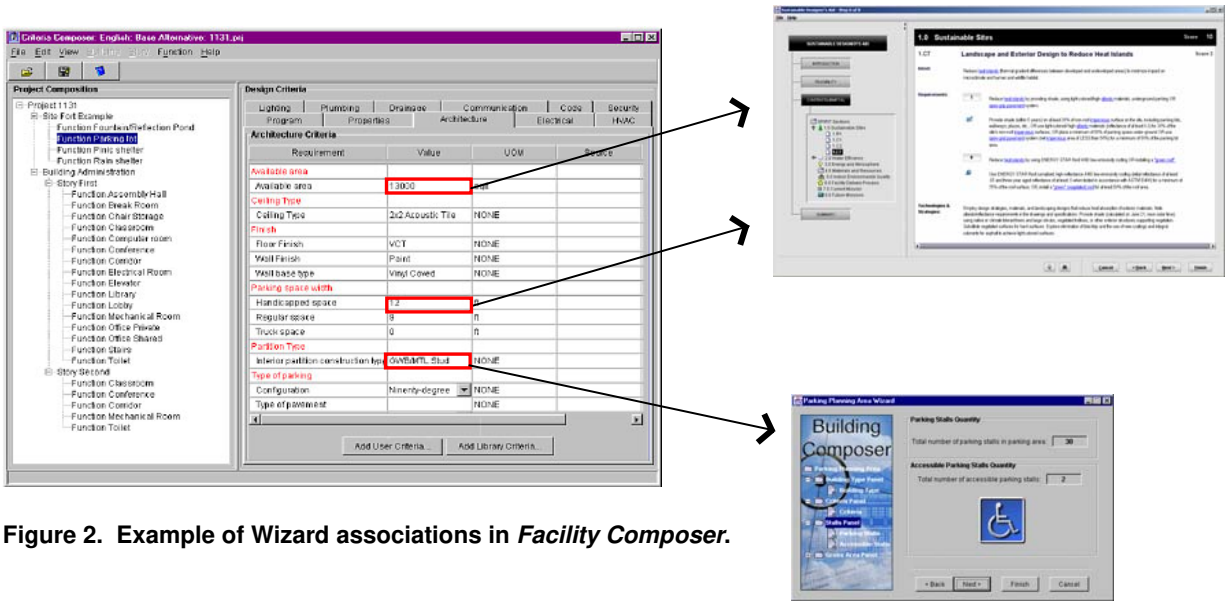


Figure 2. Example of Wizard associations in *Facility Composer*.

3 General Implementation Principles

Certain implementation issues should be considered when developing a new Wizard. In addition, *Facility Composer* offers certain capabilities that need to be factored into the planning process. Finally, this chapter offers some suggestions to aid in the conceptual process of designing a wizard.

Inductive User Interface

As it relates to a software user interface, a wizard is simply a series of guided steps. This is similar in concept to the *inductive user interface* (IUI) approach. The IUI approach presents the user with multiple screens, each of which focuses on the information and steps required to complete a specific task. This contrasts to the more common approach of presenting a single main interface for an application from which all subsequent features are accessible, but must be determined (or deduced) by the user.

When the conceptual model of an application is explicitly presented, the user is far more likely to achieve the desired objective while utilizing all of the capabilities of the program. This is, again, in contrast to traditional applications in which the users typically understand only a small number of the features. In other words, the IUI approach could be characterized as "process centric" rather than "feature centric."

Implementation of an IUI includes designing a process composed of a series of screens. Each individual screen focuses on a single, clearly stated task, and the contents of each page properly suit the task at hand. The design of an IUI based application should begin with the traditional use case analysis and feature specification (Leffingwell 2003). Note that the interface design phase must consider many screens instead of one primary screen. While this process may appear to require more work, it is actually easier than the single-menu approach. Each use case (or collection of a limited set of similarly related use cases) relates to one screen, which is much simpler than trying to design one screen to accomplish all objectives. Some items for consideration are:

- Focus each screen on a single primary task. Certain screens may exist to provide a list of tasks, whereas other screens will contain steps to achieve a given task.
- State the task on each screen by providing a specific title that induces an action (is not passive). Even novice users should be able to easily determine: (1) what can be accomplished, (2) where to go to achieve it, and (3) where to go next.
- Make the screen's content suit the task.
- Offer links to secondary tasks necessary to accomplish the primary task.
- Use consistent screen templates.
- Make it obvious how to carry out the task with the controls on the screen by making sure the sequence and instructions are clearly documented on the screen.
- Make the navigational process obvious. Provide a clear way to complete the task and to start a new one.
- Where possible, allow the user to return to a particular step without requiring them to start from the beginning. It is useful to use non-linear sequences and independent variables that let the user leave a particular step before finishing all the items, and to return at a later time without any data loss.

One example of an IUI implementation can be seen in Microsoft Money 2000. For instance, the prior version grouped the tasks of selecting, creating, and deleting accounts into one screen titled "Account Manager" (Figure 3). It was necessary for the user to first deduce the task at hand, in order to perform the necessary action.

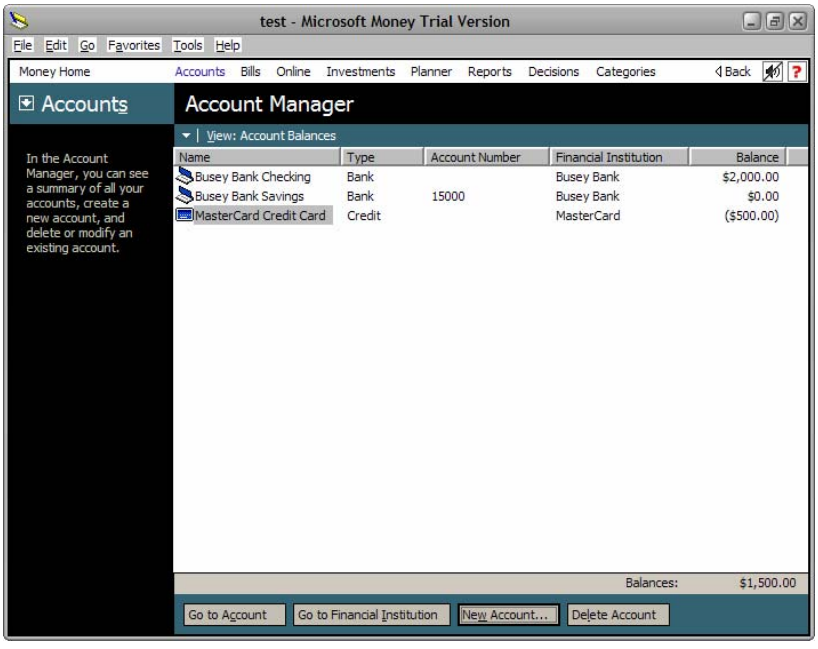


Figure 3. The Money 2000 Account Manager.

The 2004 version (Figure 4) isolated the common task of selecting an account into one individual screen, titled "Pick an account to use." The infrequent tasks of creating and deleting an account were relegated to another individual screen, titled "Set up your accounts in Money" (Figure 5).

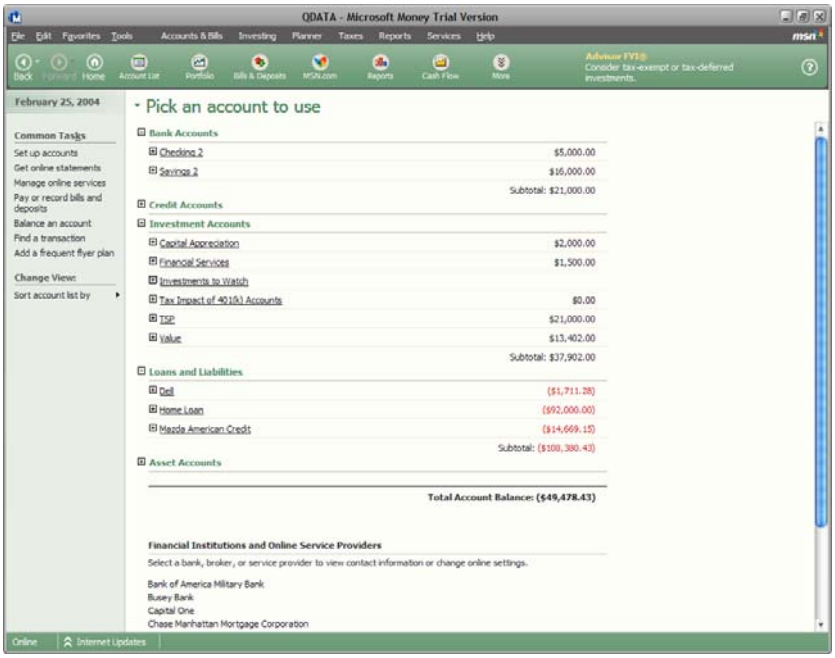


Figure 4. Money 2004 Pick Account screen.

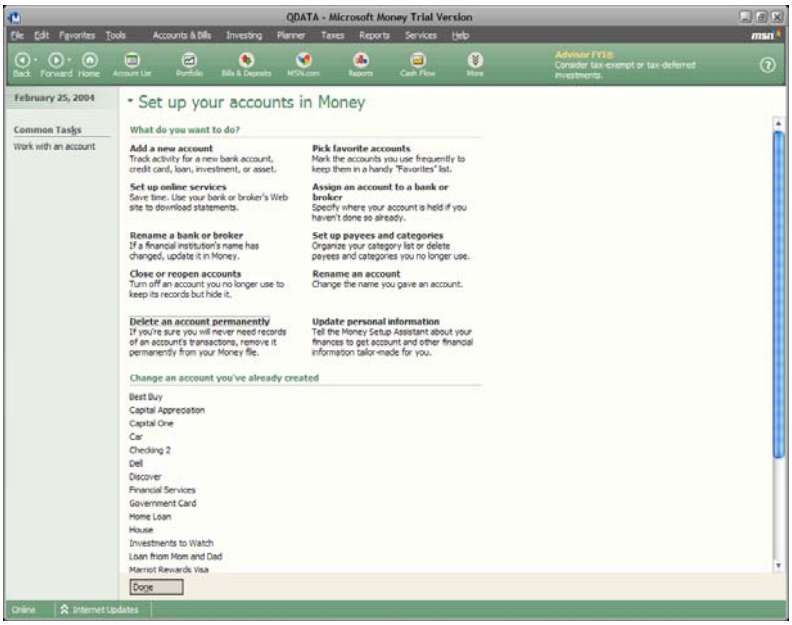


Figure 5. Money 2004 Set up accounts screen.

Appendix A gives more examples of IUI implementations.

General Considerations

Facility Composer supports the following built in data types: Whole number (Integer), Decimal (Double), Text (String), True/False (Boolean), List (Array), and a Custom complex data type (Object).

A wizard can be associated with one or more *Facility Composer* criteria at a time. When any of the criteria values are selected in Planning Composer, the appropriate associated wizard is started and the correct screen is activated. (For example, if the user had already set the selected criteria through the wizard sequence, then clicking on that value again would go directly to the specific value on that page rather than to the initial step of the sequence).

It is best to not change values of criteria that are not included explicitly in the wizard, even those only displayed as a non-editable. In other words, limit "hidden" or unexpected changes. Any time a user's decision results in a changed value, the change propagation should obvious to the user.

A wizard must be associated to a criteria library explicitly to ensure that the necessary criteria are present in the library. Therefore, a wizard has to publish what criteria it needs and what criteria values it outputs.

It is important to avoid duplication of criteria. Rather than inventing a new criteria type for building occupancy or planned area, it makes more sense to utilize the existing definitions.

Wizards that import and export data to perform application integration essentially require the process of database schema mapping. These essentially boil down to relationships of: (1) one to one, (2) one to many, or (3) many to one. Any of these links can be characterized as associations with identical semantic definitions, or associations that require some form of translation (formula/algorithm/heuristics, or user interaction).

Derived values are preferred over "magic numbers" (numbers with an implicit rationale).

It is common that separate design practices may calculate a particular criterion the same way, but with different values (data). Hence, it is important to follow the software principle of strict separation of code and data. That way, the same wizard can be reused in many different contexts if applicable, while simply switching in the appropriate database. (The Parking Allowance and Site Area Calculation Wizard is a good example of this.)

For complicated logic, it might make sense to have a domain expert outline the process using a flow diagram.

Once a sufficient level of understanding has been gained regarding the data and process required to perform the necessary calculations, a prototype interface can be mocked up using Visual Basic for Applications (VBA). This can be completed by non-programmers utilizing this built in feature in applications such as in any of the ones in Microsoft Office. To initiate VBA in Word, for example, select from the menu Tools>Macro>Visual Basic Editor, and then right click in the Project Explorer and select Insert>UserForm. This allows you to simply "drag and drop" all of the typical interface components onto a dialog box in order to test out your ideas (Figure 6).

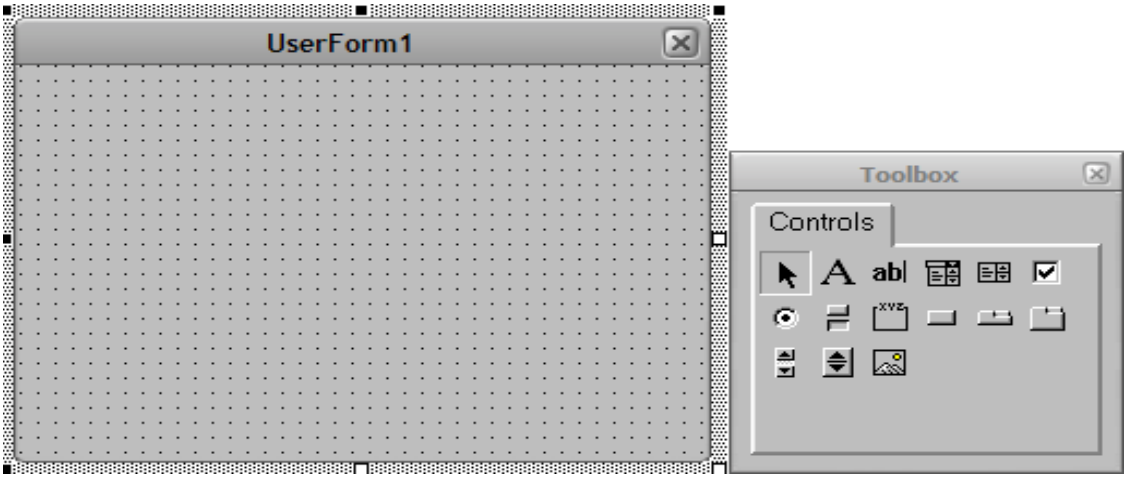


Figure 6. User Form dialog in Word.

Consider Human-Computer Interface Design patterns as a basis for developing possible solutions (Tidwell 2003).

Recall the "SMART" acronym for criteria definition, which suggests that a criteria be: Specific, Measurable, Attainable, Relevant, and Trackable.

4 Detailed Criteria Wizard Example — Parking Allowance and Site Area Calculation Wizard

The next few chapters will illustrate some concrete examples of different types of wizards that have been implemented in order to describe how they are used, but also to illustrate how other, similar, wizards could be implemented. The Parking Allowance and Site Area Calculation Wizard is designed to guide the user through a number of steps that sequentially asks the user for the building type and occupancy level to determine the total number of parking stalls, handicap accessible parking stalls, and site area required in a parking facility.

Design Criteria Data

The requirements for the number of parking stalls required in a parking facility differ based on the building type, customer and region. Table 1 contains the criteria data used for the Parking Allowance and Site Area Calculation Wizard.

Table 1. Authorized Parking Stall Quantities by Facility Type (TI 800-01, table 3-5).

Building Type Category	Criterion One	%	Criterion Two	%
Administration, Headquarters, and Office Buildings	Assigned Personnel	60		
Bank and Credit Unions (when not included in a Community Shopping Center)	Civilian employees; largest shift	38		
Cafeteria, Civilian (when not included in a Community Shopping Center)	Seating capacity	15		
Central Food Preparation Facilities	Military and civilian food service operating personnel; largest shift.	38		
Chapels	Seating capacity	15		
Child Development Centers	Staff	100	Children	25

Building Type Category	Criterion One	%	Criterion Two	%
Community Shopping Centers (may include the following functions: Bank, Commissary Store, Food Sales, Main Exchange, Miscellaneous Shops, Post Office, Restaurant, and Theater.)	Authorized customers served	04	Other criteria provided by the Defense Commissary Agency (DeCA) and Army and Air Force Exchange Service (AAFES).	
Enlisted Personnel Dining Facilities for the following: Permanent party; Garrison (to include both TOE and TDA units); Support Units; Construction Battalions, Weapon Plants; Personnel Transfer and Overseas Processing Centers.	Military and civilian food service operating personnel; largest shift	38	Enlisted personnel (patron parking) to be served during a meal period	08
Family Housing	Living Units	200		
Field House (combined with Football and Baseball Facilities)	Military strength	01		
Fire Stations, One-Company	7 stalls			
Fire Stations, Two-Company	10 stalls			
Guard Houses; Military Police Station	Guard and Staff strength	30		
Gymnasiums (when only 1 on the installation)	Military strength served	01		
Gymnasiums, Area (Regimental)	10 stalls			
Laundries and Dry Cleaning Plants	Civilian employees; largest shift.	38		
Libraries, Central	One stall for each 47 m^2 (500 sq ft) gross floor area.			
Libraries, Branch.	8 stalls			
Maintenance Shops	Assigned personnel; largest shift.	38		
Schools, dependent; without auditorium.	Number of classrooms	200		
Schools, dependent; with auditorium.	Number of classrooms	200	Auditorium seating capacity	15
Security Offices for Main Gates Population 100 – 2,000	5 stalls			
Security Offices for Main Gates Population 2,001 to 4,000	10 stalls			
Security Offices for Main Gates Population 4,001 to 6,000	15 stalls			
Security Offices for Main Gates Population 6,001 to 10,000	20 stalls			
Security Offices for Main Gates Population 10,001 and over.	To be based on a site traffic impact study.			
Service Clubs	Enlisted personnel or officer strength served	02		

Building Type Category	Criterion One	%	Criterion Two	%
Swimming Pools	Design capacity of the swimming pool	20		
Temporary Lodging Facilities	Number of bedrooms	100		
Theaters (when not included in a Community Shopping Center)	Seating capacity	25		
Unaccompanied Enlisted Personnel Housing	Maximum utilization (the minimum requirement).	70		
Unaccompanied Office Personnel Housing	Living Suites	100		
Warehouses	1 stall for each 46.5 m^2 (500 sq ft) gross floor area.		Personnel assigned to storage activity	25

For instance, if the facility is a Chapel, an example of the algorithm performed within the application would be 15 percent of the seating capacity would equal the minimum requirement for the number of parking stalls. Based on the number of parking stalls, the minimum requirement of accessible parking stalls is calculated using the criteria shown in Table 2.

Table 2. Minimum requirements for accessible parking stalls from ADAAG.

Total Number of Parking Stalls	Minimum Number of Accessible Stalls
1-25	1
26-50	2
51-75	3
76-100	4
101-150	5
151-200	6
201-300	7
301-400	8
401-500	9
501-100	2 percent of total
1001 and over	20 plus 1 for each 100 over 1000

Finally, after determining the number of parking stalls, an approximation of the site area is calculated based on the engineering rule of thumb, 400 sq ft (or 37.16 m^2) per parking stall.

Sequence

To abide by the guidelines stipulated by an Inductive User Interface (IUI), the overall process was broken down into a number of tasks in which each task is

presented on a separate page. The Parking Allowance and Site Area Calculation Wizard is composed of the following panels:

1. Introduction Panel

The first panel is the Introduction Panel (Figure 7). The panel briefly outlines the functions of the wizard and also provides the user with a check box to select if they do not want this panel to be displayed the next time the wizard is launched. The main purpose of this screen is to give the user a mental model of what to expect to have to do.

Figure 7. Introduction panel of parking allowance and site area calculation wizard.

2. Building Type Panel

Second is the Building Type Panel (Figure 8), which presents the user a list of building types to choose from. The primary task presented in this screen is for the user to select a building/facility type.

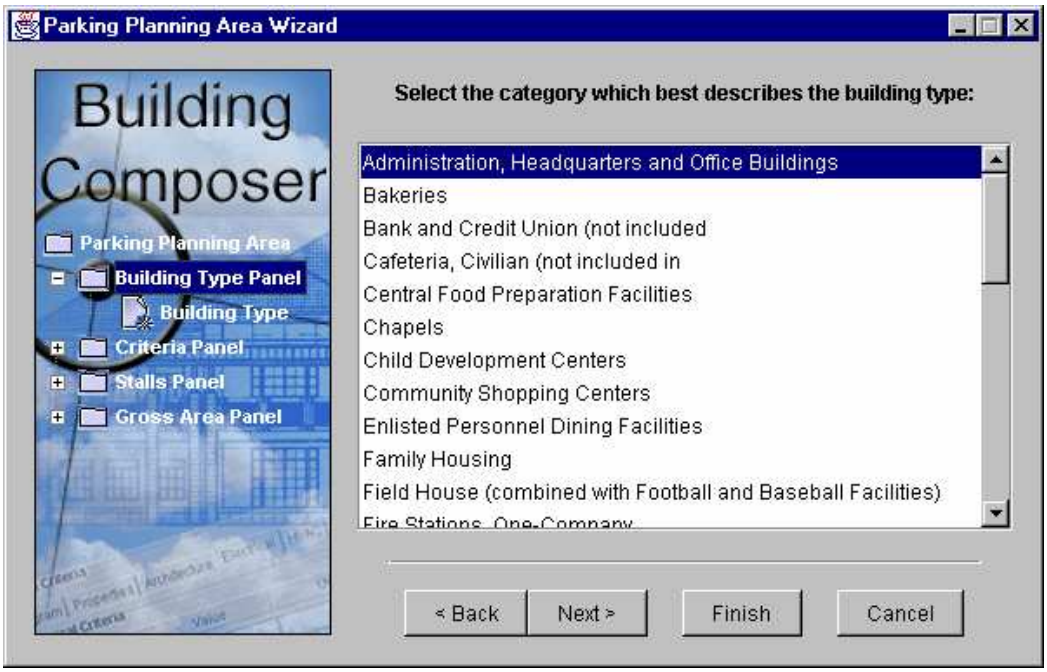


Figure 8. Building type panel.

3. Criteria Panel

The following panel displayed in the sequence of panels is the Criteria Panel (Figure 9). This panel presents the required criteria specific to the building type chosen in the previous panel. The user is expected to enter the values requested in the numeric text fields and press the **Compute** button to obtain the number of number of parking stalls required for the facility in the numeric text field found on the bottom of the screen. The user may also type in a value manually for the minimum parking stalls if the user is not satisfied by the value determined by the application's algorithm.

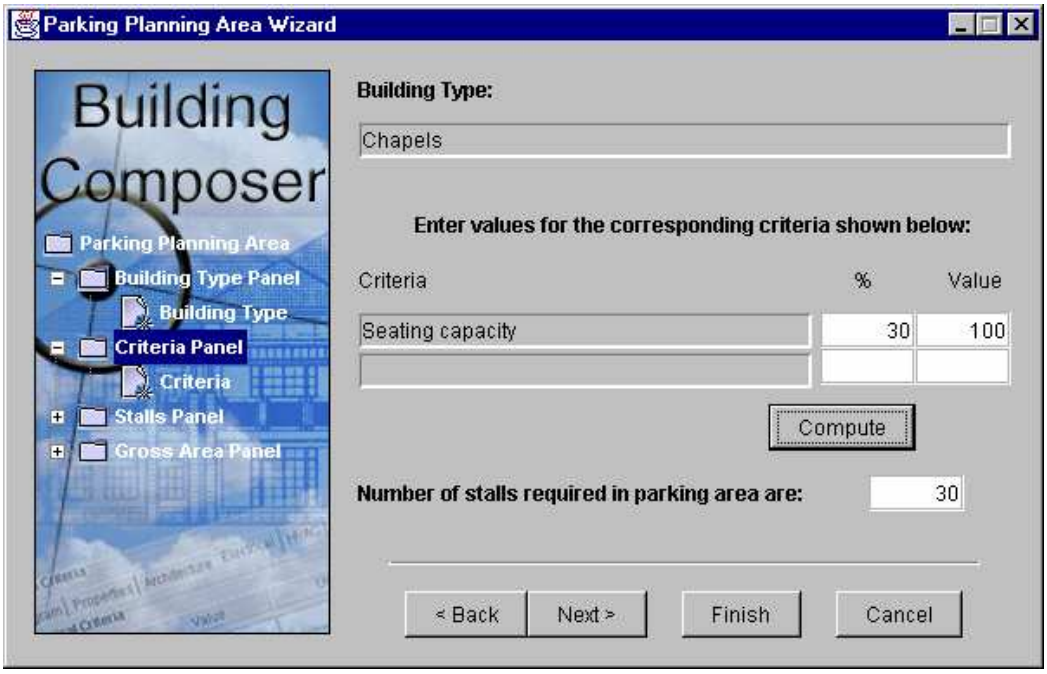


Figure 9. Criteria panel.

4. Stalls Panel

The next panel is the Stalls Panel (Figure 10). This panel simply displays the value determined or manually entered for the number of parking stalls on the previous panel. Based on that number, the minimum requirement for accessible parking stalls is also calculated and displayed.

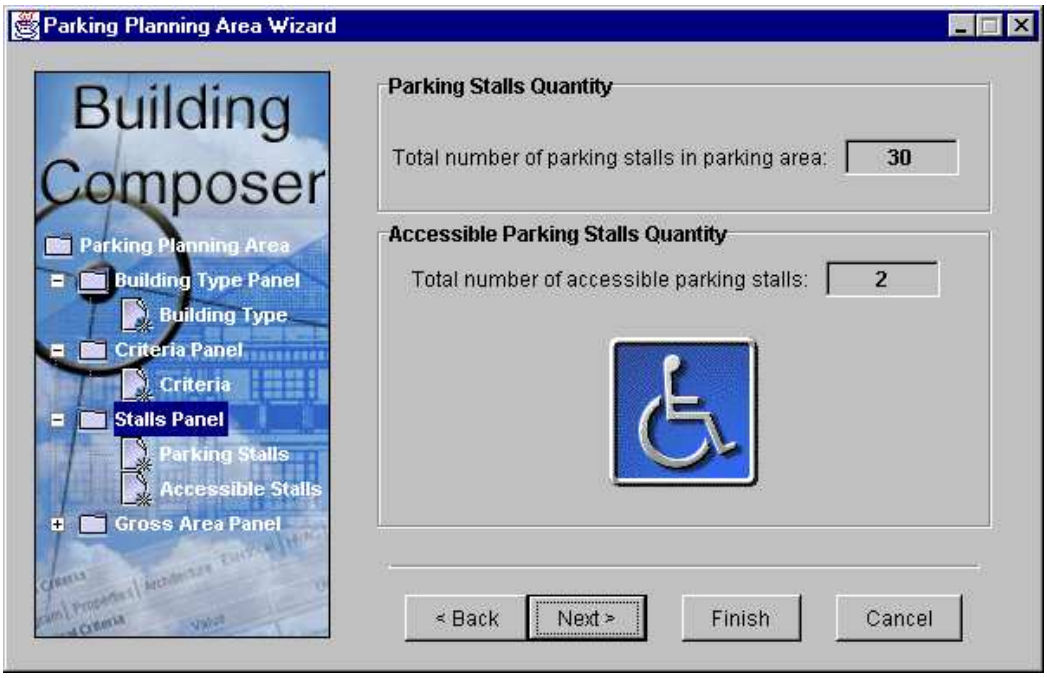


Figure 10. Stalls panel.

5. Gross Area Calculation Panel

Finally, the last step of the application is to calculate the estimated area required for the parking facility. The Area Panel displays the general engineering estimate also known as the engineering rule of thumb, which is editable by the user, used to calculate the total gross parking area based on the number of parking stalls (Figure 11). The user may alter the estimate and press the **Compute** Button, whereby the application utilizes the new value to perform the calculation.

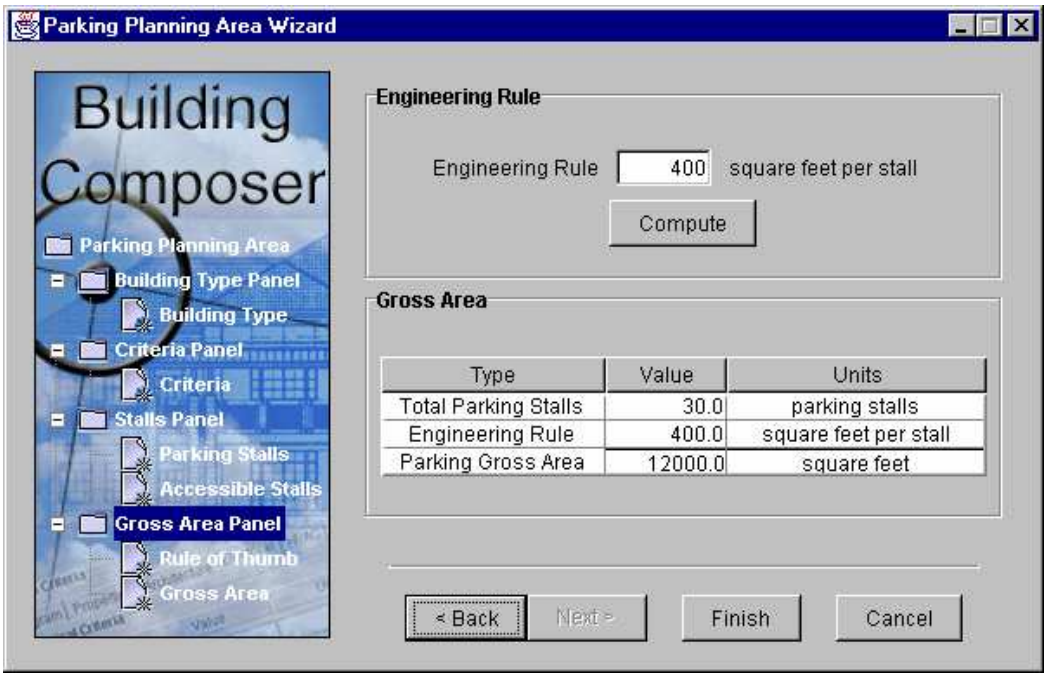


Figure 11. Gross area panel.

Implementation Issues

Template

As one may have noticed from the screen captures shown above, a wizard has certain components that are specific to each page. However, several components span across the entire application, for example, the **Back, Next, Finish** and **Cancel** buttons, and the navigational hierarchical tree on the left side of the window. These components are centralized in one location of the code consistently throughout the application, and are simply updated to reflect any changes. For example, if the panel displayed is the last panel in the sequence, the **Next** button is disabled to the user. These components are automatically populated into a wizard and managed by the class extended inside the General Wizard Framework. Chapter 8 gives more details on the implementation of these classes.

Navigational Tree

In the Parking Allowance and Site Area Calculation Wizard shown above, the *Facility Composer* Logo sits in a compositional tree that outlines the sequence of the wizard's panels. By clicking on the plus signs "+" on the left side of each tree node/leaf, one can expand the tree to view the corresponding subtitles, whereas

the subtitles indicate input or output wizard data. The tree offers non-linear means to navigate through the wizard. By clicking on any one of the titles or subtitles, one can jump directly to the desired section within the wizard. However, if the user clicks on an area that requires a previous operation that has not been completed, an error message dialog box appears indicating the requirements to be fulfilled before the user is able proceed to the selected section. Another attribute of the composition tree is that as the user runs through the application by means of pressing the **Next** and **Back** buttons, the title of the corresponding panel or section that is currently displayed is highlighted in the composition tree. This allows the user to have knowledge of the status of the sequence as well as the schematic model of the application. Navigational trees and guides provide the user with an organized and automated view of the wizard's sequence and sections.

5 Detailed Criteria Wizard Example — The Plumbing Fixture Calculation Wizard

The purpose of the Plumbing Fixture Calculation Wizard is to facilitate the fixtures calculation for a specified building type and occupancy level. The objective of the wizard is to determine information like the number of water closets, urinals, lavatories, bathtubs, showers, and drinking fountains, and in some cases, to offer the necessary area to construct restrooms.

Design Criteria

The requirements for the number of fixtures required for a facility are based on the facility type and occupancy among other things (e.g., Tables 3 to 8).

Army Reserve Training Building

Table 3. Female Fixture Allowances from UFC 4-171-05 Appendix F.

Peak Occupancy	Water Closets	Lavatories	Showers	Total	Area (sq ft [SF])
01 to 15	1	1	1	3	150 SF
06 to 35	2	2	1	5	175 SF
06 to 55	3	3	1	7	225 SF
08 to 60	4	3	1	8	250 SF
01 to 80	4	4	1	9	275 SF
01 to 90	5	4	1	10	300 SF
01 to 110	5	5	2	11	300 SF
11 to 125	6	5	2	13	350 SF
26 to 150	6	6	2	14	375 SF
51 to 170	7	6	2	15	400 SF
71 to 190	7	7	2	16	400 SF
91 to 215	8	7	2	17	425 SF
16 to 230	8	8	2	18	450 SF
31 to 270	9	8	3	20	475 SF
71 to 305	9	9	3	21	500 SF
06 to 310	10	10	3	23	500 SF
11 to 350	10	10	4	24	575 SF
51 to 390	11	11	4	26	600 SF
91 to 395	11	11	4	26	600 SF

Peak Occupancy	Water Closets	Lavatories	Showers	Total	Area (sq ft [SF])
96 to 430	12	12	4	28	625 SF
31 to 440	12	12	5	29	625 SF
41 to 470	13	13	5	31	675 SF
71 to 485	13	13	5	31	675 SF
86 to 510	14	14	5	33	700 SF
11 to 530	14	14	5	33	700 SF
31 to 550	15	15	5	35	750 SF
51 to 575	15	15	6	36	750 SF
76 to 590	16	16	6	38	800 SF
91 to 620	16	16	7	39	825 SF
21 to 630	17	17	7	41	875 SF
31 to 665	17	17	7	41	875 SF
66 to 670	18	18	7	43	900 SF
71 to 710	18	18	7	43	900 SF

Table 4. Male Fixture Allowances from UFC 4-171-05 Appendix F.

Peak Occupancy	Water Closets	Urinals	Lavatories	Showers	Total	Area (sq ft [SF])
1 to 35	2	1	2	1	6	200 SF
6 to 55	2	1	3	1	7	225 SF
6 to 60	3	1	3	1	8	250 SF
1 to 80	3	1	4	1	9	250 SF
1 to 90	3	2	4	1	10	300 SF
1 to 125	4	2	5	2	13	325 SF
26 to 150	4	2	6	2	14	350 SF
51 to 170	5	2	6	2	15	375 SF
71 to 190	5	2	7	2	16	400 SF
91 to 215	6	2	7	2	17	400 SF
16 to 230	6	2	8	2	19	450 SF
31 to 270	6	3	8	3	20	475 SF
71 to 305	7	3	9	3	22	500 SF
6 to 310	7	3	10	3	23	500 SF
11 to 350	8	3	10	4	25	575 SF
51 to 390	8	3	11	4	26	600 SF
91 to 395	9	4	11	4	28	625 SF
96 to 430	9	4	12	4	29	625 SF
31 to 440	10	4	12	5	31	675 SF
41 to 470	10	4	13	5	32	700 SF
71 to 485	10	5	13	5	33	700 SF
86 to 510	10	5	14	5	34	725 SF
11 to 530	11	5	14	5	35	750 SF
31 to 550	11	5	15	5	37	800 SF
51 to 575	12	5	15	6	38	800 SF
76 to 590	12	5	16	6	39	825 SF

Peak Occupancy	Water Closets	Urinals	Lavatories	Showers	Total	Area (sq ft [SF])
91 to 620	12	6	16	7	40	850 SF
21 to 630	12	6	17	7	42	875 SF
31 to 665	13	6	17	7	43	900 SF
66 to 670	13	6	18	7	44	925 SF
71 to 710	14	6	18	7	45	950 SF

Employee Facilities: Library, Recreational Workshop, Bowling Alley

Table 5. Water Closet Allowances for Employees from TI 800-01 Chapter 15.

Number of Employees	Water Closets
1 to 15	1
16 to 35	2
36 to 55	3
56 to 80	4
81 to 110	5
111 to 150	6
151 and over	6 for the first 150, plus 1 additional fixture for each additional 40 employees

Table 6. Lavatory Allowances for Employees from TI 800-01 Chapter 15.

Number of Employees	Lavatories
1 to 15	1
16 to 35	2
36 to 60	3
61 to 90	4
91 to 125	5
126 and over	1 additional fixture for each 45 Employees

The criterion for drinking fountains requires one drinking fountain for each 75 employees and at least one fountain per floor will be provided.

UEPH (Unaccompanied Enlisted Personnel Housing)

Table 7. UEPH fixture allowances from TI 800-01, Appendix B.

Occupants	Minimum Number of Persons Per Fixture					
	Water Closets	Shower	Lavatories	Bathtubs	Urinals	Drinking Fountains
Recruits						
Male	10	8	8	0	15	75*
Female	6	8	6	30	None	75*
E1 to E4	2	Note**	2	2	None	1 per floor
E5 to E9	1	Note**	1	1	None	1 per floor

* Additional drinking fountain for every 30 occupants per floor above the initial 75 occupant requirement.
** Shower stalls may be substituted for bathtubs.

UOPH (Unaccompanied Officer Personnel Housing) (TI 800-01 pg 15-3)

The criterion for plumbing fixtures for all UOPH, grades W1 to 06, requires a bathroom for each suite with one lavatory, one water closet, and one bathtub with shower. Each floor will include one drinking fountain

Temporary Lodging Facilities

The criterion for temporary lodging facilities requires, for every two (2) guest rooms, one water closet, two lavatories, and one shower compartment or bath-tub/shower combination. Additionally, a common toilet room will be provided for the office and lounge.

Religious, Welfare and Recreational Facilities for Persons other than Employees

Table 8. Religious, welfare and recreational facilities fixture allowances from TI 800-01, Ch 15.

Occupancy	Minimum Number of Persons Per Fixture When More than One Fixture is Required				
	Water Closets	Lavatories	Urinals	Showers	Drinking Fountains
Chapel (Congregation Only)					
Male	300	150	300	None	400
Female	150	150	None	None	400
Enlisted Personnel Service Club (Patrons Only)					
Male	150	150	200	None	500
Female	100	100	None	None	500
General Education Development Building (Students Only)					
Male	40	25	40	None	100
Female	25	25	None	None	100

Occupancy	Minimum Number of Persons Per Fixture When More than One Fixture is Required				
	Water Closets	Lavatories	Urinals	Showers	Drinking Fountains
Gymnasium, Field House					
Male	30	30	40	15	100
Female	20	25	None	15	100
Installation Restaurant or Cafeteria, NCOs' Open Mess, Officers' Open Mess (Patrons Only)					
Male	200	200	300	None	500
Female	150	150	None	None	500
Swimming Pool (Swimmers Only)					
Male	40	40	40	30	100
Female	20	40	None	30	100
Theater, Enlisted Personnel Dining Facilities (Patrons Only)					
Male	250	200	250	None	400
Female	150	150	None	None	400

Algorithms

Army Reserve Training Building

To show the values in the table, the wizard compares the value of the user input with these tables and shows the corresponding results in the table.

```
Because no specific requirement was stated in the Army Reserve
Design Guide for calculating the number of drinking fountains, we
used the rule from TI-800-01 Chapter 15 that states that a building
requires 1 drinking fountain for every 75 persons in the building
and at least one fountain per floor.  This results in the equation:
'calculate total number of drinking fountains
   DrinkingFountains = RoundDown(BuildingOccupancy/75)

   'if there are less fountains than floors then distribute fountains
   'starting with first floor going up until they run out
   'note: 1 fountain per floor minimum
   If (DrinkingFountains <= TotalNumberOfFloors)
      For Each Floor
         Floor[x].Fountains = 1
      Next

   'if there are more fountains then floors then distribute evenly
   Else
      RemainingFountains = DrinkingFountains
      RemainingFloors = TotalNumberOfFloors
      For Each Floor
         Floor[x].Fountains = RoundUp(RemainingFountains/RemainingFloors)
         RemainingFountains = RemainingFountains - Floor[x].Fountains
         RemainingFloors = RemainingFloors - 1
      Next
```

Employee Facilities

The equation to calculate the number of water closets required if the occupancy exceeds the 150 occupancy level (based on the rule of 6 for the first 150, plus 1 additional fixture for each additional 40 employees) is:

```
If (BuildingOccupancy > 150)
   ExceededOccupancy = BuildingOccupancy - 150
   Water Closet = 6 + RoundDown(ExceededOccupancy/40)
Else
   Water Closet = TableLookup(BuildingOccupancy)
```

The equation to calculate the number of lavatories required if the occupancy exceeds the 125 occupancy level is:

```
If (BuildingOccupancy > 125)
   ExceededOccupancy = BuildingOccupancy - 125
   Lavatories = 5 + RoundDown(ExceededOccupancy/45)
Else
   Lavatories = TableLookup(BuildingOccupancy)
```

To calculate the number of drinking fountains, we used the rule that states that a building requires 1 drinking fountain for every 75 persons and at least one fountain per floor. This results in the equation identical to the Army Reserve Training Building algorithm on the prior page.

UEPH (Unaccompanied Enlisted Personnel Housing)

To calculate the number of drinking fountains, the rule stated a building needed 1 drinking fountain for every additional 30 persons per floor above the initial 75 occupant requirement. This results in the equation:

```
'calculate total number of drinking fountains
DrinkingFountains = RoundDown(BuildingOccupancy/75)

'if there are less fountains than floors then distribute fountains
'starting with first floor going up until they run out
'note: 1 fountain per floor minimum
If (DrinkingFountains <= TotalNumberOfFloors)
   For Each Floor
      Floor[x].Fountains = 1
   Next

'if there are more fountains then floors then distribute evenly
Else
   RemainingFountains = DrinkingFountains
   RemainingFloors = TotalNumberOfFloors
   For Each Floor
      Floor[x].Fountains = RoundUp(RemainingFountains/RemainingFloors)
      RemainingFountains = RemainingFountains - Floor[x].Fountains
      RemainingFloors = RemainingFloors - 1
   Next



'if more than 30 occupants on a floor then add one additional fountain
```

```
'for every 30 occupants
For Each Floor

    'adjust density such that if < 30 people/floor then density = 0
FloorOccupancyDensity = RoundDown(Floor[x].Occupancy/30)

    'if more than 30 people on this floor then add the
    'additional fountains on this floor
    If (FloorOccupancyDensity) > 0)
       Floor[x].Fountains = Floor[x].Fountains + FloorOccupancyDensity
Next
```

Temporary Lodging Facilities

The equation to calculate data:

```
FixtureDensity = RoundDown(BuildingOccupancy/2)
For Each FixtureType
   FixtureType[x] = TableLookup(FixtureType, FixtureDensity)
```

Religious, Welfare and Recreational Facilities for Persons other than Employees

The equation to calculate data:

```
FixtureDensity = RoundDown(BuildingOccupancy/2)
For Each FixtureType
   FixtureType[x] = TableLookup(FixtureType, FixtureDensity)
```

Sequence

The development of the Plumbing Fixture wizard used similar design principals and IUI guidelines as the Parking Allowance and Site Area Wizard.

1. Introduction Panel

The first panel is the Introduction Panel (Figure 12). The panel briefly outlines the functions of the wizard and also provides the user with a check box to select if they do not want this panel to be displayed the next time the wizard is launched. The main purpose of this screen is to give the user a mental model of what activity to expect.

Figure 12. Introduction panel of plumbing fixture planning wizard.

2. Building Type Panel

In the Building Type Panel (Figure 13), the user must select the building type for which they want to calculate the number of plumbing fixtures. If the user does not select any item, the wizard defaults to the first selection off the list, which in this case is the "Training Building."

Figure 13. Building type panel.

3. Criteria Panel

Figures 14 and 15 shows two examples of the Criteria Panel. This panel presents the required criteria specific to the building type chosen in the previous panel. The user is expected to enter the values requested in the numeric text fields and to press the **Compute** button to obtain the number of fixtures required for the facility in the numeric text field found on the bottom of the screen.

Figure 14. Training building criteria panel.

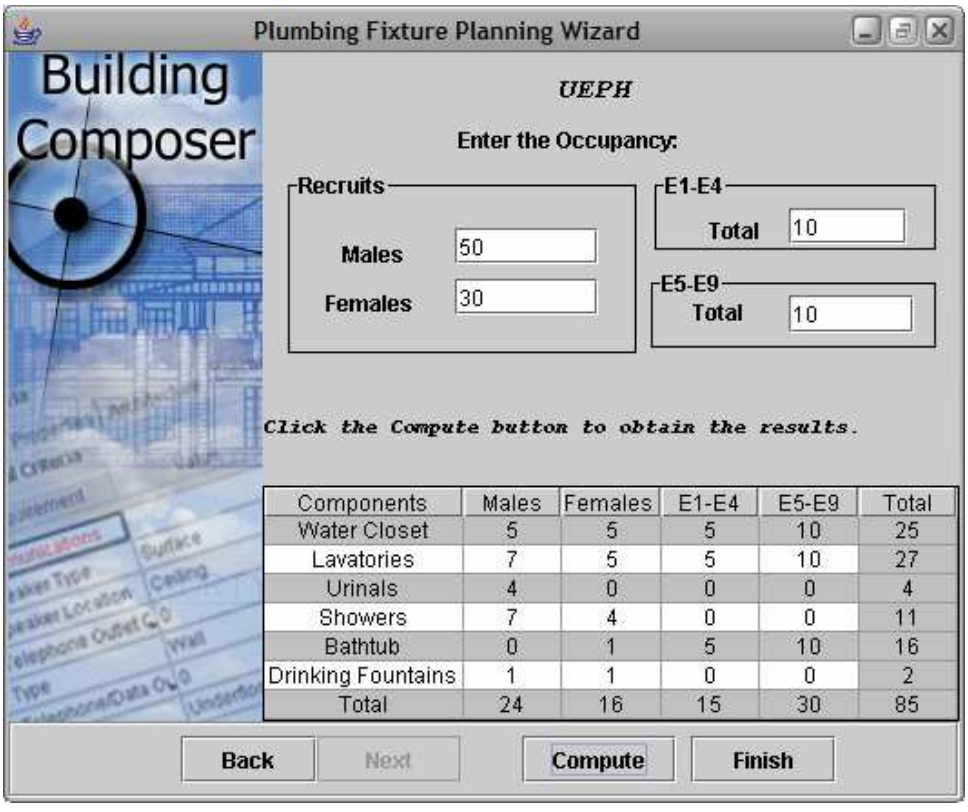


Figure 15. UEPH criteria panel.

Implementation Issues

As with the Parking Allowance and Site Area Wizard, components of the Plumbing Fixture Planning Wizard span multiple screens throughout the process, for example, the **Back, Next, Finish**, and **Cancel** buttons. These components are centralized in one location of the code, are consistent throughout the application, and are simply updated to reflect any changes. For example, if the panel displayed is the last panel in the sequence, the **Next** button is disabled to the user. These components are automatically populated into a wizard and managed by the class extended provided inside the General Wizard Framework. For more details on the implementation of these classes, see Chapter 8.

6 Detailed Criteria Wizard Example (Sustainable Designer's Aid)

The Sustainable Designer's Aid is a more complex wizard application that provides guidance to support the consideration of sustainable design and development principles in planning decisions and projects. The application is intended to be used throughout the design process to guide the project towards a sustainable solution as well as to score and rate the resulting facility. The application is programmed based on the Sustainable Project Rating Tool (SPiRiT) and determines the rating level of the project at its conclusion.

Design Criteria and Algorithms

SPiRiT consists of a list of sustainability credits sub-divided into eight sections (1.0 Sustainable Sites, 2.0 Water Efficiency, 3.0 Energy and Atmosphere, etc.). The purpose of the application is to provide facility type, regional, and customer-specific guidance to achieve an environmentally sustainable facility, in addition to providing a way to reduce the ambiguity of the requirements needed to satisfy each credit. Finally, it provides a mechanism to record the decisions that were made for clarity of record and for future reuse.

For each credit, a maximum number of points are assigned and the following entries are specified: Intent, Requirement(s) and Technologies/Strategies. The "intent" states the primary goal for the credit. The "requirement" lists quantifiable conditions necessary to achieve the stated intent. Suggested technologies, strategies, and referenced guidance on the means to achieve identified requirements are also defined for each item.

For a credit to be fulfilled, its constituent components must first be met. A credit may contain one to multiple requirements. Each requirement may, in turn, contain a number of criteria (Figure 16). In some scenarios, only a fraction of the criteria must be met to fulfill a certain requirement. However, in the majority of the cases all specified criteria must be met to fulfill a particular requirement. Finally, each requirement must be met within a specific credit in order to receive the specified score for the item.

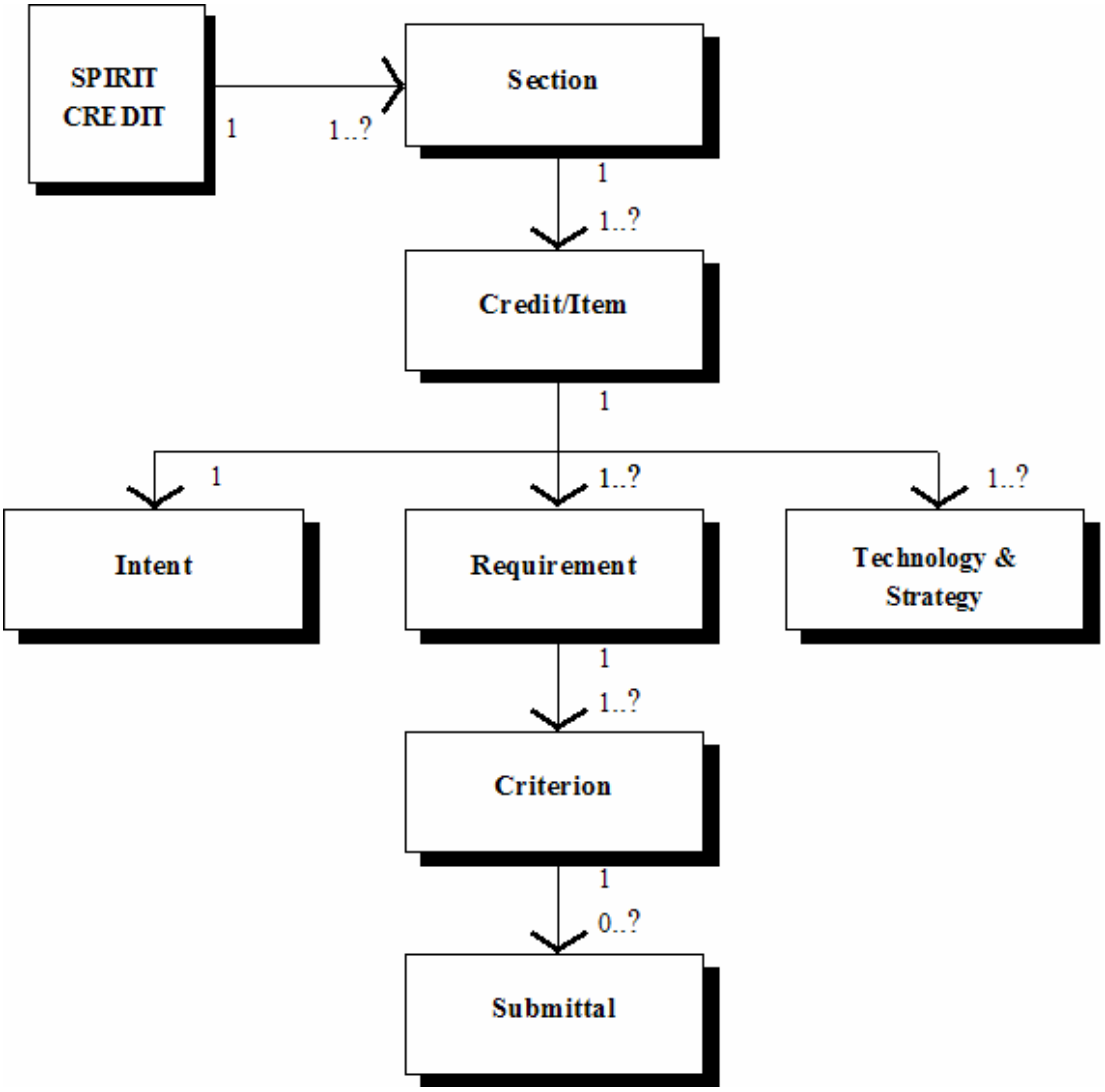


Figure 16. Components of a SPiRiT credit.

Sequence

The application is divided into five main sections:

1. Introduction

The purpose of the Introductory Section is to familiarize the user with the Sustainable Designer's Aid and to list and briefly describe the different sections within the application (Figure 17).

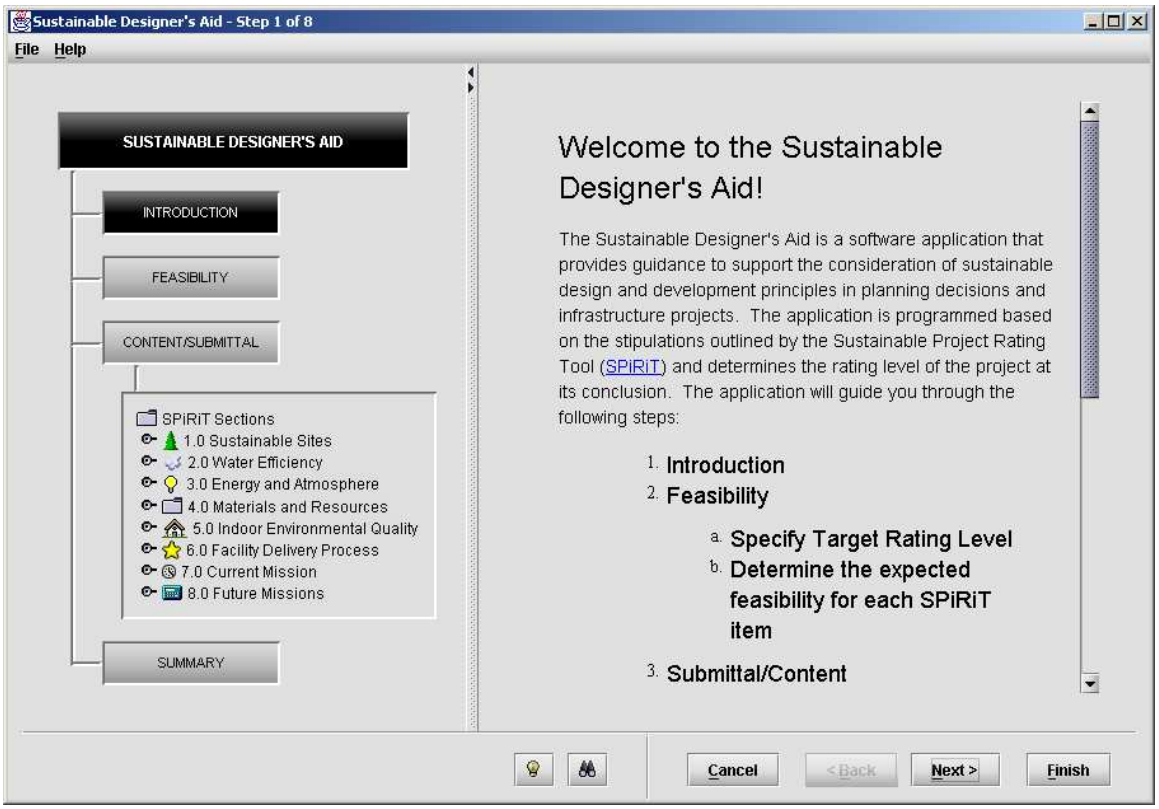


Figure 17. Introduction panel of sustainable designer's aid.

2. Feasibility Section

The "Feasibility Section" follows the Introduction. The purpose of the feasibility process is to provide the user the means to eliminate unattainable SPiRiT design credits while at the same time identifying items that might be possible (feasible) or applicable for the project. The first task for the user, within the feasibility section, is to determine the target rating level for the project under consideration. The options are platinum, gold, silver, and copper and are measured in SPiRiT point units (Figure 18).

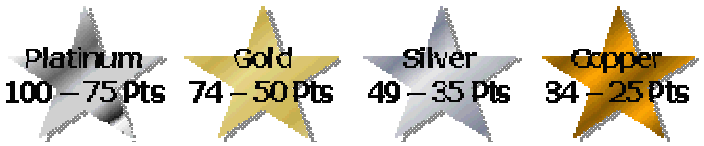


Figure 18. SPiRiT rating levels.

Keeping the desired rating level in mind, the user's responsibility is to study the list of design credits shown in the table given, reading both the description and the intent of each item. The user is to determine how many of the maximum points available for a specific item will be the target or goal for the current pro-

ject. After the user has completed this task, the user confirms that the total targeted points meet the desired rating level. The user may eliminate, or filter out, those items that are not applicable or attainable by deselecting the corresponding check box(es). Items that are not selected will not be visible in the next section of the wizard. To activate the filter or visibility option, the user is required to select the toggle button in the middle on the bottom of the wizard window (visible in the bottom center of Figure 19).

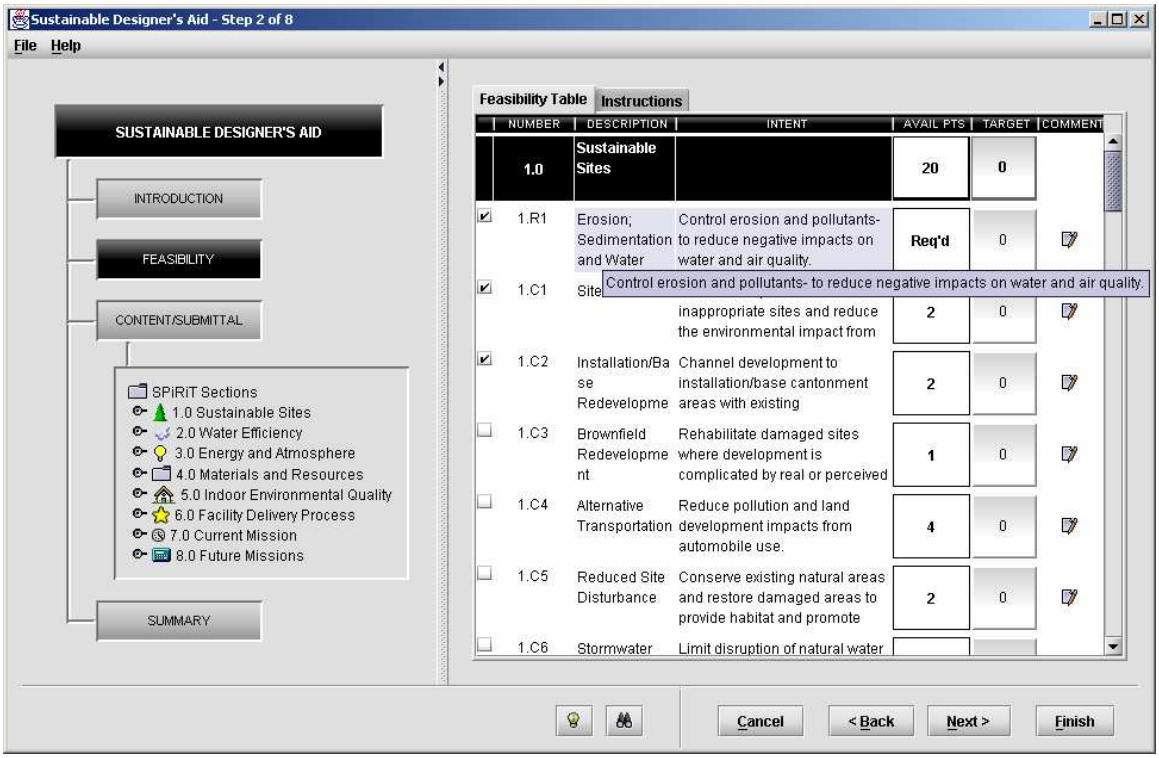


Figure 19. SDA's feasibility section.

3. Content/Submittal Section

The next section of the wizard demands that the user proceed through all SPiRiT credits to identify and submit proof for each criterion met. This section is composed of a series of panels known as Wizard Content Panels. A Wizard Content Panel displays all the components of a SPiRiT credit such as: the section title and number scheme (top black row), credit title and number scheme, maximum points assigned (highlighted in yellow), intent, requirements, criteria, and technologies and strategies. Figure 20 shows how the requirements are represented for each SPiRiT item with text fields while blue sphere bullets represent the criteria.

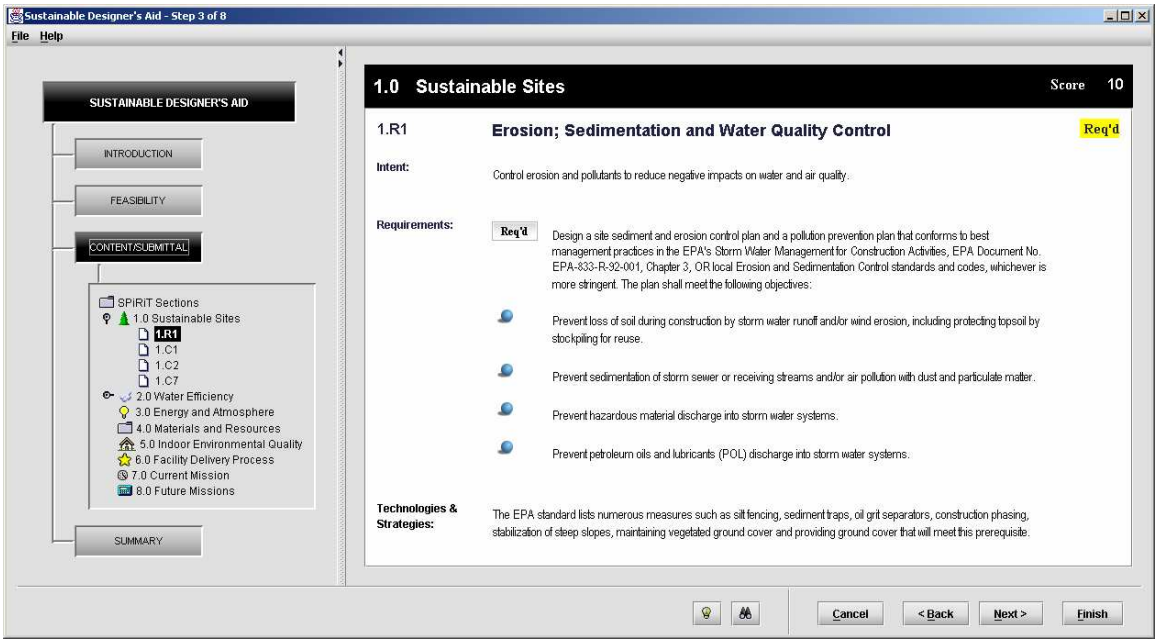


Figure 20. Wizard content panel of SPiRiT item 1.R1.

To obtain the points associated with a SPiRiT credit, the user must submit proof that all the necessary criteria have been properly fulfilled. To enter submittals, the user is required to click on the desired criterion bullet to access the corresponding Submittal Dialog. Each Submittal Dialog (Figure 21) contains a list of predefined submittals on the left hand side as well as buttons to add and remove user-defined submittals. A submittal may be a: (1) description, (2) file, (3) drawing, (4) hyperlink, (5) worksheet, or (6) Facility Composer criteria.

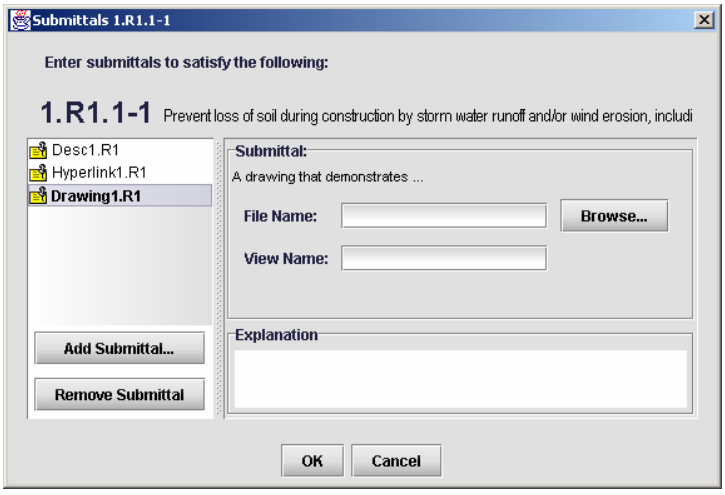


Figure 21. Submittal dialog for criterion 1.R1_1-1.

Certain criteria require that precise calculation be computed based on information entered. Predefined submittals that require calculations are known as worksheets and are presented to the user in the format shown in Figure 22.

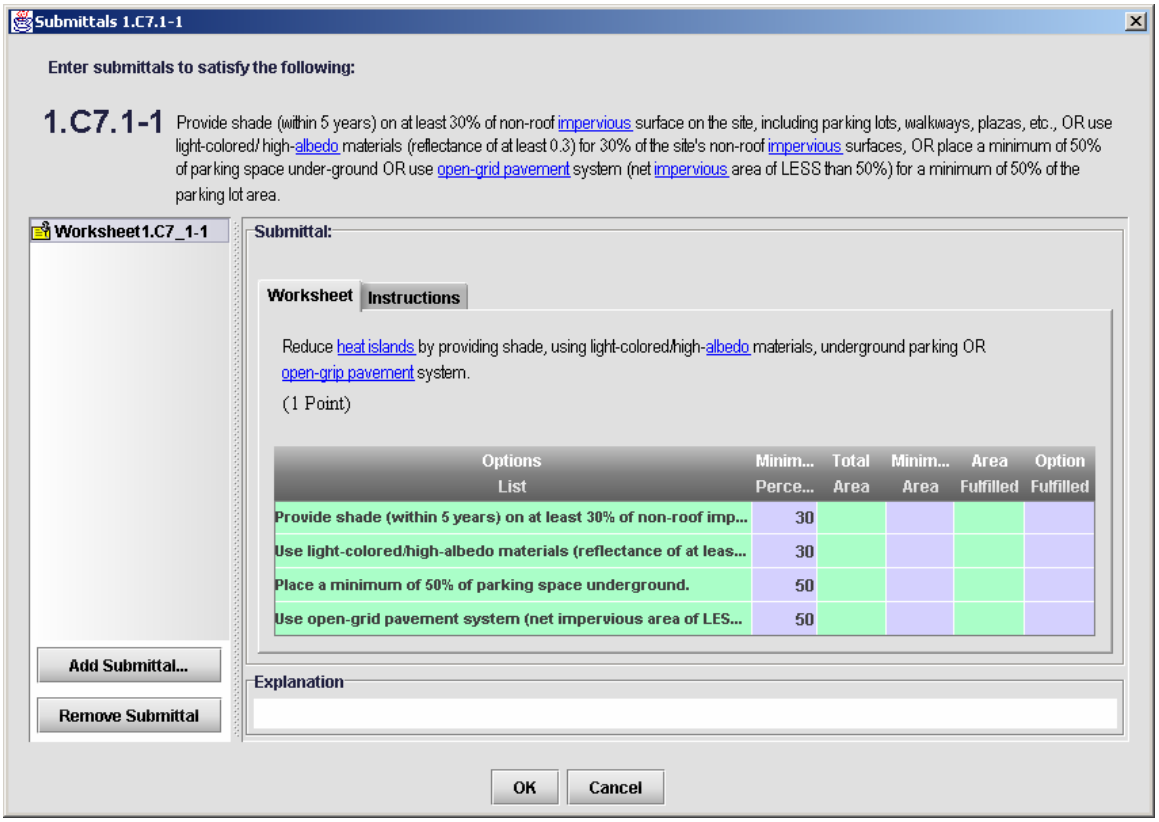


Figure 22. Submittal worksheet for criterion 1.C7_1-1.

The user may choose to add or remove certain submittals by clicking on the buttons found directly below the list of submittals on the left hand side of the Submittal Dialog. When the user presses the "**Add Submittal**..." button, the Add Submittal Dialog (Figure 23) appears allowing the user to add a submittal of the type description, file, hyperlink, or drawing.

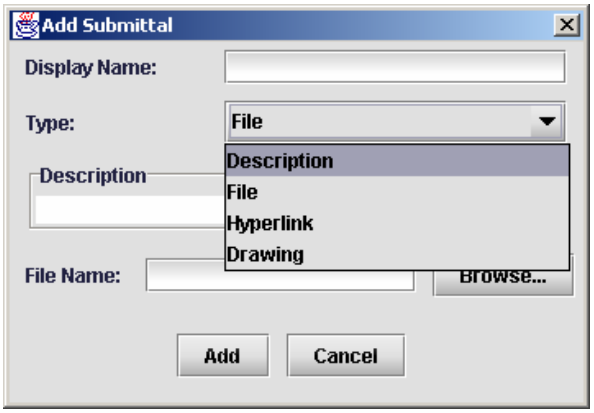


Figure 23. Add submittal dialog.

After all the required submittals have been entered, the user may press the **OK** button on the Submittal Dialog to return to the corresponding Wizard Content Panel. If the submittals are verified positively as fulfilling the corresponding criterion, a check mark replaces the blue sphere criterion bullet certifying completion (Figure 24). Subsequently, the user may continue to enter submittals for all the listed criteria to receive the points for the particular requirement.

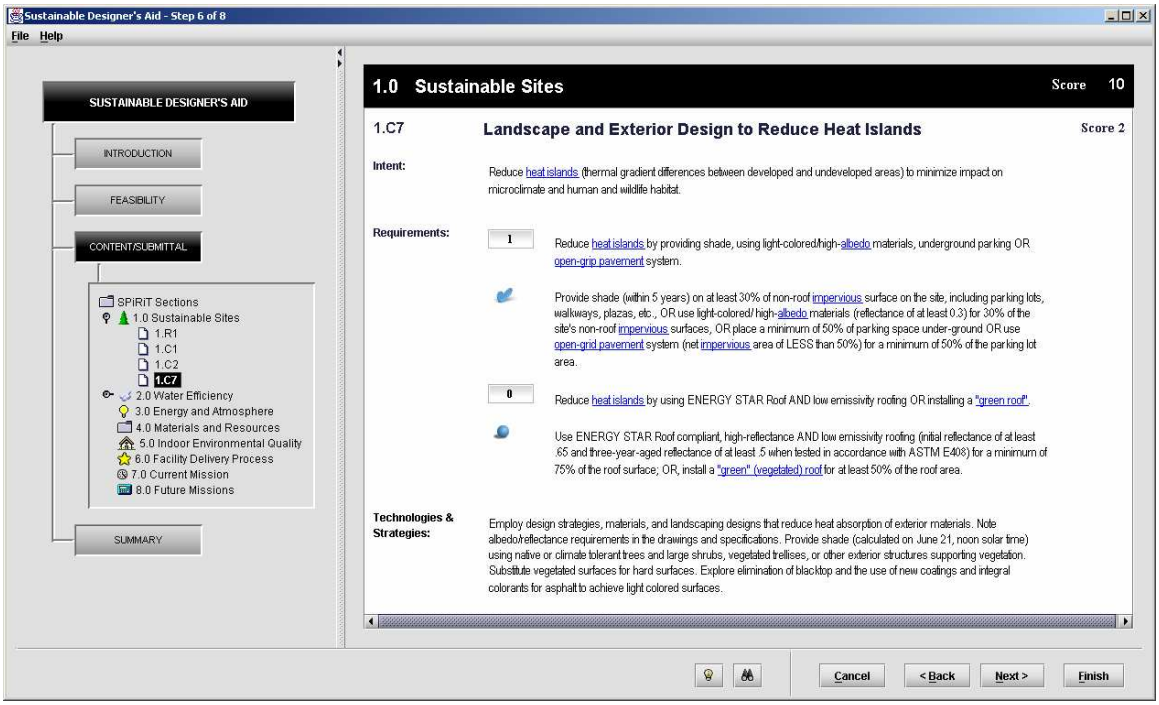


Figure 24. Fulfilled criterion for item 1.C7.

4. Summary Section

The concluding section of the application displays a table with a list of the all the SPiRiT items, and presents the user with the targeted points determined in the Feasibility Section versus the actual points obtained throughout the Content/Submittal Section. In addition, the overall rating level of the resulting facility is displayed on the bottom of the table (Figure 25).

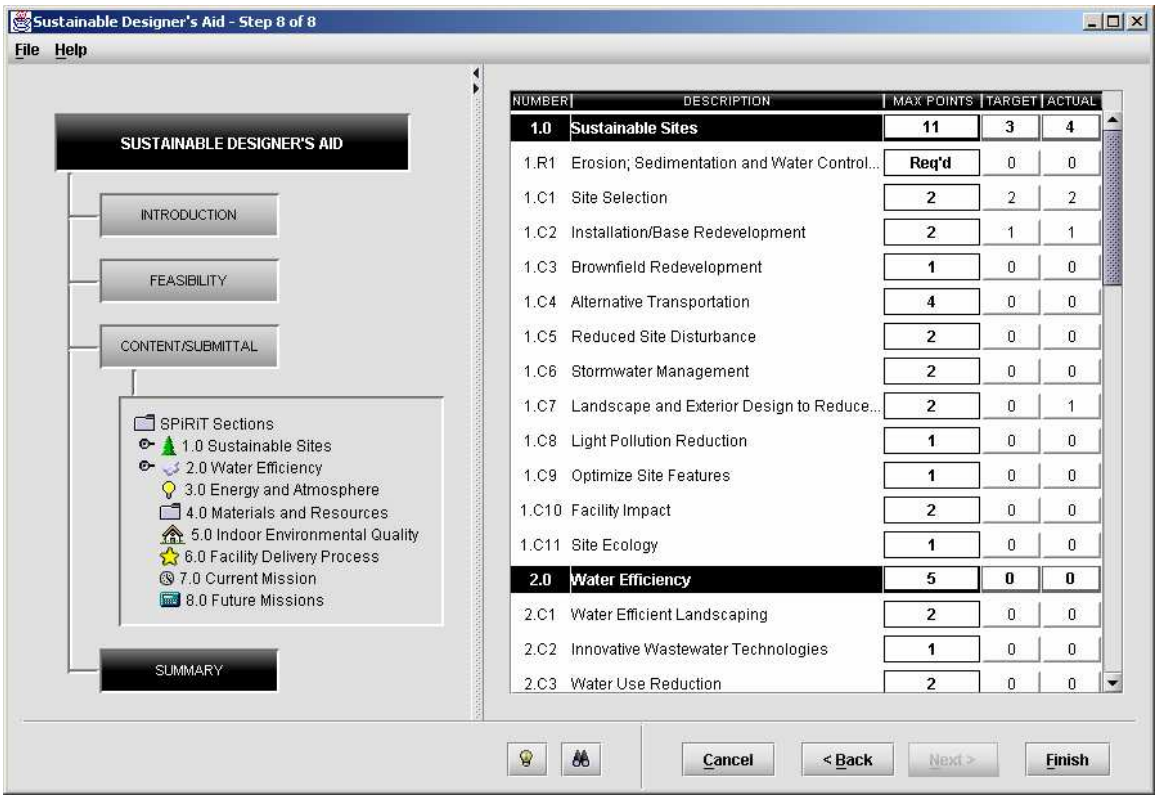


Figure 25. SPiRiT summary table.

Features and Implementation Issues

Java Web Start

The Sustainable Designer's Aid is currently available on (and may be launched from) the Internet. Java Web Start is a launching mechanism with the purpose to simplify deployment of Java applications. Launching applications with this mechanism allows the user to install an application with a single click from a web browser. The launching mechanism includes the security features of the Java 2 Platform, maintaining the integrity of the user's data and files. The first time the application is launched using Java Web Start, the packaged application is obtained from the server and stored on the user's local machine. The second time the application is launched, this will occur almost instantaneously given that the latest version has already been stored on the user's local machine. Furthermore, the second time around, Java Web Start will prompt the user for desktop or start menu icons. Finally, one of the most beneficial advantages of Java Web Start is that every time the application is launched, either from a web browser link or an icon on the user's local machine, the launching mechanism will check for new versions of the application, provided the user is connected to a

web browser. Therefore, the latest version of the software is obtained and presented to the user consistently.

Java Help

Another attribute of the Sustainable Designer's Aid is a full-featured data driven help system that permits users to view documentation that provides assistance in relation to the application through the use of Sun Microsystems's Java Help (Figure 26). Java Help supports flexible display, compression and encapsulation of files, customization and extensibility, context sensitive help, merging capabilities and dynamic updating. Java Help software, when implemented properly into an application, can be displayed to the user upon an action performed, such as selecting the Help item in the Help menu. The help system is composed of an individual frame with a split pane in the center that separates the navigator tabs on the left from the content panel on the right. Java Help includes help navigator views such as Table of Contents, Index and Full-Text Search while the content panel on the right displays the specified HTML page containing the useful, detailed and thorough help guidance.

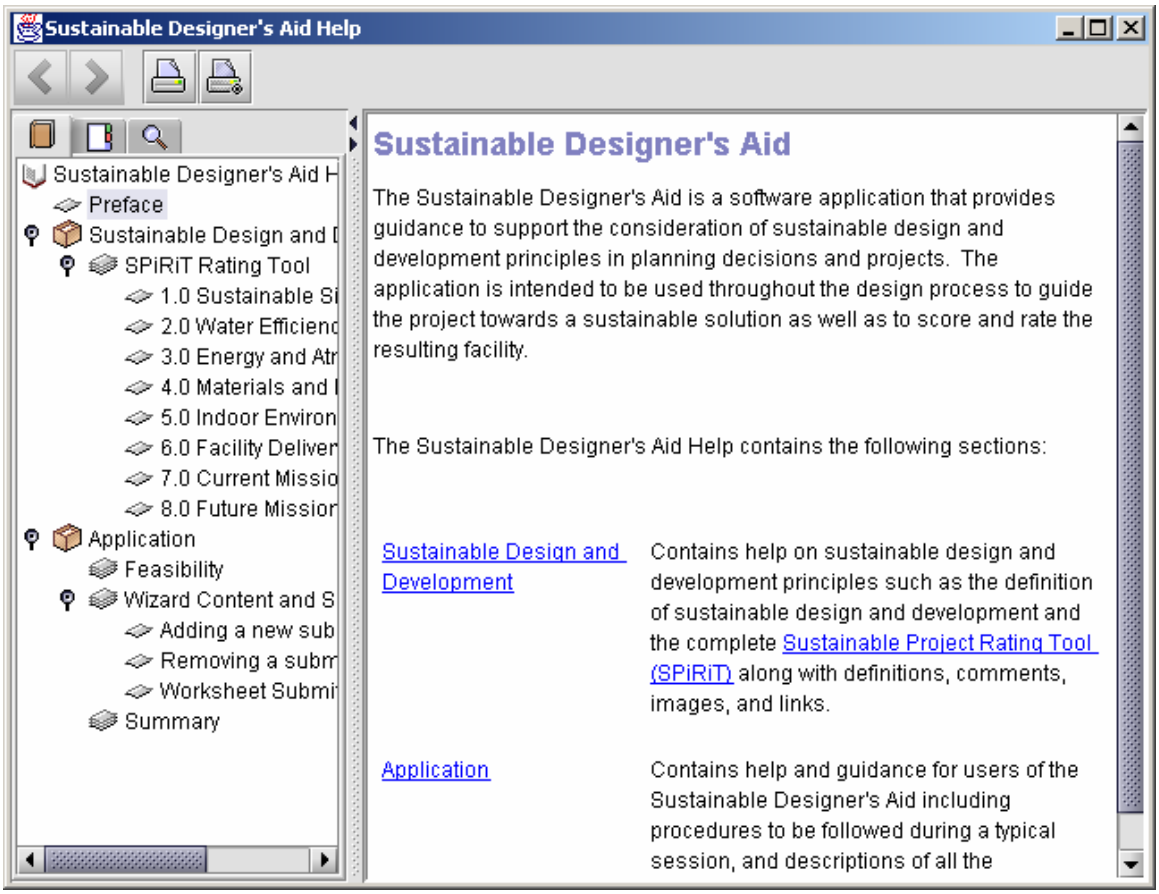


Figure 26. Sustainable designer's aid help window.

7 Detailed Analysis Wizard Example — DoD Minimum Antiterrorism Standards for Buildings (MATSB) Wizard

General Use Case Description

The purpose of the DoD Minimum Antiterrorism Standards for Buildings Wizard (MATSB) is to facilitate in the determination of all the applicable minimum antiterrorism/force protection standards addressed in UCF 4-010-01, DoD Minimum Antiterrorism Standards for Buildings. The wizard is designed to assist facility planners and designers to determine and identify a projects minimum applicable antiterrorism/force protection requirement by prompting the User for project-specific input and then walking them through a decision tree process. Once through the wizard a set of minimum standoff distance requirements and a list of recommended Force Protection design requirements/recommendations for site planning and design of structural, architectural, electrical, and mechanical systems are generated.

Design Criteria

After careful analysis of UFC 4-010-01, an approach for establishing standoff requirements was developed (Table 9). This approach is based on defining the category of the project being developed.

Table 9. Approach for establishing standoff requirements.

	Controlled Perimeter	Roadways/ Parking	Trash Containers	Adjacent Buildings
New Buildings				
No Exemption	Minimum standoff can be met; or must do hardening with analysis with lower minimum	Minimum standoff can be met; or must do hardening with analysis with lower minimum	Minimum standoff can be met; or must do hardening with analysis with lower minimum	Minimum standoff can be met; or must do hardening with analysis with lower minimum
Roadways Parking Exemption	Minimum standoff can be met; or must do hardening with analysis with lower minimum	Minimum is recommended	Minimum standoff can be met; or must do hardening with analysis with lower minimum	Minimum standoff can be met; or must do hardening with analysis with lower minimum
Full Force Protection Exemptions	Minimum is recommended	Minimum is recommended	Minimum is recommended	Minimum is recommended
Existing Buildings				
No Exemption	Minimum standoff can be met; or must do hardening with analysis with lower minimum	Minimum standoff can be met; or must do hardening with analysis with lower minimum	Minimum standoff can be met; or must do hardening with analysis with lower minimum	No requirement for existing buildings
Roadways Parking Exemption	Minimum standoff can be met; or must do hardening with analysis with lower minimum	Recommend to meet minimum or parking and other measures	Minimum standoff can be met; or must do hardening with analysis with lower minimum	No requirement for existing buildings
Full Force Protection Exemptions	Minimum is recommended	Recommend to meet minimum or parking and other measures	Minimum is recommended	No requirement for existing buildings.

However, to effectively determine the minimum requirements, a series of questions in three areas must be answered in sequential order. Step one involves questions to determine the project category. Step two involves the standoff distance analysis and step three of the process has questions relating to progressive collapse analyses.

Figure 27 shows an example of one of the decision tree diagrams developed for determining the category of a project. As you can see, there are many ways that one of the six categories can be selected simply based on the answers provided. All other areas of the process have a decision tree. The full set of decision tree logic documents can be found in Appendix B of this technical report.

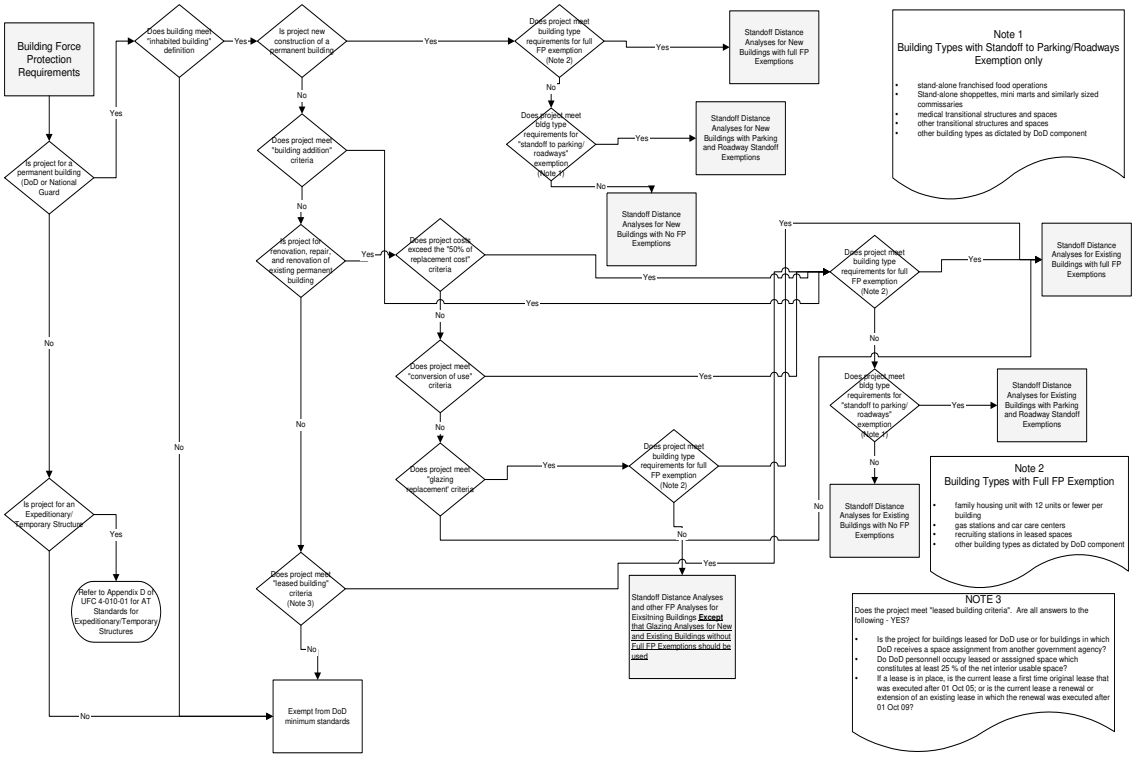


Figure 27. Flow chart for wizard.

Sequence

The DoD Minimum Antiterrorism Standards for Buildings Wizard is composed of an Introduction and five "Step" panels.

1. Introduction Panel

The first panel of the wizard is the Introduction Panel (Figure 28). The panel briefly outlines the process and functions of the wizard. The main purpose of this panel is to describe to the user what to expect when using the wizard.

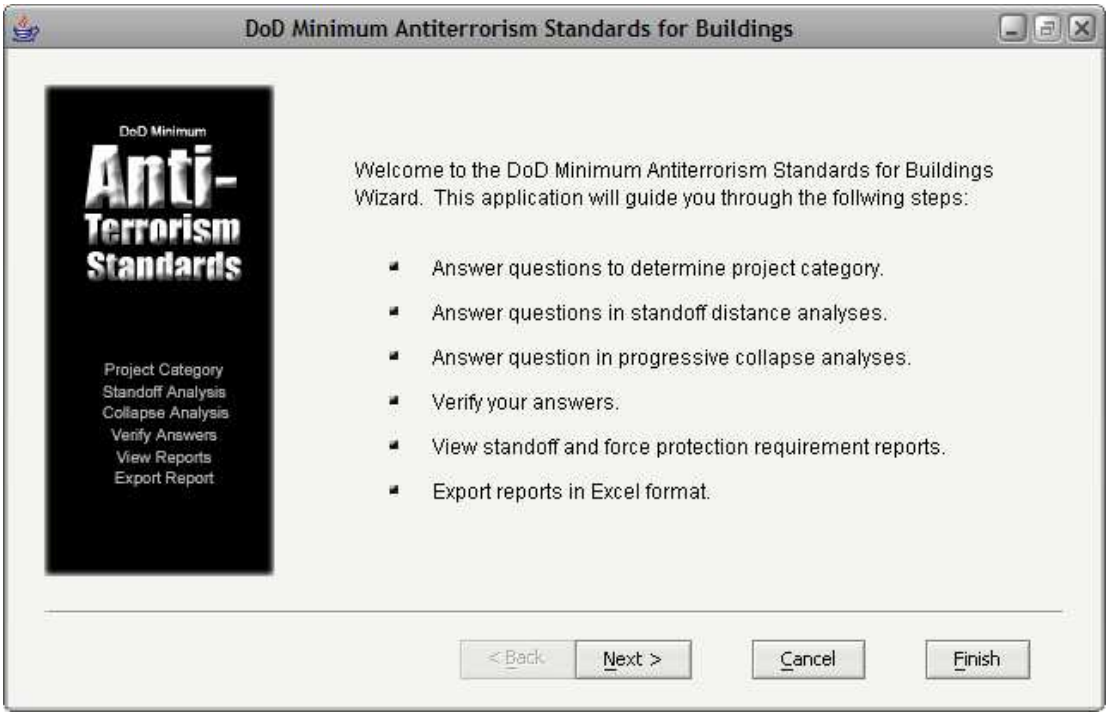


Figure 28. Introduction panel.

2. Step 1 Panels

As stated previously, this wizard consists of three main steps to identify the minimum requirements. Step 1 walks the user through the decision tree for determining the project category. Figure 29 shows the four areas of the panel. The top area of the panel describes the function of Step 1. The middle portion of the panel lists the question being asked and then contains two radio buttons, one for an answer of Yes and the other for an answer of No. Once the user selects an answer, the **NEXT** button is enabled to continue the process.

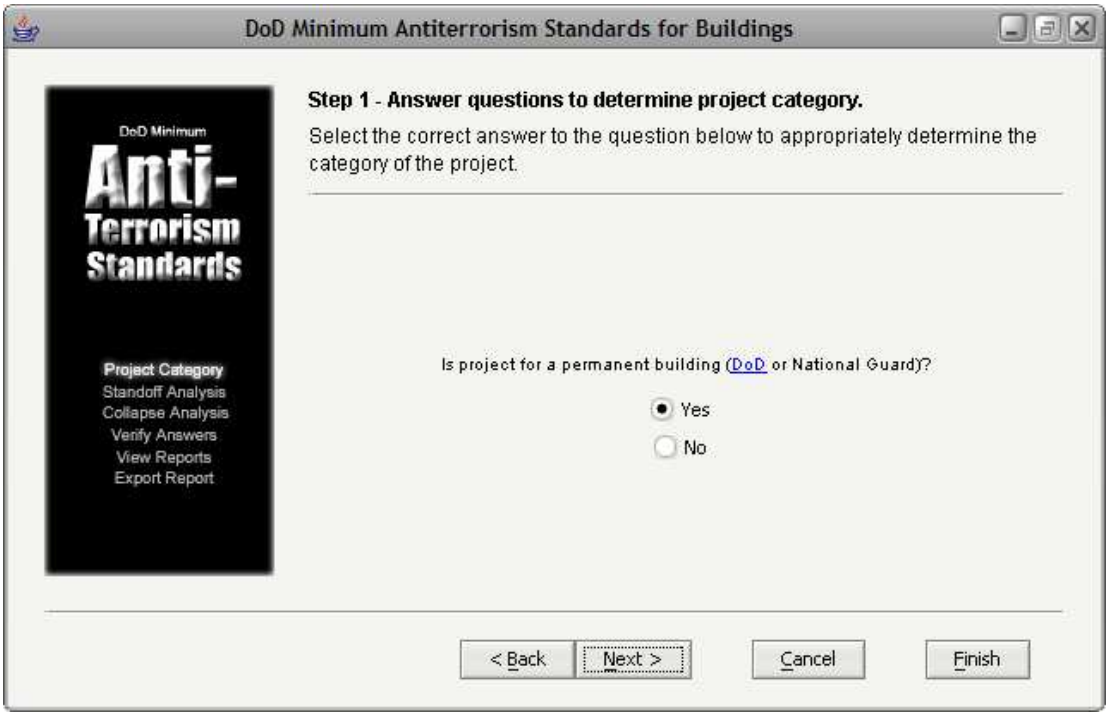


Figure 29. Determine project category panel.

3. Step 2-3 Panels

Once the project category is determined, a series of questions needs to be answered to determine the standoff distance requirements and progressive collapse analysis. Figures 30 and 31 show the layout of panels in Step 2 and 3, which are essentially the same as those in Step 1.

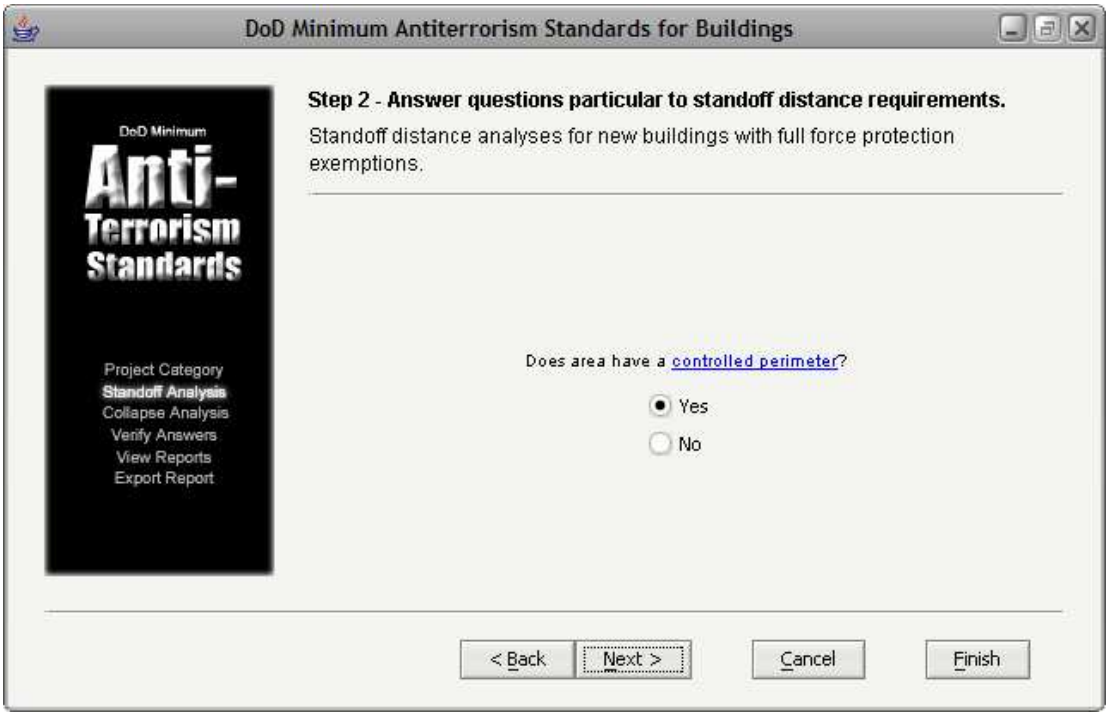


Figure 30. Standoff analysis panel.

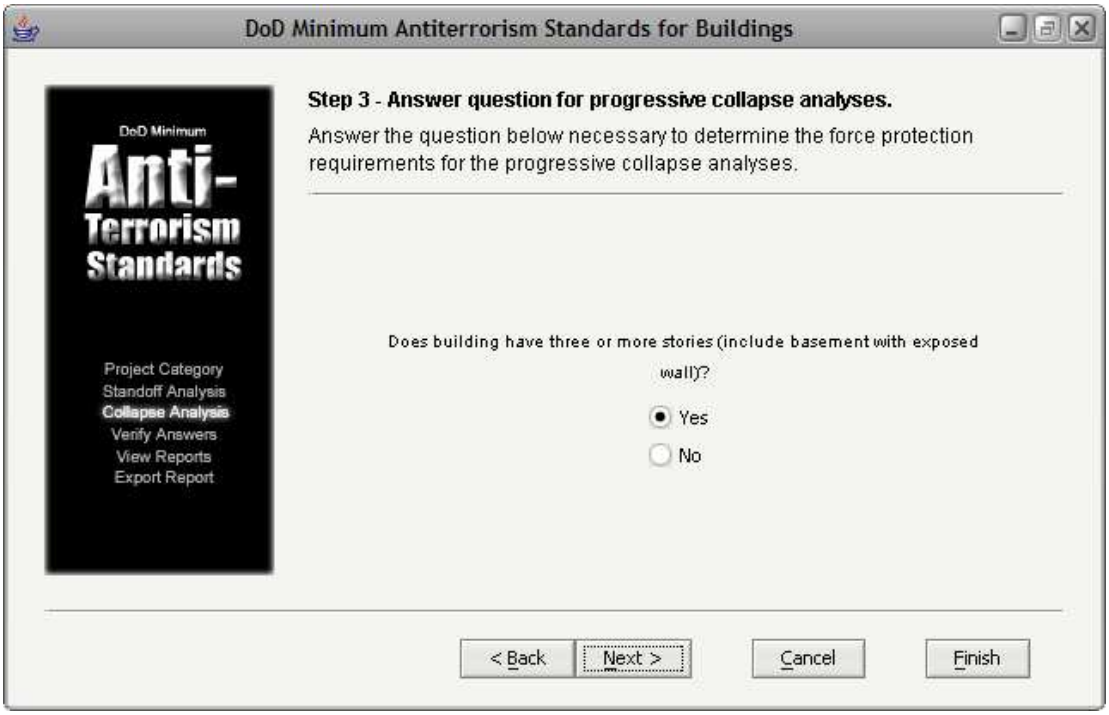


Figure 31. Collapse analysis panel.

4. Step 4 Panel

After all questions in the first three steps are answered, the verification panel (Figure 32) is displayed. At this time the user can go back and review the selected answers. To change an answer, the user simply selects the answer and the program will return to that question

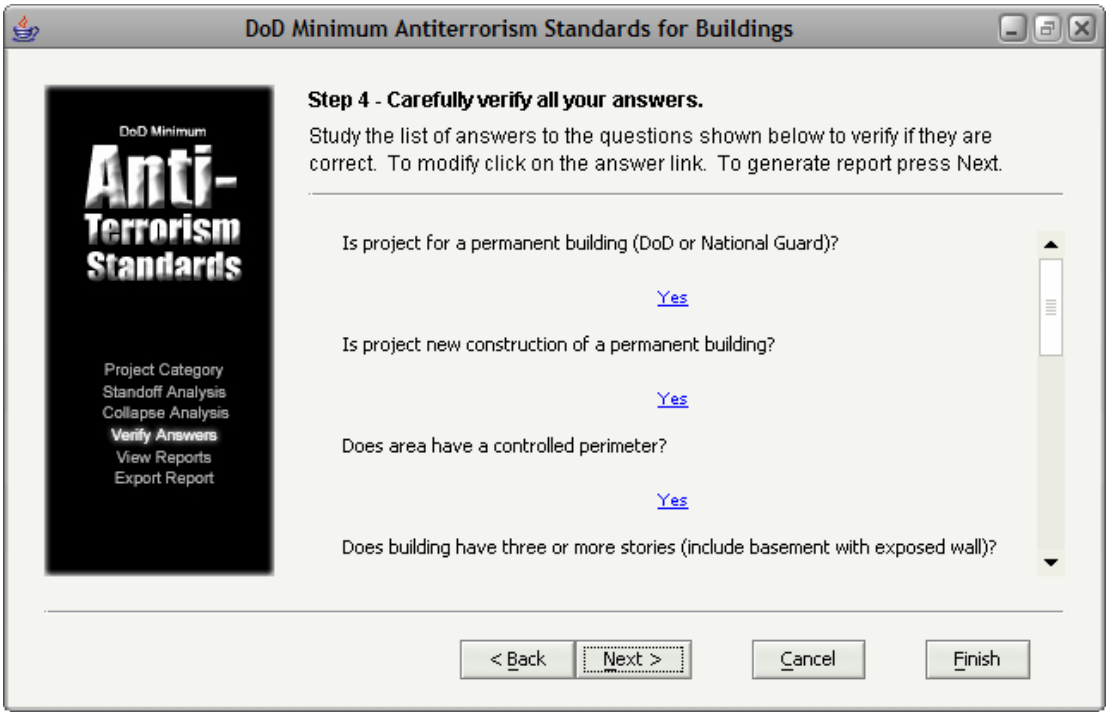


Figure 32. Verification panel.

5. Step 5 Panels

If all answers seem correct, simply click the **Next** button on the bottom of the Step 4 Panel to display the Project Category. From here, a series of Panels will display the requirements required for the particular project (Figure 33). The user then steps through the summaries for Site Planning, Structural Design, Architectural Design, and finally Electrical and Mechanical Design,

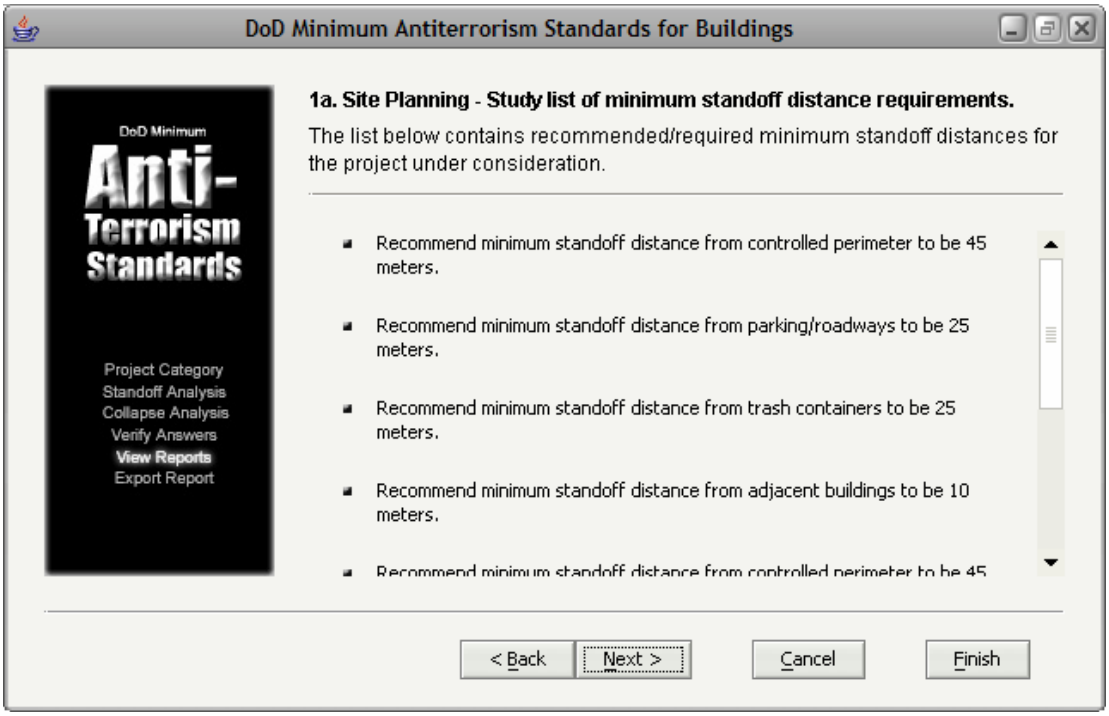


Figure 33. Sample report panel.

Once through all the summary panels, the user now has the ability to save the information to an Excel Spreadsheet on their computer to view the analysis at a later time (Figure 34).

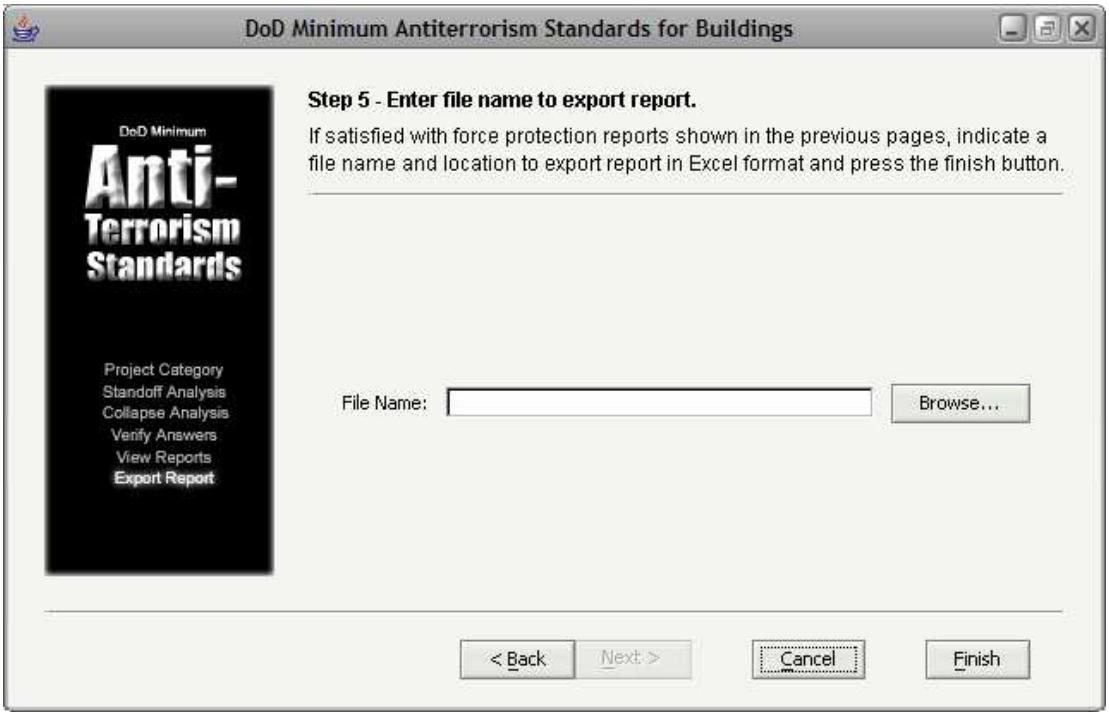


Figure 34. Export panel.

Implementation

Java Help

The MATSB Wizard contains a full-featured data driven help system that permits users to view documentation that provides assistance in relation to the application through the use of Sun Microsystems's Java Help (Figure 35). Java Help supports flexible display, compression and encapsulation of files, customization and extensibility, context sensitive help, merging capabilities and dynamic updating. Java Help software, when implemented properly into an application, can be displayed to the user upon an action performed, such as selecting specific text within the program. As in the Sustainable Designer Aid Wizard, the help system is composed of an individual frame with a split pane in the center that separates the navigator tabs on the left from the content panel on the right. Java Help includes help navigator views such as Table of Contents, Index and Full-Text Search while the content panel on the right displays the specified HTML page containing the useful, detailed, thorough help guidance.

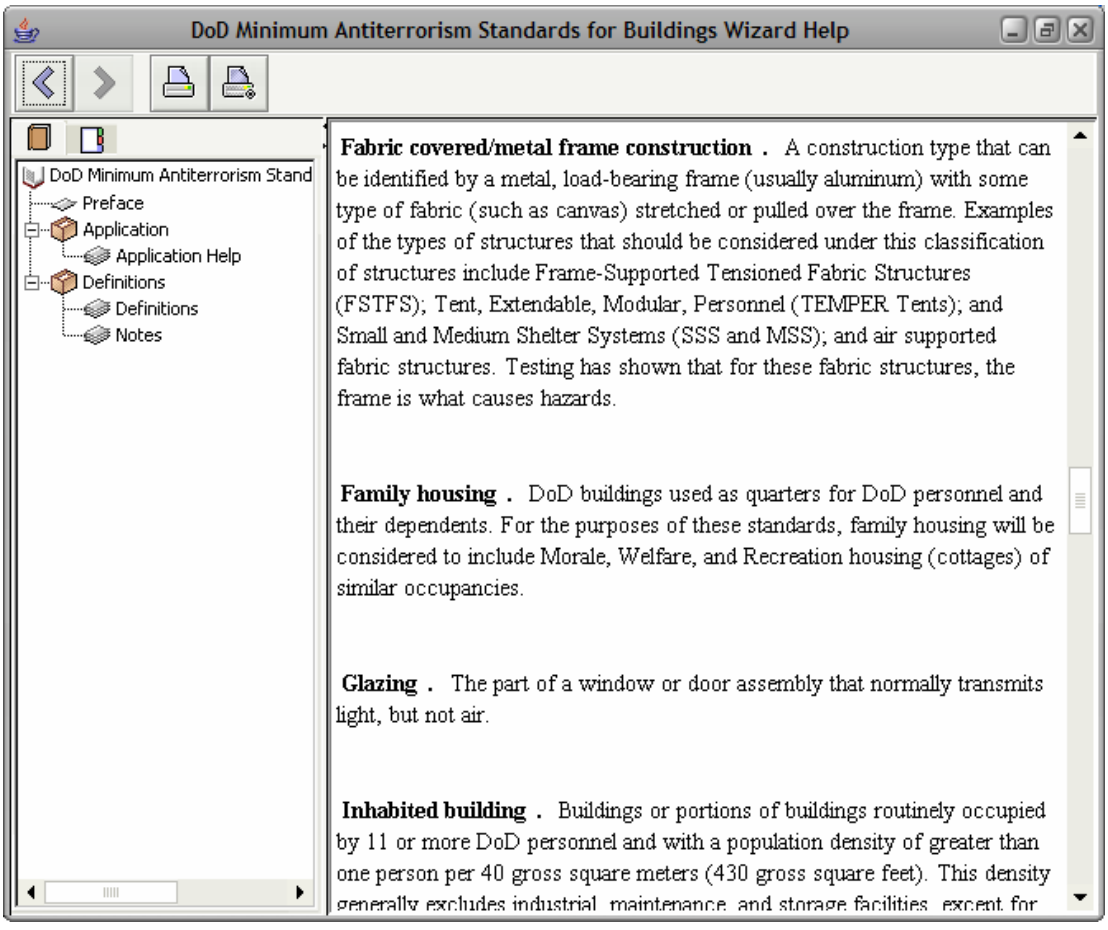


Figure 35. MATSB wizard help window.

Process Tree

As with the Parking Allowance and Site Area Calculation Wizard, the MATSB Wizard contains a compositional tree that outlines the sequence of the wizard's panels. An important attribute of the composition tree is that as the user runs through the application by means of pressing the **Next** and **Back** buttons, the title of the corresponding panel or section that is currently displayed is highlighted in the composition tree (Figure 36). This allows the user to have knowledge of the status of the sequence as well as the schematic model of the application. Navigational trees and guides provide the user with an organized and automated view of the wizard's sequence and sections.

Figure 36. Composition tree.

8 Wizard Framework

Introduction

Numerous studies suggest that a significant portion of resources and code (some suggest up to 80 percent) in a typical software development project are dedicated to the functionality required to execute the basic features in the program, rather than on features that are specific to the problem the software is trying to solve (referred to as "domain-specific" issues) (Gauven 2004). A software framework that generalizes out the common elements can assist in minimizing the amount of effort spent on periphery issues and allow the developers to focus more of their efforts on the specific problem being addressed. Wizards are well suited to this approach because of their many common components, same general approach of support provided, and also for the desirability of providing uniformity among the many individual implementations. An application framework, referred to here as the Wizard Framework, was developed for these reasons, and some basic issues related to its technical usage useful in the development of new wizards will be described. Please refer to the API documentation for more specific information.

To simplify the creation of wizards a series of classes were developed consisting of common components and functions that pertain to all wizards in general and grouped within a package titled the Generalized Wizard Framework. Moreover, to create a wizard, use and extension of the Generalized Wizard Framework classes should be organized inside another application-specific framework denoted as the Specific Wizard Framework.

Generalized Wizard Framework

Devising a common and generalized wizard framework simplifies the creation of wizards. The framework helps to minimize the effort required to create each individual wizard and ensures uniformity among the numerous applications. This framework is structured and equipped with functions programmed to handle the different components of the wizard.

The framework consists of two principal packages: the *BC Framework* package and the Wizard Framework package. The *BC Framework* package contains the classes that need to be extended and initiated to create a new wizard whereas the *Wizard Framework* package contains useful utility classes and other classes used by the *BC Framework* package.

The BC Framework Package

Currently (as it is still under development) the *BC Framework* Package consists of three single classes separated into two packages, *GUI (Graphical User Interface),* and *Data* (Figure 37). The *BCFramework.GUI* package consists of the *Basic Wizard* class and the *Basic Wizard Page* class while the *BCFramework.Data* package consists solely of the *Main Data* class.

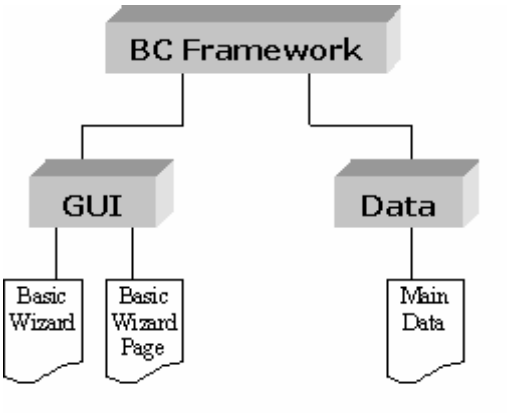


Figure 37. Diagram of BC framework package structure.

BCFramework.GUI

Foremost, to create a wizard and utilize the framework, one must extend the BC Framework's *Basic Wizard* class found inside the *GUI* package. In Java, a class that extends another inherits all the characteristics and behavior of the parent or, "super" class. Also extending a class allows for customization by means of overriding the methods provided by the parent class. For example, extending the *Basic Wizard* class by creating a new class called *Wizard Test Class* allows one to customize the wizard by setting the following attributes: title, author, description, version, icons, and other images. Moreover, from within the *Wizard Test Class*, the developer can initiate and pass the instances of each wizard page to the parent class by overriding the *add Wizard Page* method. The *Basic Wizard* class creates the wizard frame along with the button panel seen on the bottom (Figure 38). This class is responsible for, but not limited to, setting the frame title, managing the stack of wizard pages, and handling actions performed on the bottom navigational buttons.

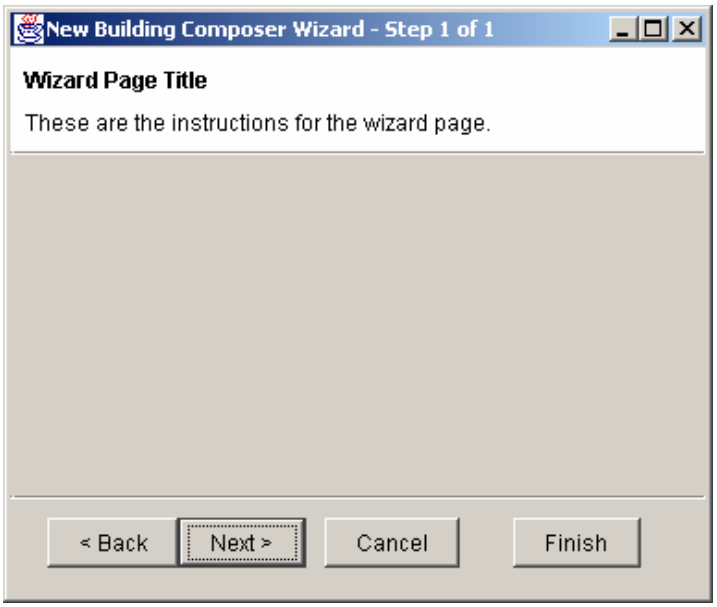


Figure 38. Screen capture of an empty wizard page.

To create a new wizard page, one must extend the functionalities of the framework's *Basic Wizard Page* in a new class called, for example, *Page One*. Customizing this class includes setting the page title, authoring instructions, and providing the contents of the page with the necessary fields. The *Basic Wizard Page* is an abstract class consisting of two abstract methods that must be defined and provided for upon extending the page. The first of these—the *start Page* method—is called when the page is first displayed, after the user clicks the Next button on the previous page. The second—the *finish Page* method—is called when the Next button on the current page is pressed. These methods should include the necessary operations or validations that must occur as the user proceeds from page to page. The *Basic Wizard Page* creates a page containing an instructions band on the top and an empty panel on the bottom to accommodate the contents of the page.

BCFramework.Data

The *Main Data* class within *BC Framework's Data* package is designed to take a simple properties text file and load the properties into a runtime hash table. Each text file, with file name extension "**.properties*," must be generated such that every key is associated with a value. The intended use for the *Main Data* class is to instantiate it by passing the URL of the corresponding properties text file. This class may be instantiated as many times as necessary, once for every properties file employed by the wizard application. During runtime, the application may query the instantiated class to retrieve any values from the resulting hash table. The values may be returned as strings, string arrays, integers, dou-

bles, or Booleans. Throughout the course of a session, values may also be saved into the hash table. On completion of the wizard, the hash table may be saved by writing out to the same, or another, specified properties text file. This mechanism of saving to a text file is the simple persistence mechanism employed by the wizard applications that make use of this framework.

The Wizard Framework Package

The *Wizard Framework* Package consists of three sub packages: *GUI, Data* and *Utils* (Figure 39).

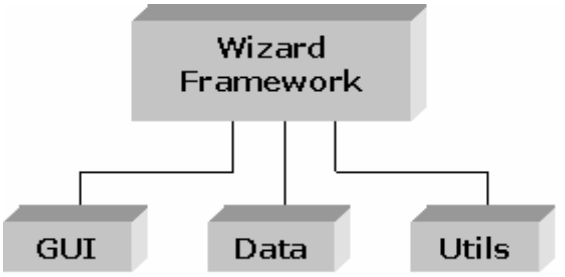


Figure 39. Diagram of the wizard framework structure.

The GUI Package

The *GUI* package contains classes to create specialized combined components as well as classes that assign specific behavior to individual components. For instance, the *Field Row* is a class that places a label and a text field side by side inside a panel with a *Box Layout Manager*. The *Box Layout Manager* maintains the components side by side in case the user resizes the window. This prevents the last component from being "bumped" to the next row. The code places the label on the panel, and then places the text field to the right of the label.

Another class within the GUI package is the *Component Row* class, which was modeled after the Facility Composer *Field Row* class. This class creates a component row with more flexibility than the *Field Row*. In other words, instead of requiring the component next to the label to be a text field, the class allows the component to be just about any control—a combo box, text area, password field, button, etc. In addition, the *Component Row* class contains a constructor method that supports an additional component, two more components, to be exact, in addition to the label. This is useful when a label, a text field, and another label is needed where the second label is used to display the units of the value within the text field (Figure 40).

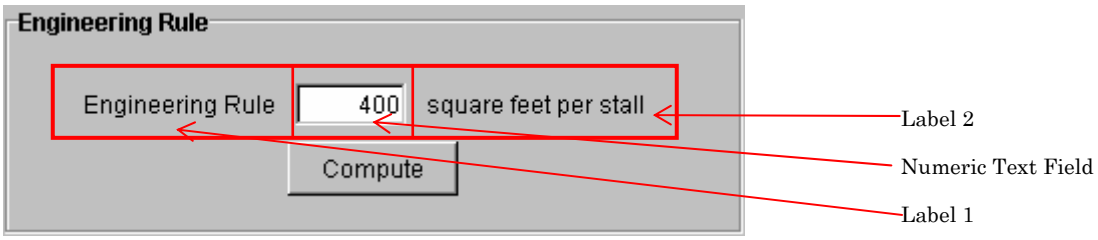


Figure 40. Example of a component row.

Another useful class in the *GUI* package is the *Numeric Text Field* class that comes in handy when text fields are restricted to accept only numeric values. A numeric text field accepts only numbers and beeps when illegal characters are typed in such as text, spaces, or symbols. Another benefit of the numeric text fields is the support of increments through the use of the arrow up and down buttons. Use of this component simply requires that the developer instantiate this class and set values through one of the many constructors with the option to set any or all of the following: the initial value, the text field column size, the minimum and maximum number of whole units, the minimum and maximum number of decimal units, or the minimum and maximum values allowed (or one can simply specify the Boolean to restrict the numeric text field to positive values instead).

In addition to the classes within the *GUI* package, this package also contains a sub package titled the *Persistent GUI* package. This package contains a number of classes aimed at persisting the properties of GUI window components such as location and size. The JFrameP abstract class is programmed to write out the values of the size and location of a javax.swing.JFrame in a *.properties text file to be saved within the user home directory. Subsequently, the next time the component is instantiated, it will possess the same properties as in the last session. To make use of this utility, one needs to extend this class and implement the two abstract methods: getInstanceID and getNameSpace. The JDialogP abstract performs the same function as the JFrameP class with a JDialog instead of a JFrame.

The *Data* package contains classes used by the *Main Data* class inside the *BC Framework's Data* package while the *Utils* package contains one very commonly used utility class called the *Utils* class. This class contains methods that read properties from a text file, copy a file to a particular project, replace characters with the desired character, display an error message dialog, and many more.

Specific Wizard Framework

When it comes to coding rather extensive and delicate applications, it is necessary to create a framework to organize, limit, and restrict your code in many ways. Besides employing a generalized common framework, the code that is specific to each individual wizard application should also be organized and maintained within a structured framework. Organizing and separating code simplifies and minimizes effort brought about by the procedures to make changes in an application. For instance, if the code is organized such that the *GUI* classes are separated from the *Domain* and *Data* classes, when the time comes to alter the appearance of the wizard, the developer only has to make changes to the *GUI* classes while the *Domain* classes, which retrieve data from the *Data* classes, need not be altered greatly (if at all).

The framework contains four main divisions: *Application*, *GUI*, *Domain*, and *Data*. The arrows in Figure 41 show the communication that is allowable between each division. The *Application* package is made up of a single class that contains the *main method* for the application. The A*pplication* communicates with the *GUI* and triggers the graphical components within that division to be generated and displayed. The *GUI* package is to contain all the classes and functions that make up the graphical design of the wizard along with the extended *Basic Wizard* class and each extended *Basic Wizard Page* class. The User communicates with the *GUI* components by pressing a back or next button, selecting an item from a scroll list, entering a value into a text field and so forth. Subsequently, the *GUI* division communicates with the D*omain* division, alerting it of the change. The *Domain* contains methods that encompass all the engineering rules and methodologies used to calculate and compute the necessary data required by the application, in other words, application-specific methods.

This division also contains all the functions that obtain the information needed by the wizard from the *Data* division. The *Domain* is called upon by the *GUI*, which in turn calls upon the *Data* to obtain the information specified by the *GUI*. or instance, the *GUI* contains the code to build and display a frame on the screen monitor. Of course, the *GUI* can only be triggered by the *Application* that contains the *main method*. The GUI also contains a function to set a title for that frame. Instead of having that text title "hard coded" inside our *GUI* class, it is necessary to separate and store that data into a properties text file that is handled by the *Data* division. Yet, one of the restrictions set forth by the framework

is that the *GUI* cannot communicate directly with *Data* (cf. Figure 41). Therefore, it is necessary for the *Domain* to intervene and communicate with *Data* instead. Consequently, the *Domain* must have a function called *getFrameTitle()* that calls on the Data's function called *getString*[*]*(key)*. Finally, the Data passes that string title back to the *Domain*, which in turn returns that same string to the *GUI* division, which then finally displays that string title on the frame.

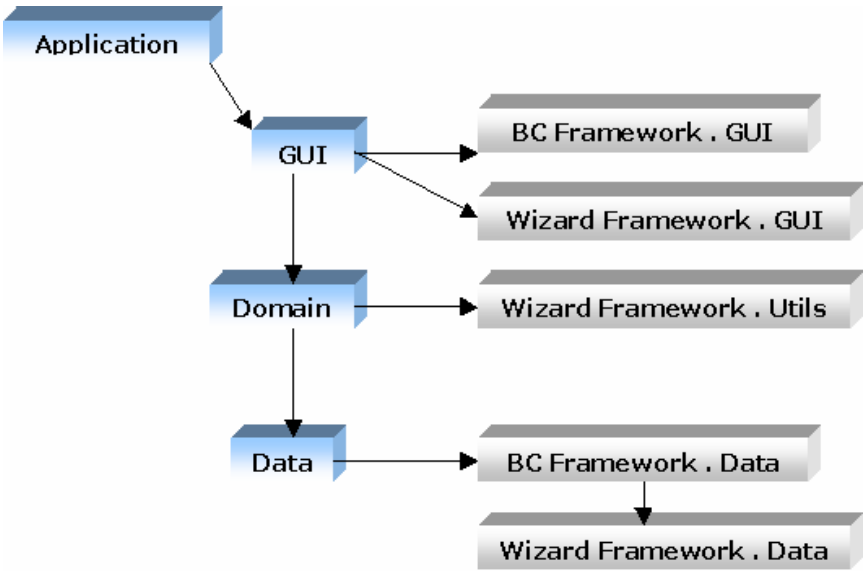


Figure 41. Diagram of specific wizard framework structure.

[*] A string contains characters such as letters, numbers, symbols, etc. String is used to denote anything that is stored as text in Java.

9 Implementation Example Using the Wizard Framework

This chapter provides developers a step-by-step guide on how to implement the generalized wizard framework described in the previous chapter. For demonstration purposes (and due to its simplicity), this chapter will use the implementation of the *Parking Allowance and Site Area Calculation Wiza*rd as a representative example.

Diagram Wizard Logic

The first step in planning a wizard, after having acquired all the documentation necessary, is to determine the purpose of the program and identify the types of output and input necessary. A simple diagram can assist in organizing the logic of the application. Figure 42 shows an initial logic diagram for the parking wizard; the purple boxes indicate user input, the blue boxes represent output.

The next step is to brainstorm the procedures necessary to complete the program. For instance, the parking wizard should provide the user with a list requiring the user to choose a single building type. Upon the user's selection of a building type, the wizard is to obtain values for percentages of various criteria specific to the building type chosen and required to compute the number of parking stalls as discussed in Chapter 3. This computed value of parking stalls is to be adjusted by the user if so desired. Following this, the minimum number of accessible parking stalls is computed based on the adjusted number of parking stalls. Finally, the wizard is to obtain the engineering rule, or "rule of thumb," which will be used to calculate the approximate parking area.

Drawing from the initial diagram and the procedures outlined above, identify the fundamental steps that are essential to achieve the computations necessary to produce the required output values. This can be done by generating a more elaborate and detailed logic diagram (e.g., Figure 43).

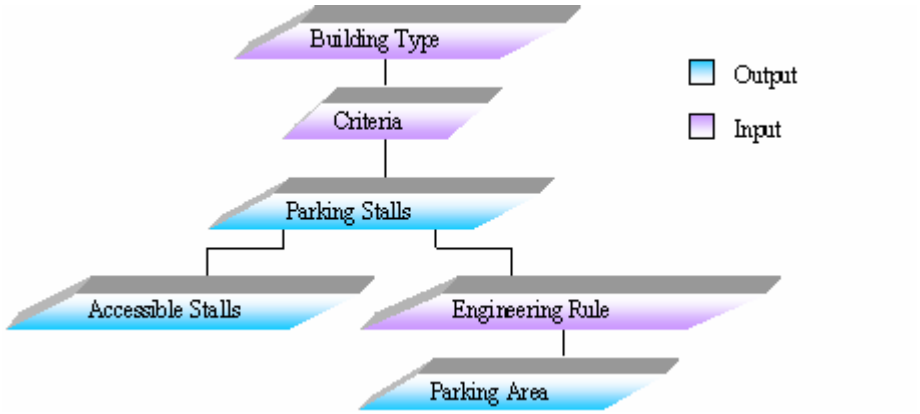


Figure 42. Initial logic diagram.

Plan Wizard Sequence

Using the detailed logic diagram created for the wizard, identify and isolate primary tasks by assigning each task to a single wizard page as stipulated by the Inductive User Interface guidelines mentioned in Chapter 2. For each wizard page, mock up prototype wizard pages by selecting proper controls to achieve the primary task, making sure that the contents of each page suits the task appropriately.

Code Specific Wizard Framework

To begin adapting the application code to the wizard framework, separate the code into the four main packages outlined by the Specific Wizard Framework guidelines in the previous chapter: (1) Application, (2) GUI, (3) Domain, and (4) Data. Bear in mind that the *Application* package will call on *GUI* to start the program by initializing and populating the necessary components. Furthermore, the *GUI* will communicate with the *Domain* to obtain all the data, which will be retrieved from the *Data* package. Note that there is no communication between the *GUI* and *Data* packages.

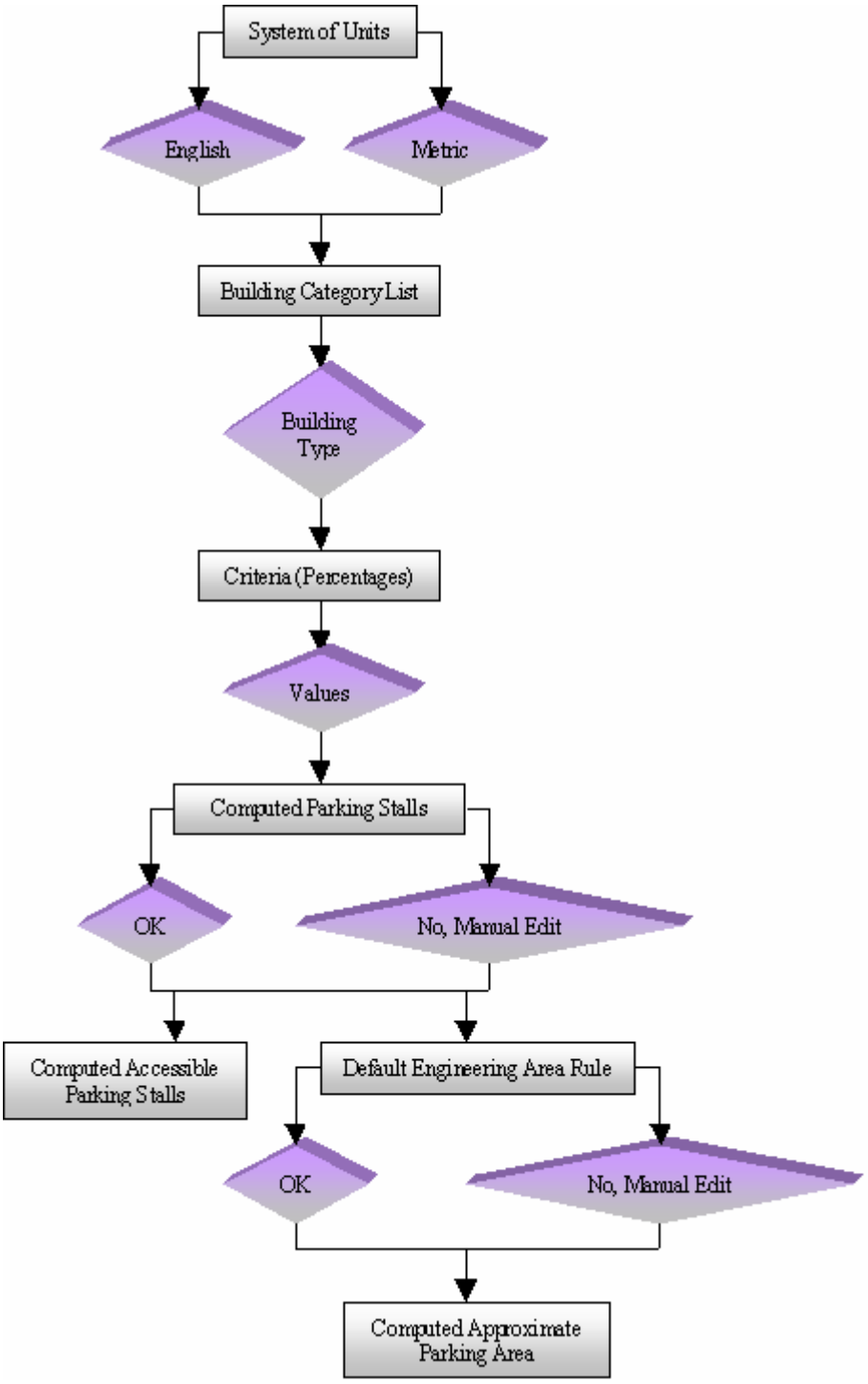


Figure 43. Detailed logic diagram.

Application Package

The *Application* package will contain a single class called the *Application* class. The objective of the Application class is to start up the program, in other words this class contains the *main* method. The main method will create a new instance of the class *Parking Wizard*, which will be explained with more detail ahead. Also notice how in the figure below, the main method contains a line of code that enables an aqua theme "Look and Feel" for the application.* Observe the Aqua Theme Look and Feel in the screen captures of the wizard that follow.

```
/**
 *  Description of the Method
 *
 *@param  args  Description of Parameter
 */
public static void main(String args[]) {

        try {
                URL theme = Application.class.getClassLoader().getResource("lib/aquathemepack.zip");
                SkinLookAndFeel.setSkin(SkinLookAndFeel.loadThemePack(theme));
                SkinLookAndFeel.enable();
        }
        catch (Exception e) {
                e.printStackTrace();
        }

        UIManager.put("ClassLoader", Application.class.getClassLoader());
        UIManager.getLookAndFeelDefaults().put("ClassLoader", Application.class.getClass());

        ParkingWizard frame = new ParkingWizard();
        frame.show();
}
```

Figure 44. Screen capture of the main method within the Application Class.

GUI Package

The *GUI* package will contain all the classes necessary to create the wizard and the corresponding wizard pages. However, remember to commit to the framework's limitations and boundaries in the sense that *GUI* is responsible only for generating the graphical components of the wizard and does not deal with the logic of program and does not store data values, as this is the task of the *Domain* and the *Data* packages.

* http://www.l2fprod.com/index.php

Extending the Basic Wizard Page

Start by creating the first wizard page. The introduction page of the parking wizard will consist of an HTML page, which is populated with the introductory text, loaded and centered on the wizard page. In addition to this, the panel will contain a check box along the bottom to give the user the option to never display the introduction page of the wizard in future sessions (Figure 45).

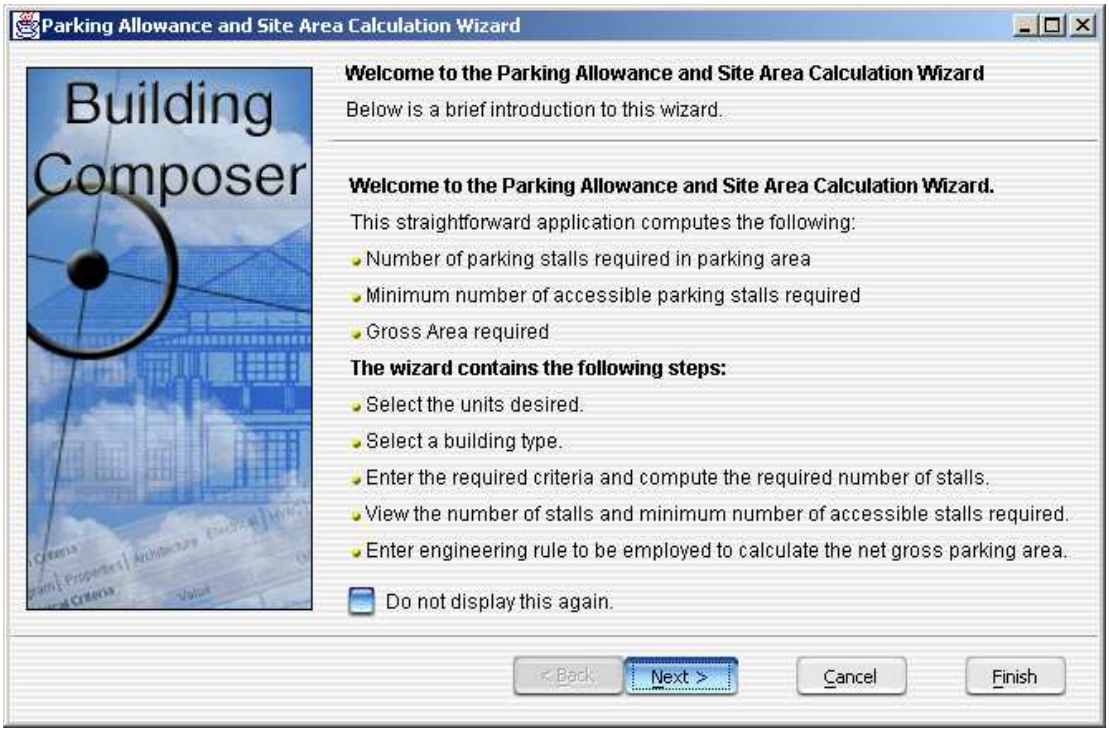


Figure 45. Parking wizard's introduction page.

To make a wizard page, create a new class titled, for example, *Intro Page* that extends the *Basic Wizard Page* class imported from within the BC framework's GUI package. Moreover, make use of the methods within the *Basic Wizard Page* class that allow one to set the instructions bar along the top of the page, the page title and the instructions. Finally, to assign the contents of the page make a call to the parent class method whose objective is to add the content panel either centered or anchored north (Figure 46).

```
import bcframework.gui.BasicWizardPage;
public class IntroPage extends BasicWizardPage {
   /**
    *  Constructor for the IntroPage object
    */
   public IntroPage() {
      this.setInstructionsBar(true);
      this.setTitle("Welcome to the … Wizard”);
      this.setInstructions("Below is a brief introduction to this
wizard.");
initPage();
super.addContentPageCenter(base);
   }
   /**
    *  Description of the Method
    */
```

Figure 46. Sample code for Intro page.

Recall that the *Basic Wizard Page* is an abstract class, which requires the implementation of two particular abstract methods: the *start Page* method and the *finish Page* method. The *start Page* method can consist of a call to the *Domai*n package, which can in turn retrieve the default or user data check box state from the *Data* package (explained later in this chapter). Finally, the *finish Page* method, which is called when the user clicks on the Next button to proceed to the following page, can also contain a call that passes the final and actual check box state to the *Domain* package, where it can be stored in the appropriate data hash table, the runtime application memory (also explained later in this chapter). Then create all the wizard page classes, one at a time, following this procedure.

Extending the Basic Wizard Class

After creating all the wizard pages, it will be necessary to create the all-encompassing wizard frame by extending the *Basic Wizard* class in a new class called, for example, *Parking Wizard*. As mentioned in the previous chapter, the *Basic Wizard* class creates the wizard frame along with the button panel on the bottom. This class is responsible for (but not limited to) managing the stack of wizard pages and handling actions performed on the bottom navigational buttons. One can implement this class by either extending the class or by making a new instance of it. Furthermore, it is possible to customize the wizard by indicating the different attributes such as: wizard/frame title, description, version, icons, images, etc.

The *Basic Wizard* class accepts instances of the *Basic Wizard Page* class through the use of the *add Wizard Page* method (Figure 47). The *Basic Wizard* class encapsulates these pages within its code and populates the center panel with each one. The center panel is controlled by the *Card Layout Manager*, which manages the various panels passed in just like a stack of cards, displaying one at a time. Future modifications of the wizard framework will support indicating the *Basic Wizard* class whether the page is to be displayed or not during runtime, by the use of a Boolean parameter in the *add Wizard Page* method.

```
public class ParkingWizard extends BasicWizard {

        /**
         *  Constructor for the frame object
         */
        public ParkingWizard() {

                //Set Logo on the west panel
                ImageIcon icon = Utils.getIcon(Initialization.getInstance().getValue(Initialization.LOGO_IMAGE));
                JLabel imageL = new JLabel(icon);
                JPanel imageP = new JPanel(new GridBagLayout());

                imageP.add(imageL, new GridBagConstraints(0, 0, 1, 1, 0.0, 0.0
                                , GridBagConstraints.CENTER, GridBagConstraints.NONE, new Insets(10, 10, 10, 10), 0, 0));
                imageL.setBorder(BorderFactory.createLineBorder(Color.black, 1));
                this.setWestComponent(imageP, BorderLayout.CENTER);

                //Set variables for wizard
                setTitle(Initialization.getInstance().getValue(Initialization.FRAME_TITLE));
                this.setDescription(Initialization.getInstance().getValue(Initialization.WIZARD_DESCRIPTION));
                this.setVersion(Initialization.getInstance().getValue(Initialization.WIZARD_VERSION));

                //this.setResizable(false);
                IntroPage page1 = new IntroPage();
                UnitsPage page2 = new UnitsPage();
                BuildingPage page3 = new BuildingPage();
                CriteriaPage page4 = new CriteriaPage();
                StallsPage page5 = new StallsPage();

                //Add the wizard pages to the wizard
                this.addWizardPage(page1);
                this.addWizardPage(page2);
                this.addWizardPage(page3);
                this.addWizardPage(page4);
                this.addWizardPage(page5);

                this.pack();
                this.setSize(this.getWidth() + 5, this.getHeight() + 10);
                this.centerWizard();

        }
}
```

Figure 47. Sample code from the parking wizard class.

Domain and Data Packages

The *Domain* package is responsible for bridging the gap between the *GUI* and the *Data* packages. Therefore the *Domain* package will be the one responsible for retrieving the bits of information needed from the *Data* to return it to the *GUI* package to be displayed on the corresponding graphical components.

The Data Package

As mentioned in the previous chapter, the wizard framework operates on a simple properties text file persistence mechanism. All the data necessary to execute the program can be provided for in the form of text files containing the **.properties* extension. Each bit of information will be populated into the text file in key-value pairs where each value is associated to a unique key ID.

The properties text files used can be saved in a separate directory called the *Data* package. The most common properties text files used by simple and small applications are the following: System Data, Domain Data, and User Data (e.g., Figure 48). The conventional use of the System Data properties text file is for links, URL's, default system values, and other related data. Likewise, User Data is generally used to save values entered or selected by the user during each wizard session. Finally, the Domain Data properties text file, the most replete, contains the data that will be populated into the wizard during runtime such as titles, text, values, etc. At runtime, each of these text files will need to be represented as a new instance of the *Main Data* class found within the BC framework's Data package. When creating a new instance of the *Main Data* class, the relative or full path of the text file assigned is passed in through the constructor.

```
#Domain Properties File

frameTitle = Parking Allowance and Site Area Calcu
description = This application is programmed to ca
version = 1.0

#================================================

#Parking Tree Text and Properties

parkingTreeRoot=Parking Planning Area
parkingTreeParents=Building Type Panel;Criteria Pa

parkingTreeChildren1=Building Type
parkingTreeChildren2=Criteria
parkingTreeChildren3=Parking Stalls;Accessible Sta
parkingTreeChildren4=Rule of Thumb;Gross Area

#================================================

#Units Page

unitsTitle = Choose the system of units.
unitsInstructions = Select the desired choice of u
unitButtonLabels = English System, Metric System

#================================================

#Introduction Page
```

Figure 48. Portion of the parking wizard's domain data properties text file.

These instances can be initiated within the *Domain* package, as will be explained in the following section. Upon initialization, the *Main Data* class retrieves all the information within the assigned text file and stores it inside a hash table. This hash table serves as the "runtime memory" of the program. The *Main Data* class also manages the information inside the hash table through the use of its methods that retrieve String values, Boolean values, integers, or doubles as well as those methods that store new values or alter existing ones. Use of this class will also be explained in the following section.

Guidelines of customary wizard behavior stipulate that values entered or selections made by the user throughout a particular session should only be saved upon the completion of the application, in other words, when the user has pressed the Finish button. The *Main Data* class is equipped with a method that saves all the values from the hash table back to the properties text file when indicated, the *save Hash table* method. One can make use of this method inside the *Domain* package, or where the *Main Data* instances were initialized and stored, at the close of a completed application if and only if the user pressed the Finish button indicating approval of the output values of the program.

The Domain Package

The Parking Wizard's *Domain* package contains a solitary class called the *Initialization* class, which makes use of a singleton instance.* The code for this class creates three new instances of the *Main Data* class, one for each of the following properties text files: System Data, Domain Data, and User Data.

Following that, an instance of the *Nested Primitive Database* class is created and assigned the three instances of the *Main Data* class initialized previously (Figure 49). The *Nested Primitive Database* class is imported from within the Wizard Framework's Data package and is responsible for recursively looking through the three specified instances of data for specified values. For example, the *Initialization* class contains the *get Check Box State* method, which retrieves a Boolean value from a properties text file. However, the value containing the same key may be found inside the System Data properties text file as well as it can be found in the User Data properties text file.

*For more on singleton instances, see: When is a Singleton not a Singleton? and Implementing the Singleton Pattern.

```
private static MainData SYSTEM_DATA;
private static MainData DOMAIN_DATA;
private static MainData USER_DATA;
private static NestedPrimitiveDatabase DATA;

   /**
    *  Constructor for the Initialization object
    */
   public Initialization() {

      this.SYSTEM_DATA = new
MainData(Parking/data/systemdata.properties");
      this.DOMAIN_DATA = new
MainData(Parking/data/domaindata.properties");
      this.USER_DATA = new MainData(Parking/data/userdata.properties");

      DATA = new NestedPrimitiveDatabase(new
PrimitiveDatabaseInterface[]{USER_DATA, SYSTEM_DATA, DOMAIN_DATA});
```

Figure 49. Creation of Nested Primitive Database class.

The System Data file contains the default value while the User Data contains the value saved by the user in a previous session of the application. Therefore, if the check box state key and value is found within the User Data properties text file, the application should disregard the value found within the System Data properties text file. Consequently, this is the reason why the instance of the *Main Data* class for the User Data was assigned before the System Data instance in the *Nested Primitive Database*, this way the database searches for the key-value pair within User Data before it moves on to System Data.

The Domain's *Initialization* class is mostly equipped with methods that get data from the *Nested Primitive Database* instance, or methods that set user data. Methods that set data are those that store particular values to the "runtime memory" hash table (Figure 50).

Saving user data unto runtime hash table:

```
public void setParkingStalls(String parkingStalls) {

      this.USER_DATA.setString(this.PARKING_STALLS,parkingStalls);

}
```

Figure 50. "Runtime memory" hash table.

At the end of the application, when the user has pressed the Finish button, the user data hash table can be saved back onto the properties text file, also known as the "persistent memory."

Saving the hash table to the properties text file:

```
public void saveHashtable(String filename) {

          this.USER_DATA.saveHashtable(filename);

}
```

Figure 51. "Persistent memory" hash table.

Create a Wizard Gallery

Package the Code

After the wizard application has been developed and properly tested, it is necessary to package the code inside a .JAR file and place it within a new directory called, for instance, "Components" inside Facility Composer's code. Facility Composer will then identify all the JAR files within this directory as individual applications and properly execute each one when necessary. Additionally, a wizard gallery can be generated to contain all the individual applications within Facility Composer.

Make Use of the Basic Wizard Gallery *Class*

A new class, currently under development, is called the *Basic Wizard Gallery* class (Figure 52). Any principal software program, which contains a number of small applications, can make use of this class. The purpose of this class is to create a gallery that will contain a list of the individual applications available to the user within the main program. The gallery will display information about each application such as the title, description, version, homepage, etc., and will also provide a control to start up the selected application. For example, Facility Composer can have a wizard gallery frame that lists all the wizard applications that it employs.

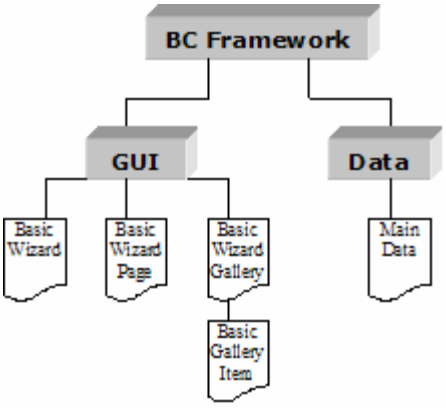


Figure 52. New basic wizard gallery classes within the BC framework.

To make use of this class, one simply needs to create a new instance of the *Basic Wizard Gallery* class by passing a single parameter that will contain the pathname of the directory "Components." Recall that the directory Components will contain all the .JAR files of the individual applications employed by the main program. The purpose of this class is to look under the specified directory and create a *Basic Gallery Item* for each .JAR file, or application, present in the directory.

Each JAR file should contain a properties text file saved with the following filename: "componentdata.properties." The instance created of the *Basic Wizard Gallery* will search within each JAR file for this properties text file and obtain the following information from each: title, description, comment for tool tip, version, web site link, and icon pathname (Figure 53). Figure 54 shows a screen capture of the prototype wizard gallery generated for Facility Composer.

```
#Component Data

title = Parking Allowance and Site Area Calculation Wizard
description = This application determines the number of parking stalls, access
tooltip = Parking Allowance and Site Area Calculation Wizard
version = 1.0
link1 = <a href=\"http://bc.cecer.army.mil/bc/wizards.jsp\">Building Composer
icon = classes/Program/GUI/img/parking1.gif
```

Figure 53. Screen capture of Component Data Properties Text File.

Figure 54. Wizard Gallery for Facility Composer.

10 Summary

Facility Composer addresses many of the problems associated with the decentralized, non-computationally explicit, ad-hoc definition, distribution, and utilization of design criteria. However, customer feedback regarding the tools designed to assist designers working with the criteria shows that design practices commonly vary by regional, organizational, or facility-specific differences. Such factors can change the priority of certain criteria, or require that certain (possibly identical) criteria be computed upon very differently. This report outlined the concept and application of "Wizards." This work also developed a framework for the efficient development of wizards, as well as a sample set of wizards which are now available as part of the Facility Composer system. Most importantly, the wizard approach was developed to provide design automation support that accommodates common variations in design practices, and also to provide modularized extensibility.

Bibliography

Government Publications

Technical Instructions (TI) 800-01, *Design Criteria* (Headquarters, Department of the Army [HQDA], 20 July 1998).

Unified Facilities Criteria (UFC) 4-171-05, *Design: Guide for Army Reserve Facilities* (Headquarters, Department of the Army [HQDA], 01 November 2003).

ADA Accessibility Guidelines for Buildings and Facilities (ADAAG) (amended through September 2002), accessible though URL: http://www.access-board.gov/adaag/html/adaag.htm

Nongovernment Publications

2000 International Building Code (International Code Council, Inc., Falls Church, VA, 2000).

Cockburn, A., *Agile Software Development* (Addison-Wesley, Boston, MA, 2002).

Gauven, Simon, and Trevor Smedley, "Vivid: A Framework for Creating Visual Programming Languages," *Proceedings of the 12th International Conference on Intelligent and Adaptive Systems and Software Engineering (IASSE-2003), July 9-11, 2003, San Francisco, CA* (International Society for Computers and Their Applications, Cary, NC, 2003), pp 223-226.

Highsmith, J., *Agile Software Development Ecosystems* (Pearson Education, Boston, MA, 2002).

Leffingwell, D., and Don Widrig, *Managing Software Requirements: A Use Case Approach*, 2d ed. (Addison-Wesley, Boston, MA, 2003).

Tidwell, Jennifer, *Common Ground: A Pattern Language for Human-Computer Interface Design* (2003), available through URL: http://www.mit.edu/~jtidwell/common_ground.html

Weigers, Karl E., *Software Requirements*, 2d ed. (Microsoft Press, Seattle, WA, 2003).

Appendix A: IUI Precedent Studies

Below are three additional examples of Inductive User Interface implementations.

Wise for Windows Installer 5.2

Wise for Windows Installer is the easy way to create professional, reliable installations for Windows Installer. It provides you with a complete installation toolkit designed specifically to enable compliance with Microsoft Windows Installer technology.

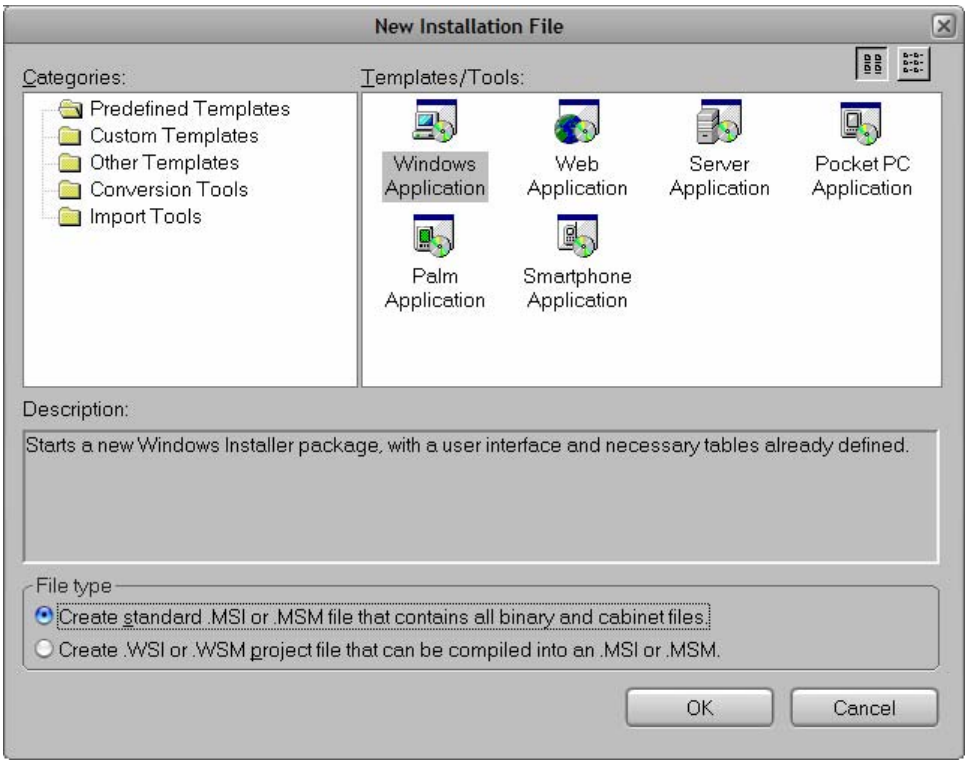


Figure A1. The "New Installation File" dialog box.

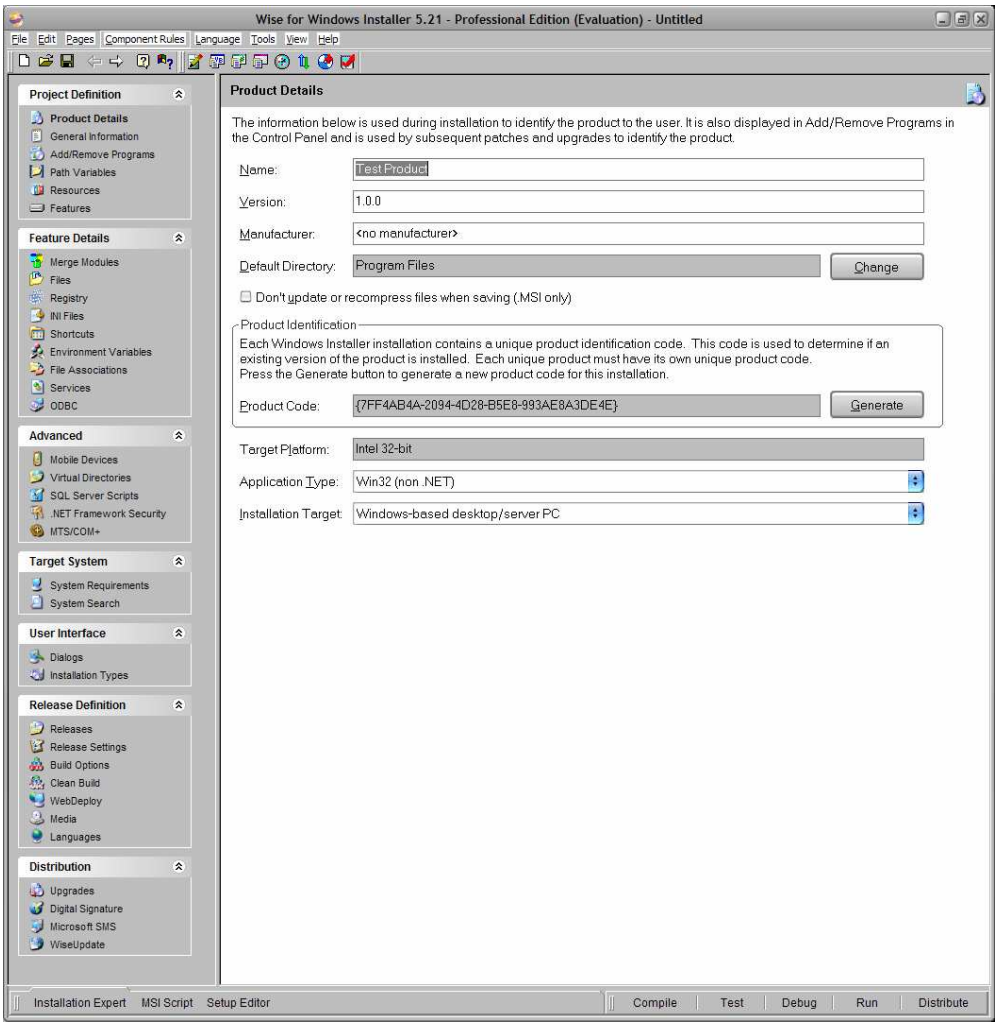


Figure A2. The Wise Installation Expert.

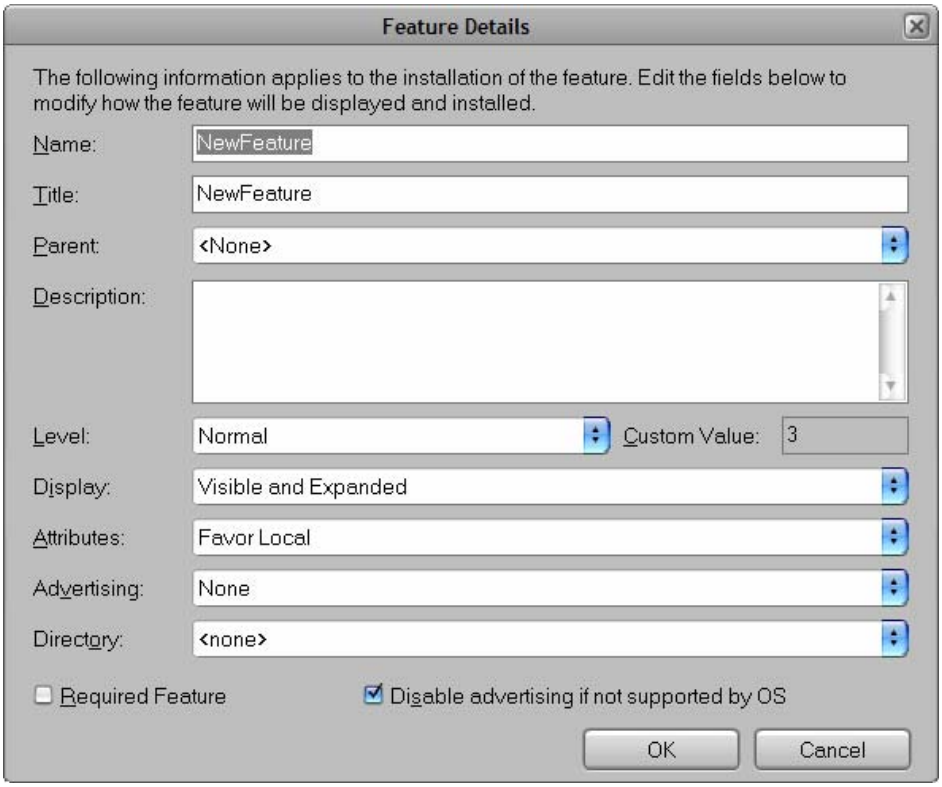


Figure A3. "Add New Installation Feature" dialog box.

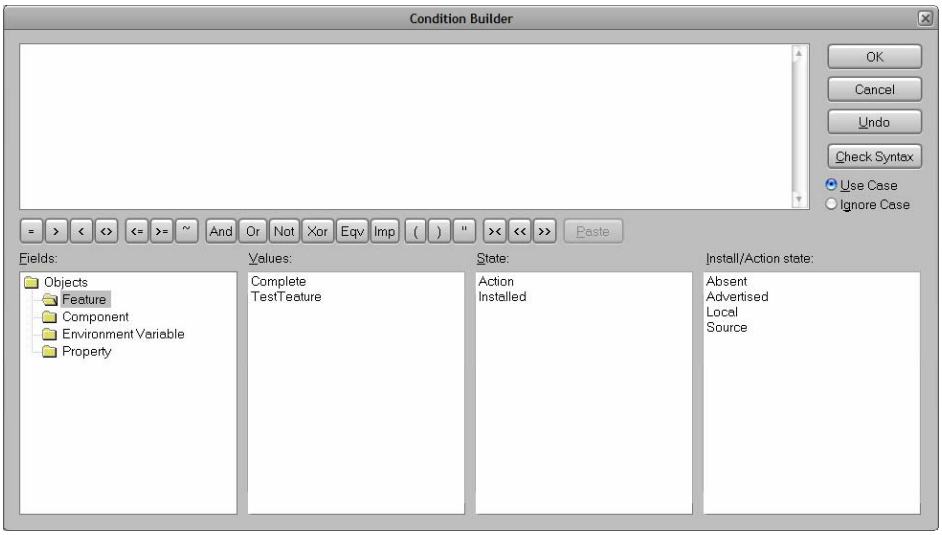


Figure A4. "Condition Builder" dialog box.

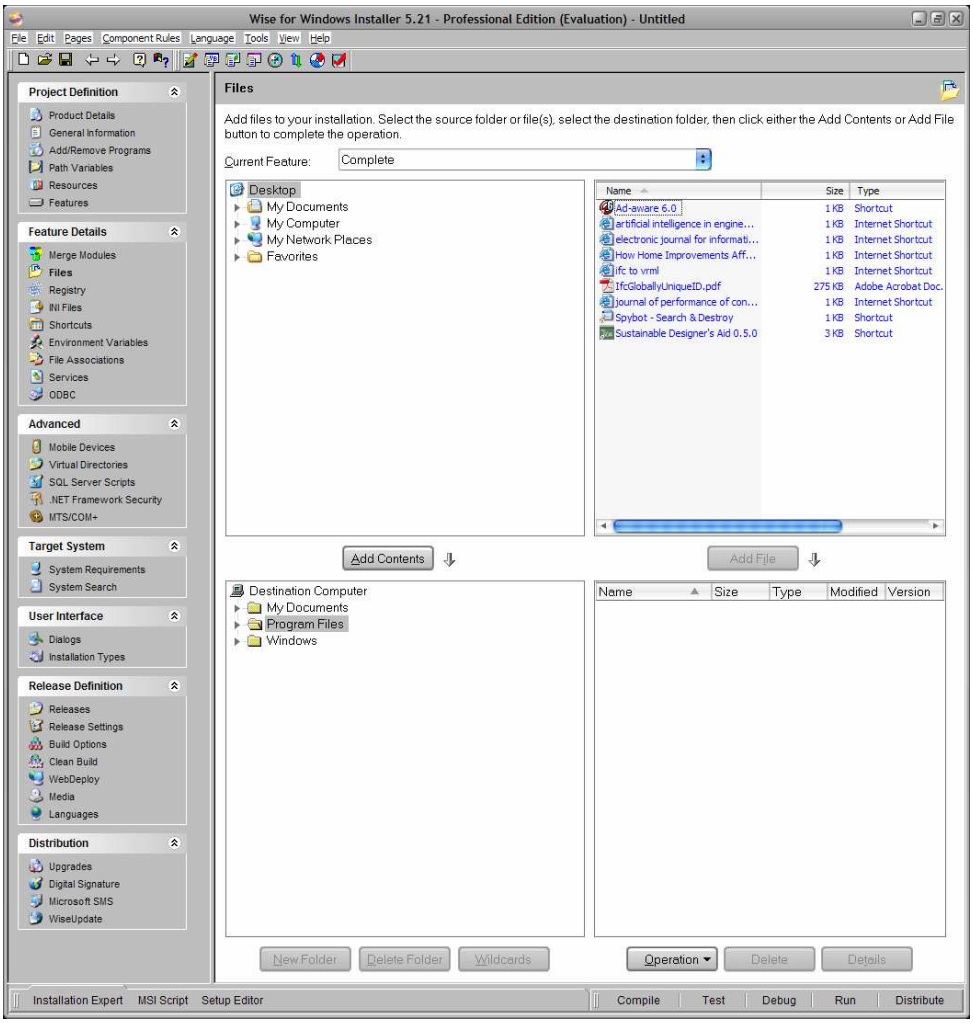


Figure A5. Adding Files to Features.

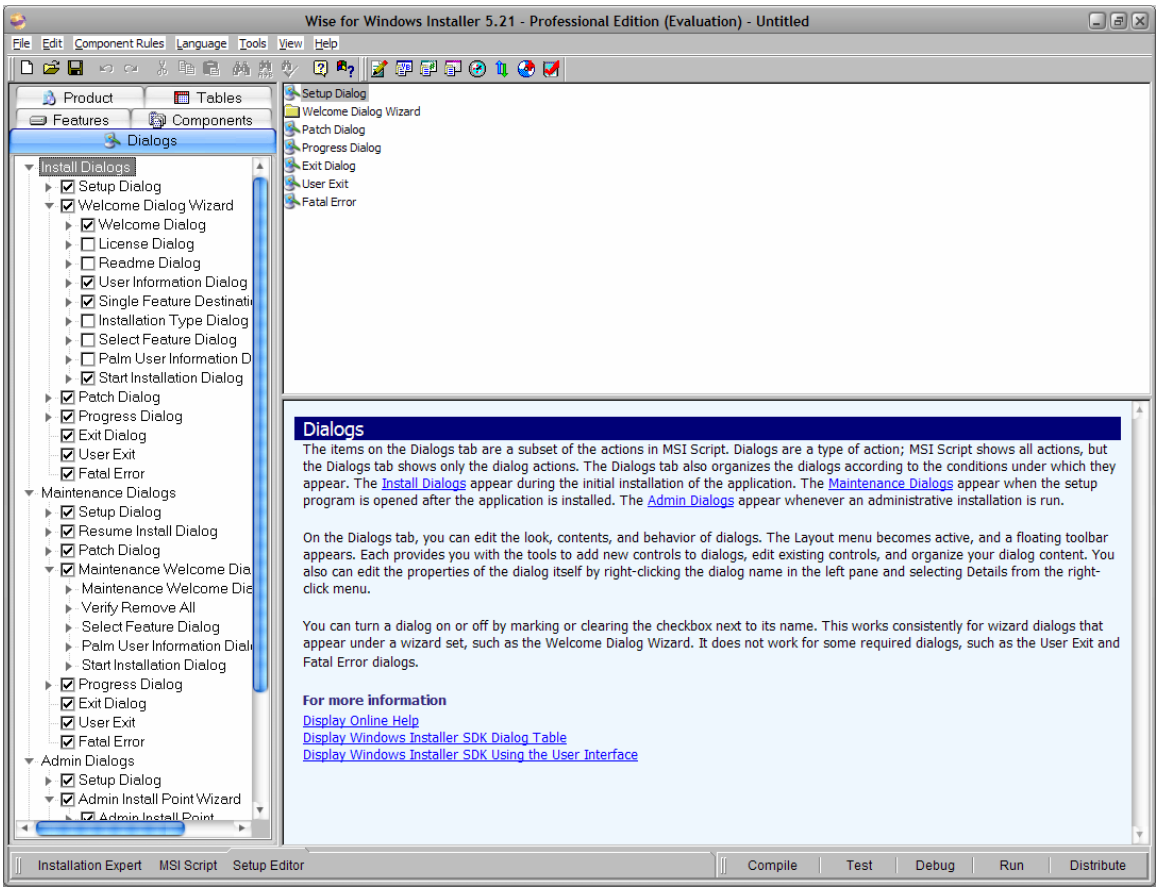


Figure A6. "Setup Editor" dialog.

Microsoft Update Web Application

Microsoft Windows Update is a perfect example of a web page as an inductive/wizard interface. The web page allows you to get the latest updates for your system by scanning your computer and providing a list of updates tailored for their system.

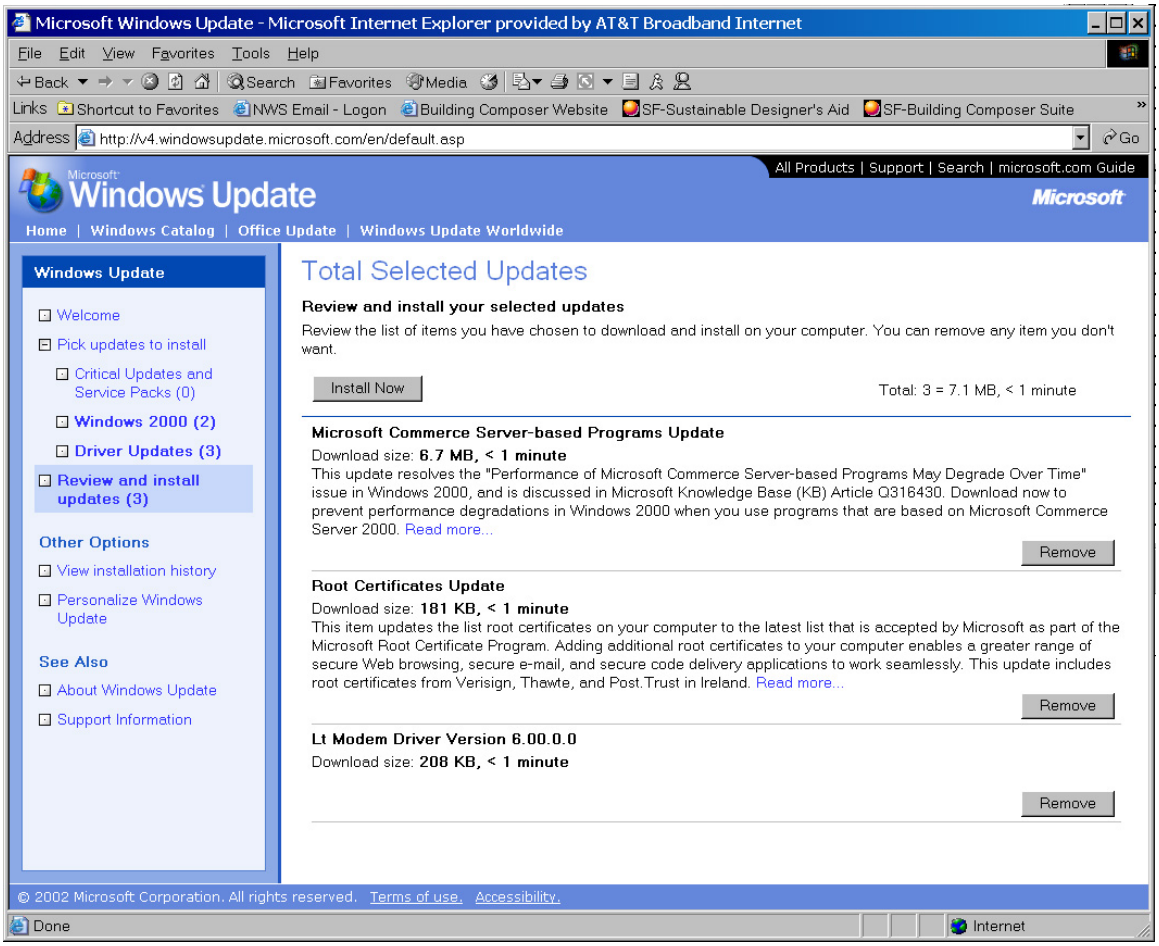


Figure A7. "Review and Install Updates" screen.

JDiskReport

JDiskReport is a Java utility developed by JGoodies, a product development, software consulting, and design company. It enables users to understand how much space files and directories take up on their disk drives.

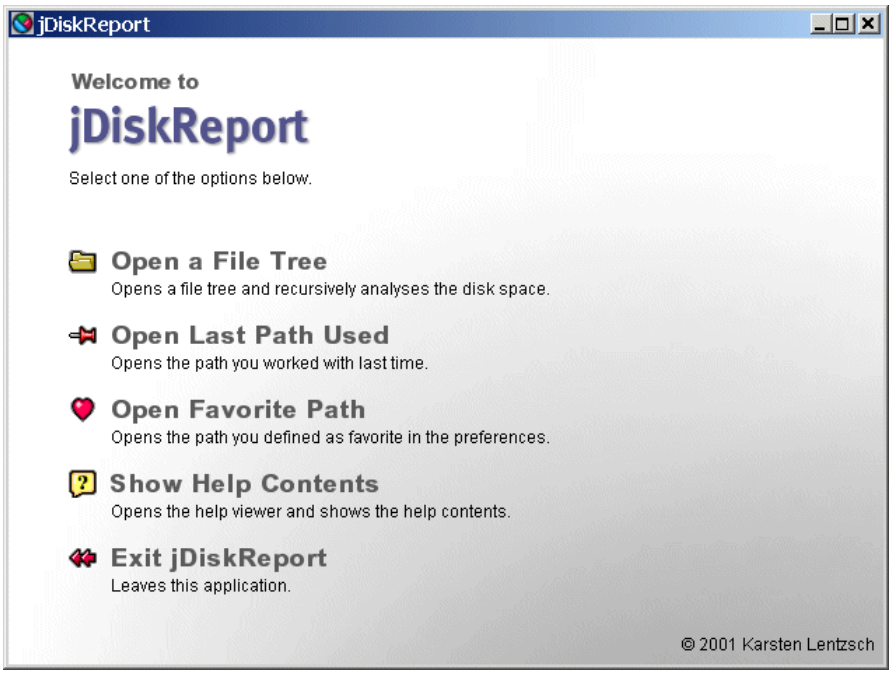


Figure A8. "jDiskReport Welcome" screen.

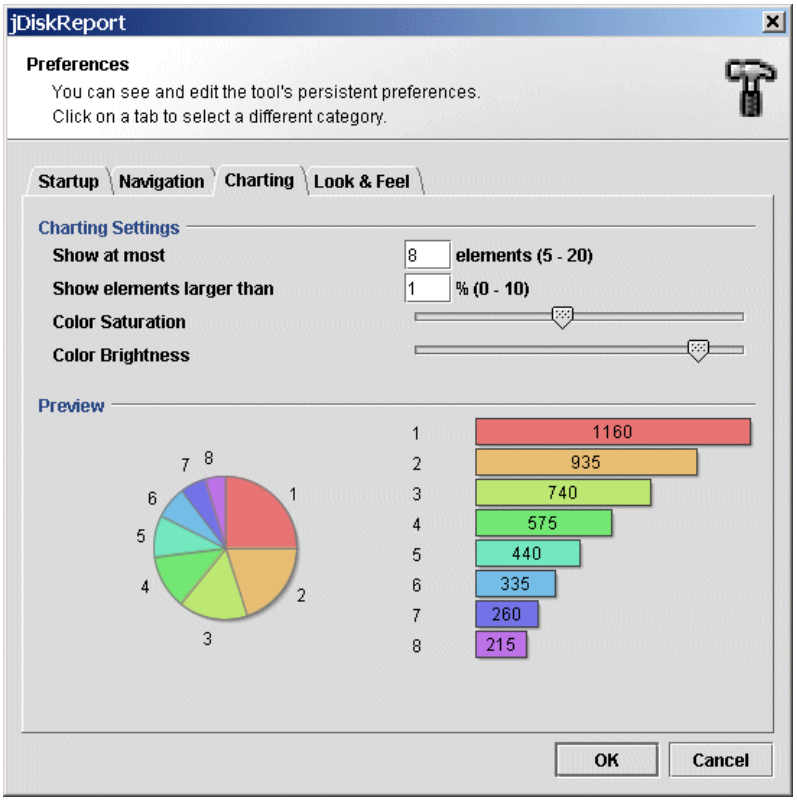


Figure A9. "jDiskReport Preferences" dialog.

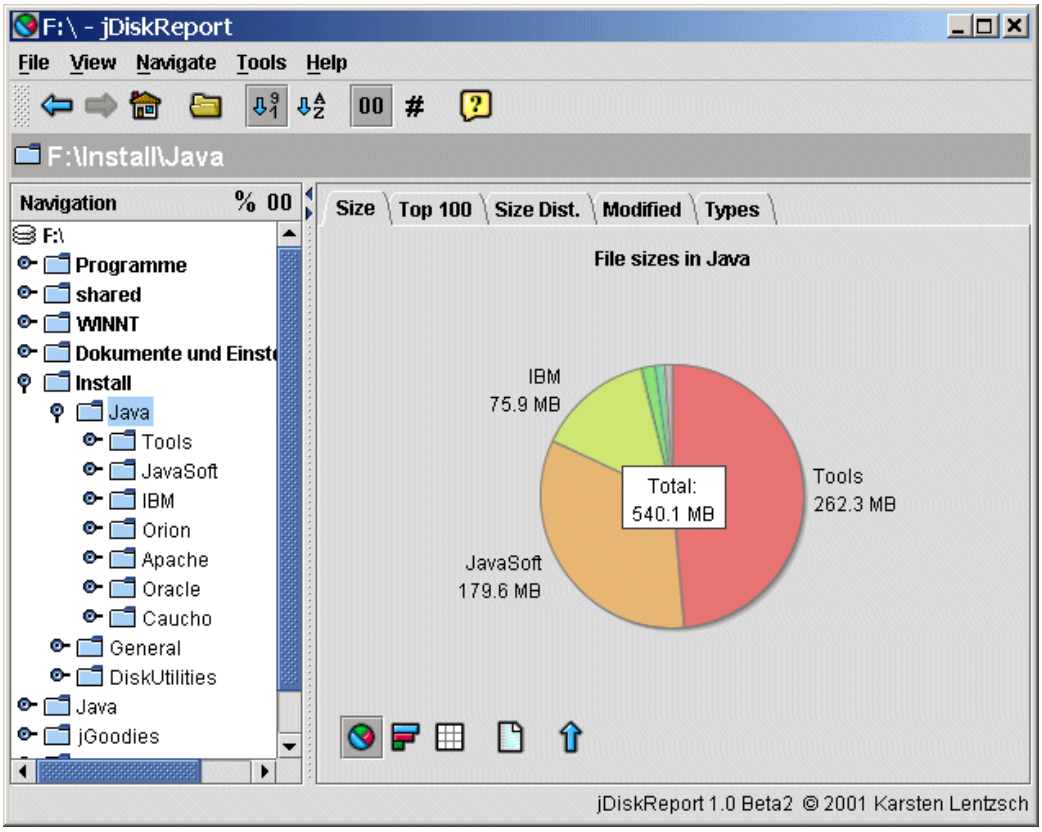
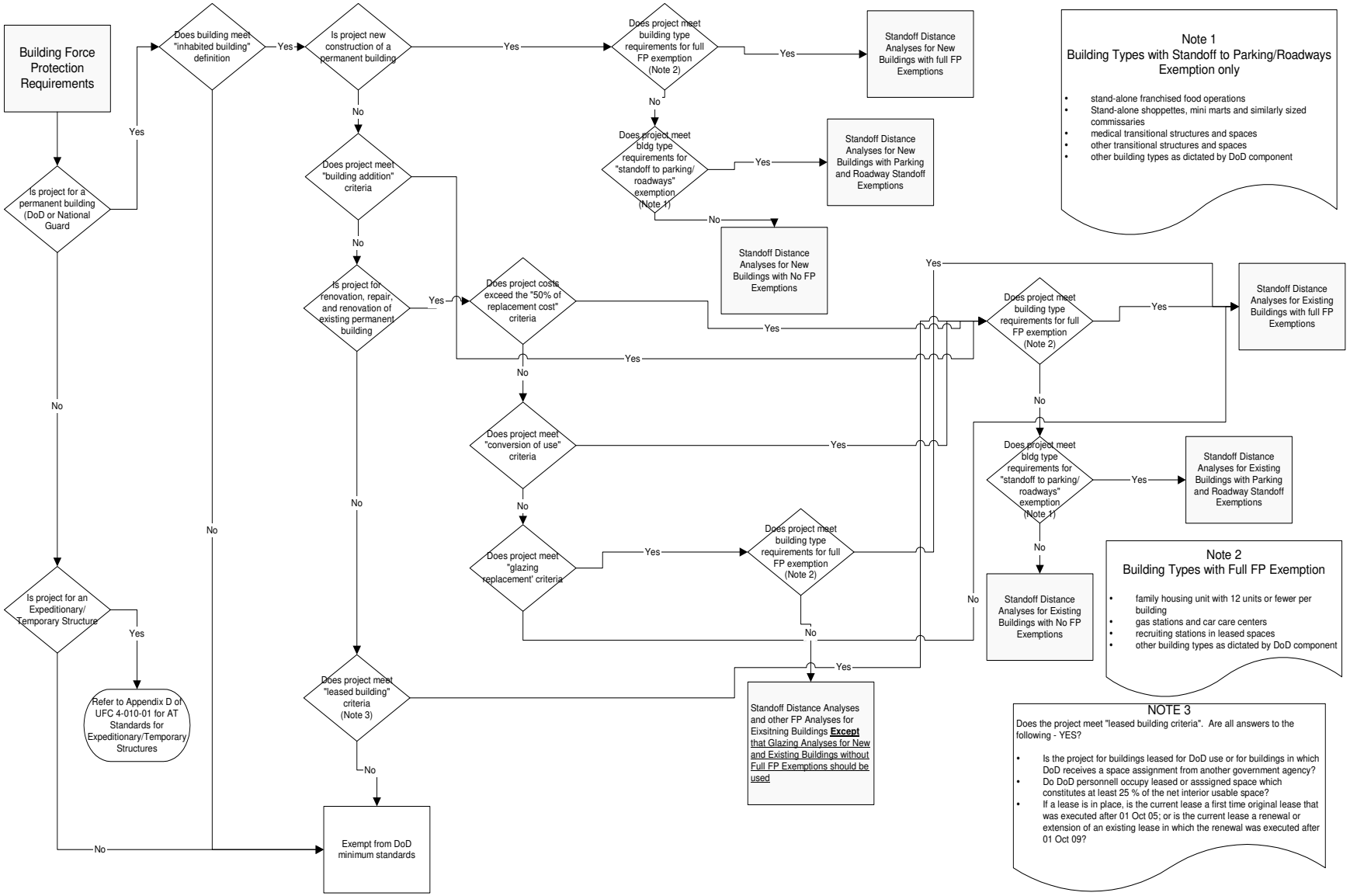
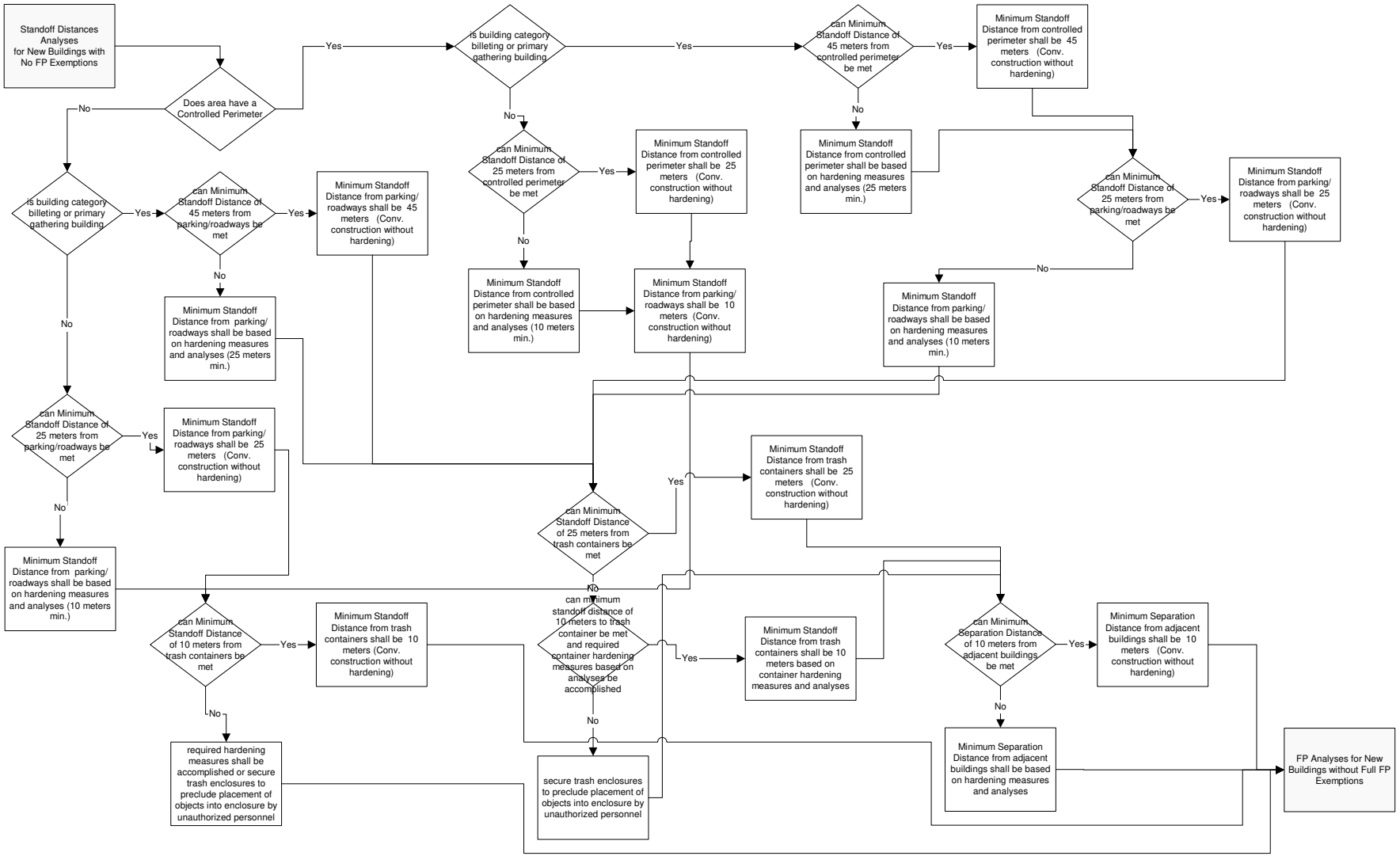


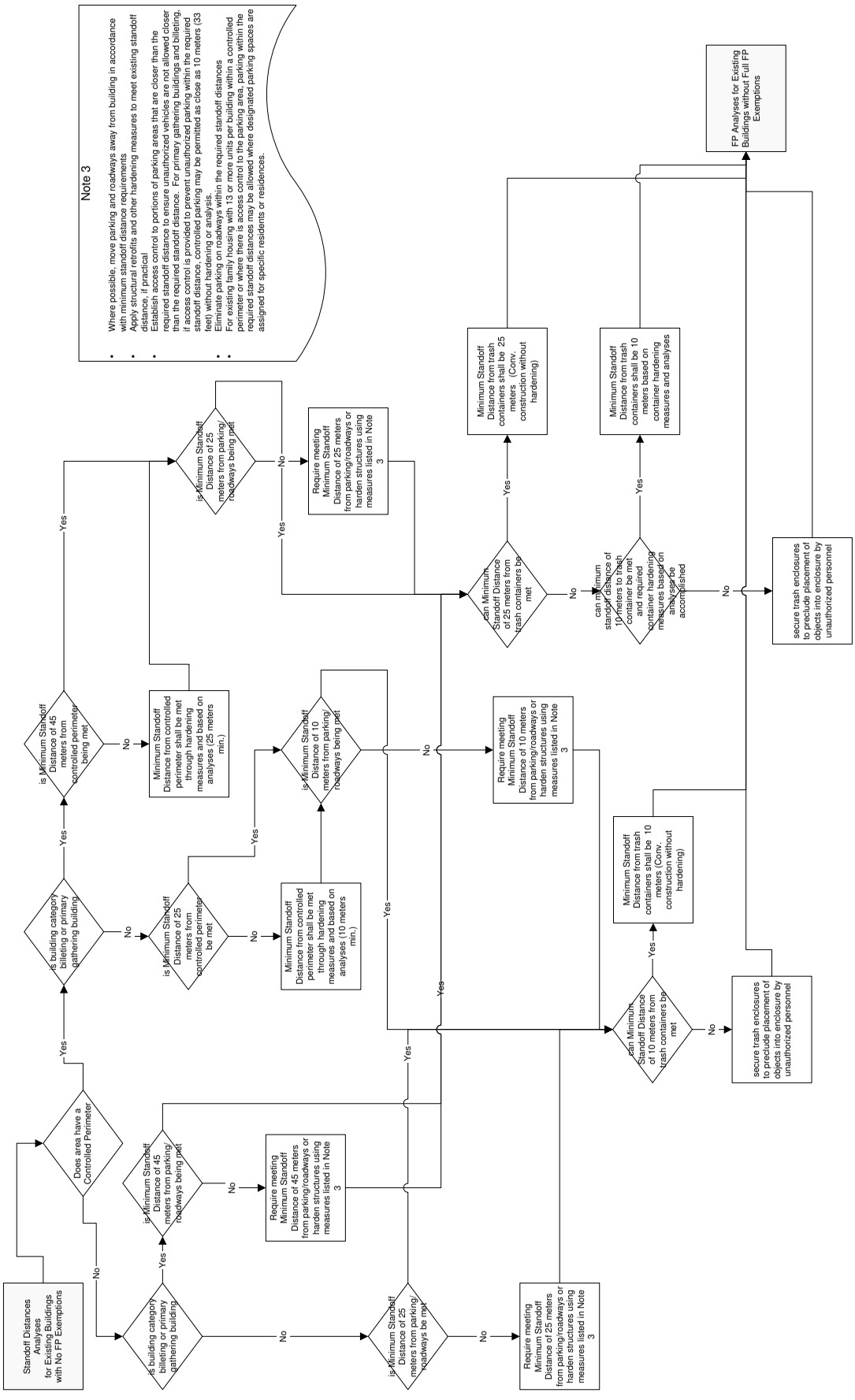
Figure A10. "jDiskReport" analysis screen.

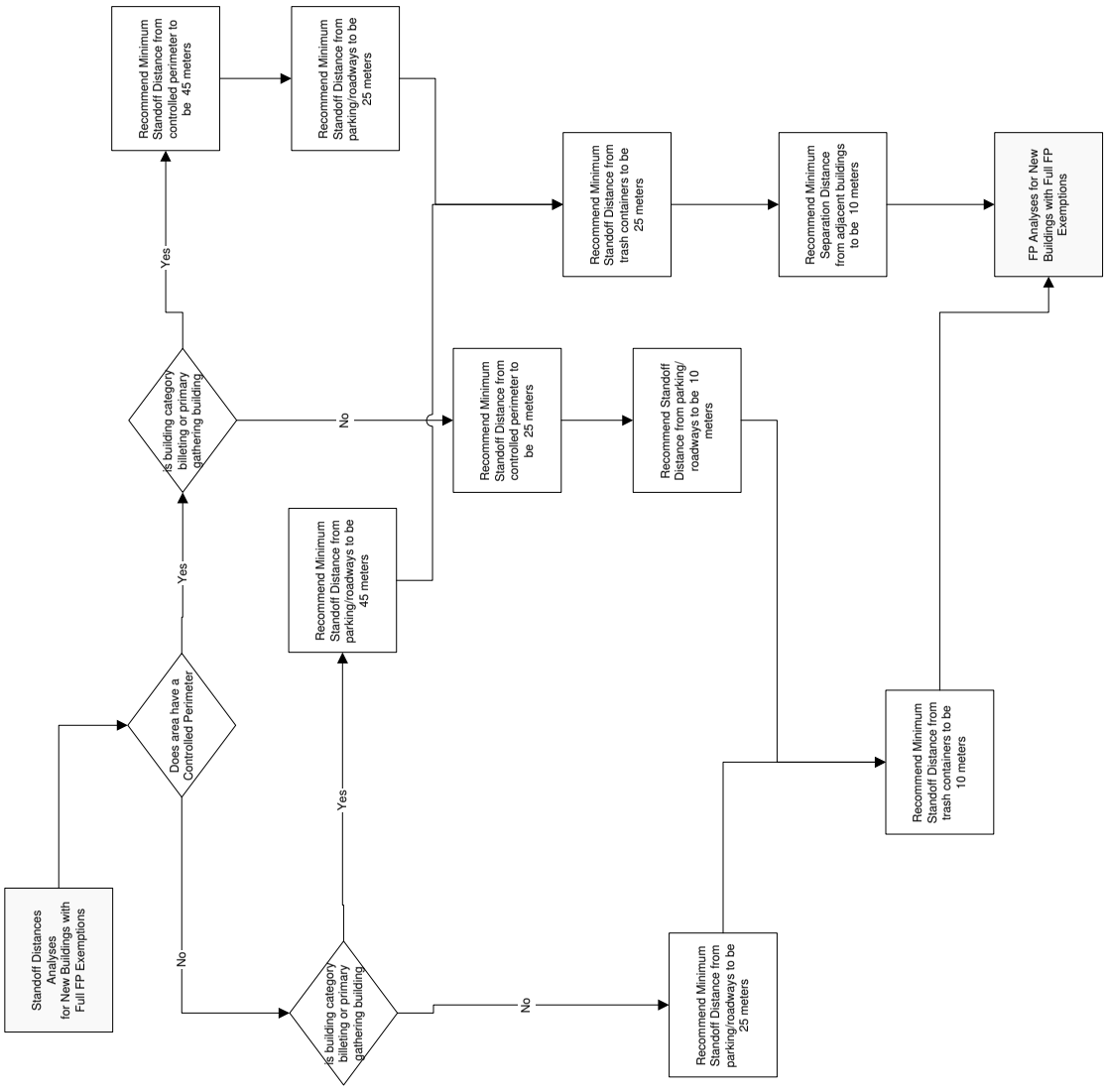
Appendix B: Decision Tree Logic for Minimum Anti-Terrorism Standards for Buildings Wizard

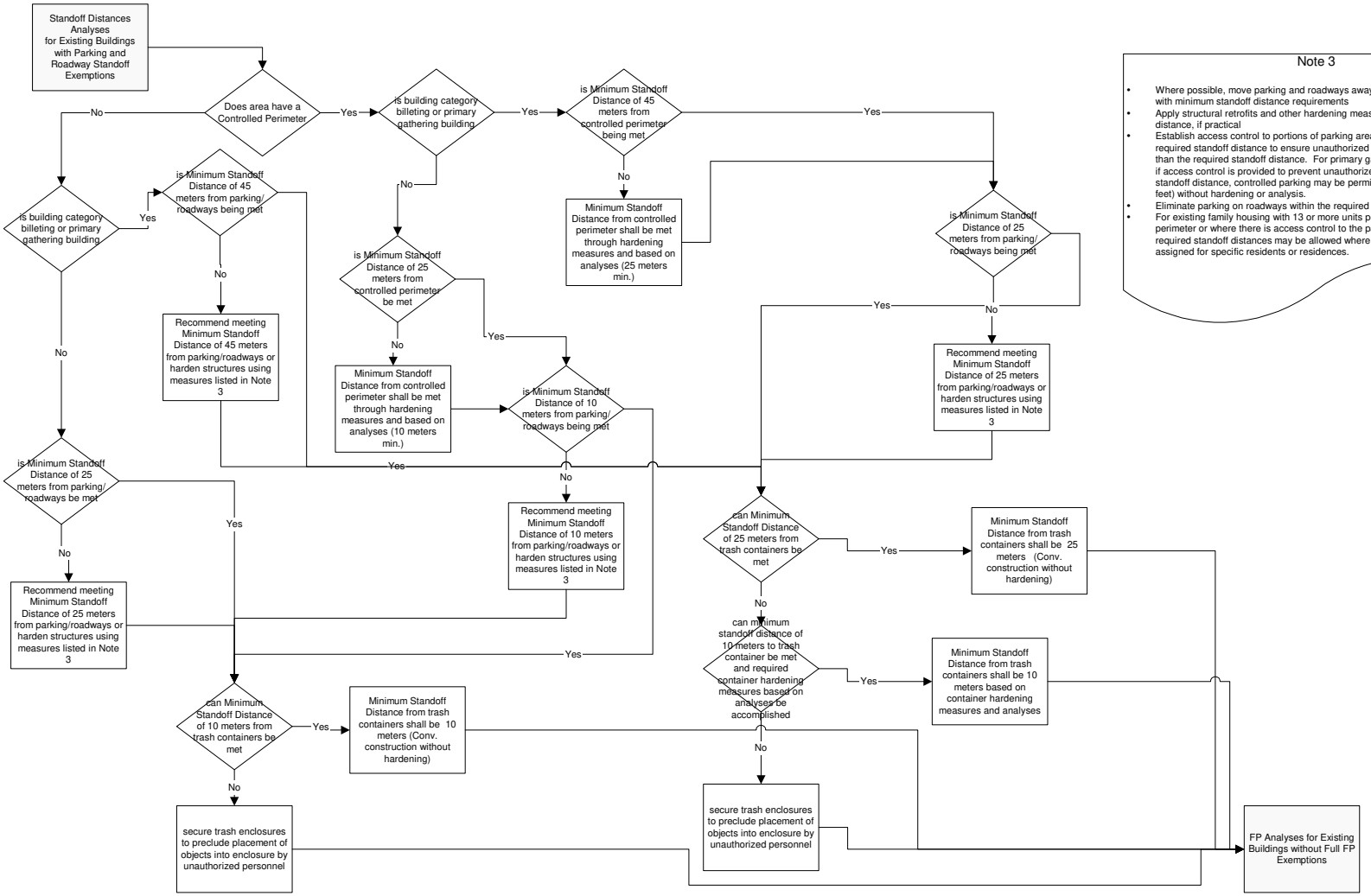
Diagrams included on the following pages outline the decision tree logic that was developed for the Minimum Anti-Terrorism Standards for Buildings Wizard described in Chapter 7. The subject matter expert that performed this analysis was David Bailey of the CF-M branch.

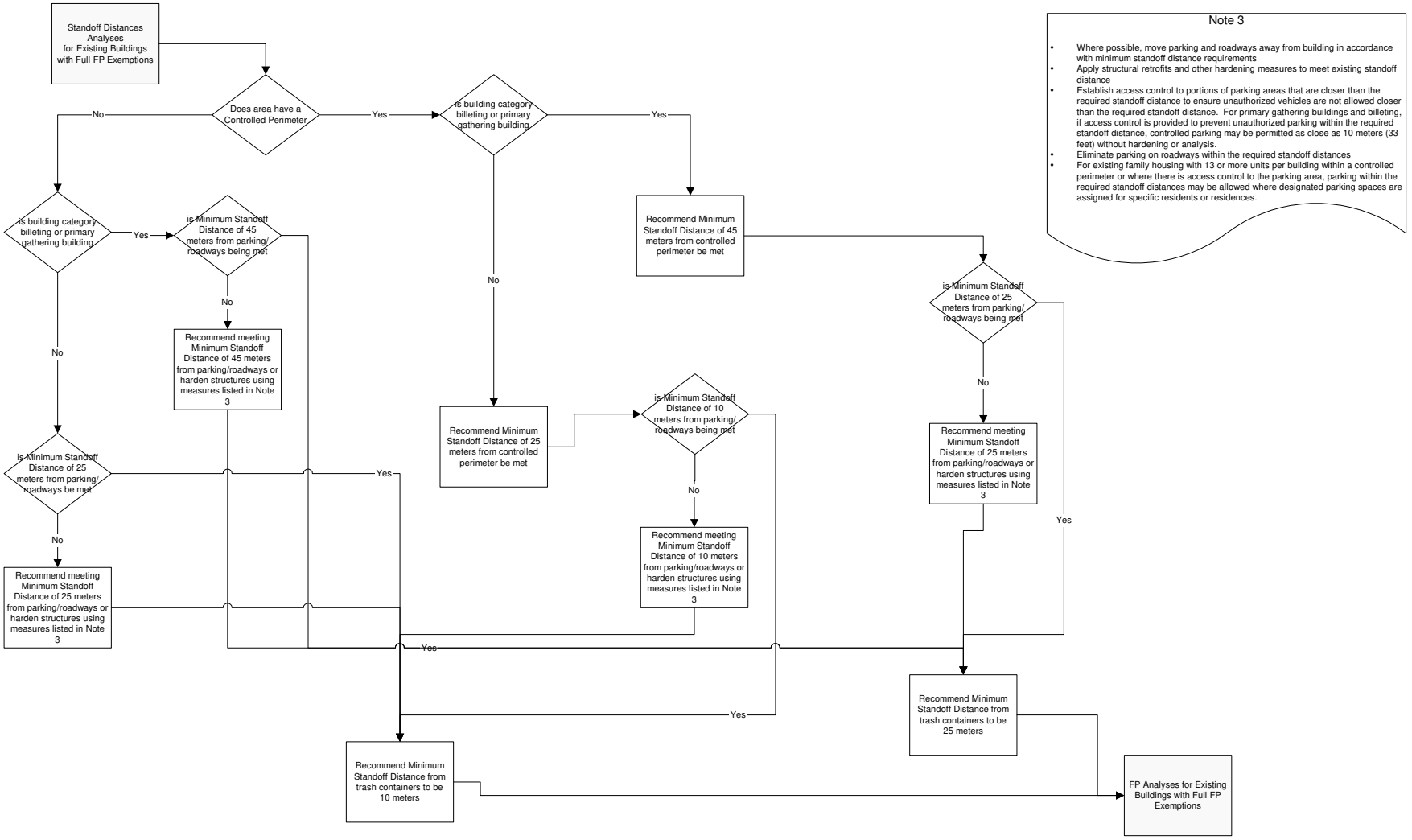

Building Force Protection Requirements
Is project for a permanent building (DoD or National Guard
Does building meet "inhabited building" definition
Is project new construction of a permanent building
Does project meet building type requirements for full FP exemption (Note 2)
Standoff Distance Analyses for New Buildings with full FP Exemptions
Does project meet bldg type requirements for "standoff to parking/ roadways" exemption (Note 1)
Standoff Distance Analyses for New Buildings with Parking and Roadway Standoff Exemptions
Standoff Distance Analyses for New Buildings with No FP Exemptions
Does project meet "building addition" criteria
Is project for renovation, repair, and renovation of existing permanent building
Does project costs exceed the "50% of replacement cost" criteria
Does project meet "conversion of use" criteria
Does project meet "glazing replacement' criteria
Does project meet building type requirements for full FP exemption (Note 2)
Does project meet "leased building" criteria (Note 3)
Standoff Distance Analyses and other FP Analyses for Eixsitning Buildings Except that Glazing Analyses for New and Existing Buildings without Full FP Exemptions should be used
Is project for an Expeditionary/ Temporary Structure
Refer to Appendix D of UFC 4-010-01 for AT Standards for Expeditionary/Temporary Structures
Exempt from DoD minimum standards
Does project meet building type requirements for full FP exemption (Note 2)
Standoff Distance Analyses for Existing Buildings with full FP Exemptions
Does project meet bldg type requirements for "standoff to parking/ roadways" exemption (Note 1)
Standoff Distance Analyses for Existing Buildings with Parking and Roadway Standoff Exemptions
Standoff Distance Analyses for Existing Buildings with No FP Exemptions
Note 1
Building Types with Standoff to Parking/Roadways Exemption only
stand-alone franchised food operations
Stand-alone shoppettes, mini marts and similarly sized commissaries
medical transitional structures and spaces
other transitional structures and spaces
other building types as dictated by DoD component
Note 2
Building Types with Full FP Exemption
family housing unit with 12 units or fewer per building
gas stations and car care centers
recruiting stations in leased spaces
other building types as dictated by DoD component
NOTE 3
Does the project meet "leased building criteria". Are all answers to the following - YES?
Is the project for buildings leased for DoD use or for buildings in which DoD receives a space assignment from another government agency?
Do DoD personnell occupy leased or asssigned space which constitutes at least 25 % of the net interior usable space?
If a lease is in place, is the current lease a first time original lease that was executed after 01 Oct 05; or is the current lease a renewal or extension of an existing lease in which the renewal was executed after 01 Oct 09?

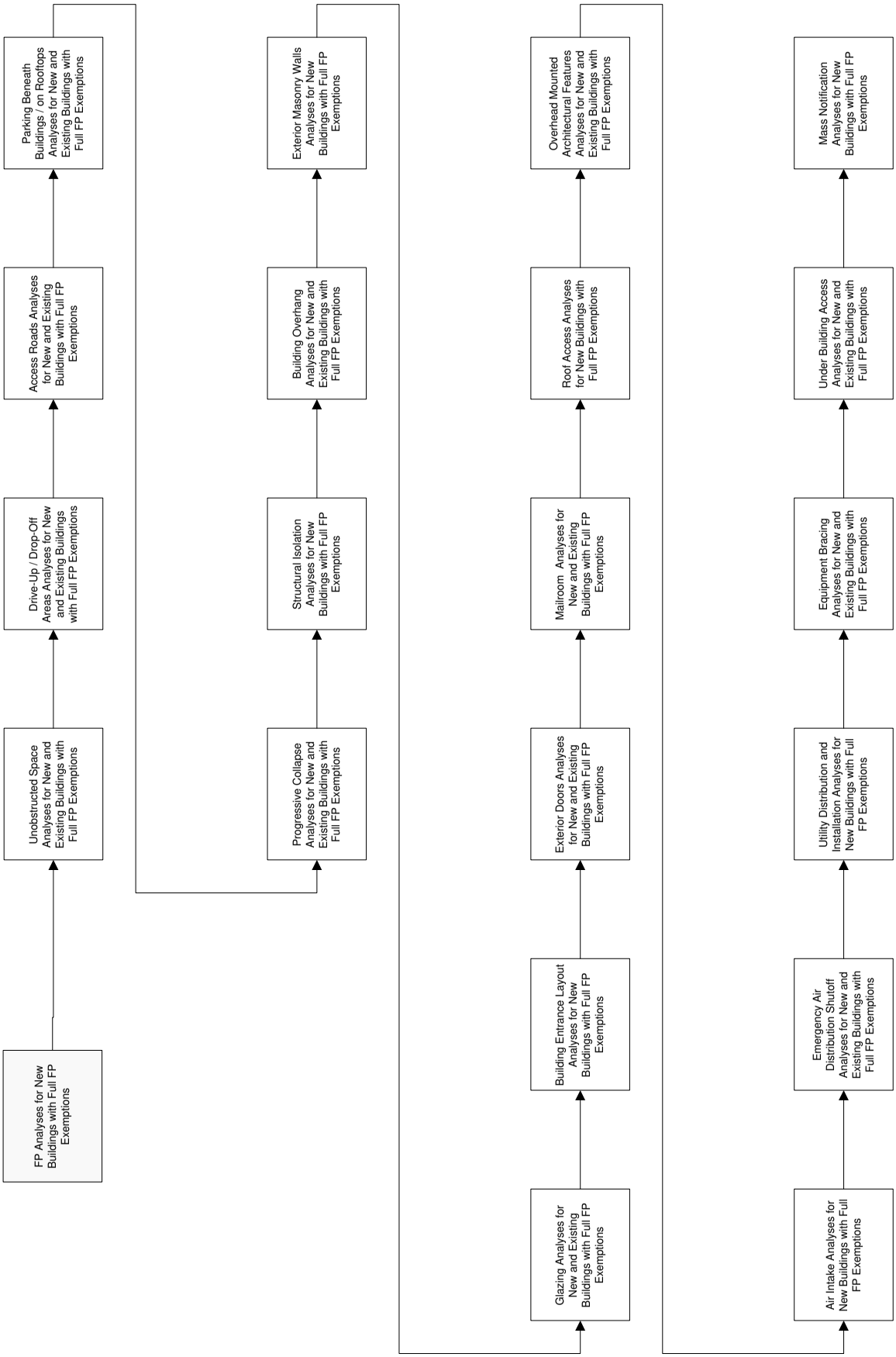

Standoff Distances Analyses for New Buildings with No FP Exemptions
Does area have a Controlled Perimeter
Yes
No
is building category billeting or primary gathering building
Yes
No
can Minimum Standoff Distance of 45 meters from controlled perimeter be met
Yes
No
Minimum Standoff Distance from controlled perimeter shall be 45 meters (Conv. construction without hardening)
Minimum Standoff Distance from controlled perimeter shall be based on hardening measures and analyses (25 meters min.)
can Minimum Standoff Distance of 25 meters from parking/roadways be met
Yes
No
Minimum Standoff Distance from parking/ roadways shall be 25 meters (Conv. construction without hardening)
Minimum Standoff Distance from parking/ roadways shall be based on hardening measures and analyses (10 meters min.)
is building category billeting or primary gathering building
Yes
No
can Minimum Standoff Distance of 45 meters from parking/roadways be met
Yes
No
Minimum Standoff Distance from parking/ roadways shall be 45 meters (Conv. construction without hardening)
Minimum Standoff Distance from parking/ roadways shall be based on hardening measures and analyses (25 meters min.)
can Minimum Standoff Distance of 25 meters from controlled perimeter be met
Yes
No
Minimum Standoff Distance from controlled perimeter shall be 25 meters (Conv. construction without hardening)
Minimum Standoff Distance from controlled perimeter shall be based on hardening measures and analyses (10 meters min.)
Minimum Standoff Distance from parking/ roadways shall be 10 meters (Conv. construction without hardening)
can Minimum Standoff Distance of 25 meters from parking/roadways be met
Yes
No
Minimum Standoff Distance from parking/ roadways shall be 25 meters (Conv. construction without hardening)
Minimum Standoff Distance from parking/ roadways shall be based on hardening measures and analyses (10 meters min.)
can Minimum Standoff Distance of 25 meters from trash containers be met
Yes
No
Minimum Standoff Distance from trash containers shall be 25 meters (Conv. construction without hardening)
Minimum Standoff Distance from parking/ roadways shall be based on hardening measures and analyses (10 meters min.)
can Minimum Standoff Distance of 10 meters from trash containers be met
Yes
No
Minimum Standoff Distance from trash containers shall be 10 meters (Conv. construction without hardening)
required hardening measures shall be accomplished or secure trash enclosures to preclude placement of objects into enclosure by unauthorized personnel
can minimum standoff distance of 10 meters to trash container be met and required container hardening measures based on analyses be accomplished
Yes
No
Minimum Standoff Distance from trash containers shall be 10 meters based on container hardening measures and analyses
secure trash enclosures to preclude placement of objects into enclosure by unauthorized personnel
can Minimum Separation Distance of 10 meters from adjacent buildings be met
Yes
No
Minimum Separation Distance from adjacent buildings shall be 10 meters (Conv. construction without hardening)
Minimum Separation Distance from adjacent buildings shall be based on hardening measures and analyses
FP Analyses for New Buildings without Full FP Exemptions

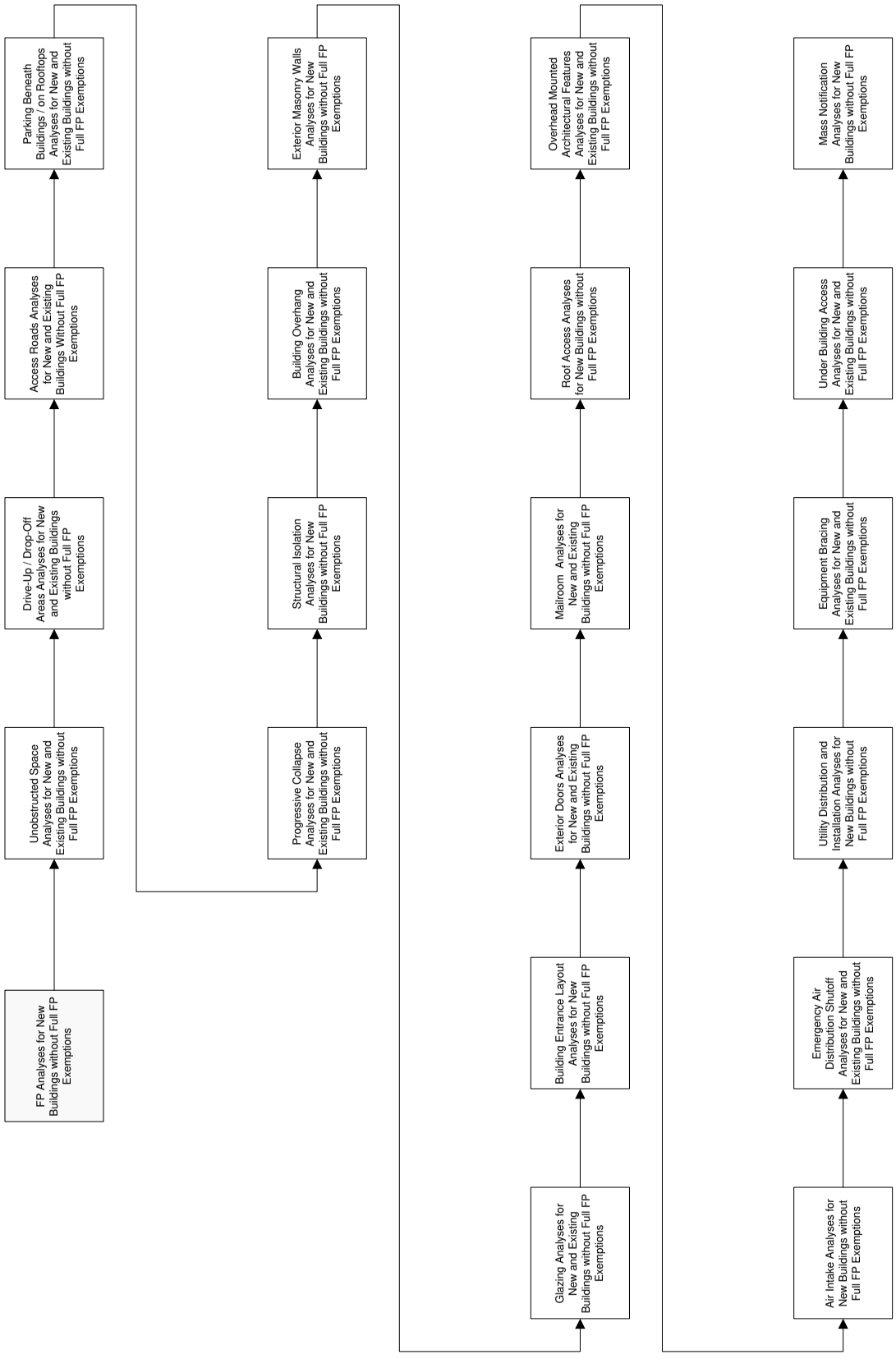

Standoff Distances Analyses for Existing Buildings with No FP Exemptions
Does area have a Controlled Perimeter
is building category billeting or primary gathering building
is Minimum Standoff Distance of 45 meters from controlled perimeter being met
Minimum Standoff Distance from controlled perimeter shall be met through hardening measures and based on analyses (25 meters min.)
is Minimum Standoff Distance of 25 meters from controlled perimeter be met
Minimum Standoff Distance from controlled perimeter shall be met through hardening measures and based on analyses (10 meters min.)
is Minimum Standoff Distance of 10 meters from parking/roadways being met
Require meeting Minimum Standoff Distance of 10 meters from parking/roadways or harden structures using measures listed in Note 3
Can Minimum Standoff Distance of 10 meters from trash containers be met
Minimum Standoff Distance from trash containers shall be 10 meters (Conv. construction without hardening)
secure trash enclosures to preclude placement of objects into enclosure by unauthorized personnel
is building category billeting or primary gathering building
is Minimum Standoff Distance of 45 meters from parking/roadways being met
Require meeting Minimum Standoff Distance of 45 meters from parking/roadways or harden structures using measures listed in Note 3
is Minimum Standoff Distance of 25 meters from parking/roadways being met
Require meeting Minimum Standoff Distance of 25 meters from parking/roadways or harden structures using measures listed in Note 3
is Minimum Standoff Distance of 25 meters from parking/roadways being met
Require meeting Minimum Standoff Distance of 25 meters from parking/roadways or harden structures using measures listed in Note 3
Can Minimum Standoff Distance of 25 meters from trash containers be met
Minimum Standoff Distance from trash containers shall be 25 meters (Conv. construction without hardening)
can minimum standoff distance of 10 meters to trash container be met and required container hardening measures based on analyses be accomplished
Minimum Standoff Distance from trash containers shall be 10 meters based on container hardening measures and analyses
secure trash enclosures to preclude placement of objects into enclosure by unauthorized personnel
FP Analyses for Existing Buildings without Full FP Exemptions
Yes
No
Note 3
Where possible, move parking and roadways away from building in accordance with minimum standoff distance requirements.
Apply structural retrofits and other hardening measures to meet existing standoff distance, if practical.
Establish access control to portions of parking areas that are closer than the required standoff distance to ensure unauthorized vehicles are not allowed closer than the required standoff distance. For primary gathering buildings and billeting, if access control is provided to prevent unauthorized parking within the required standoff distance, controlled parking may be permitted as close as 10 meters (33 feet) without hardening or analysis.
Eliminate parking on roadways within the required standoff distances
For existing family housing with 13 or more units per building within a controlled perimeter or where there is access control to the parking area, parking within the required standoff distances may be allowed where designated parking spaces are assigned for specific residents or residences.

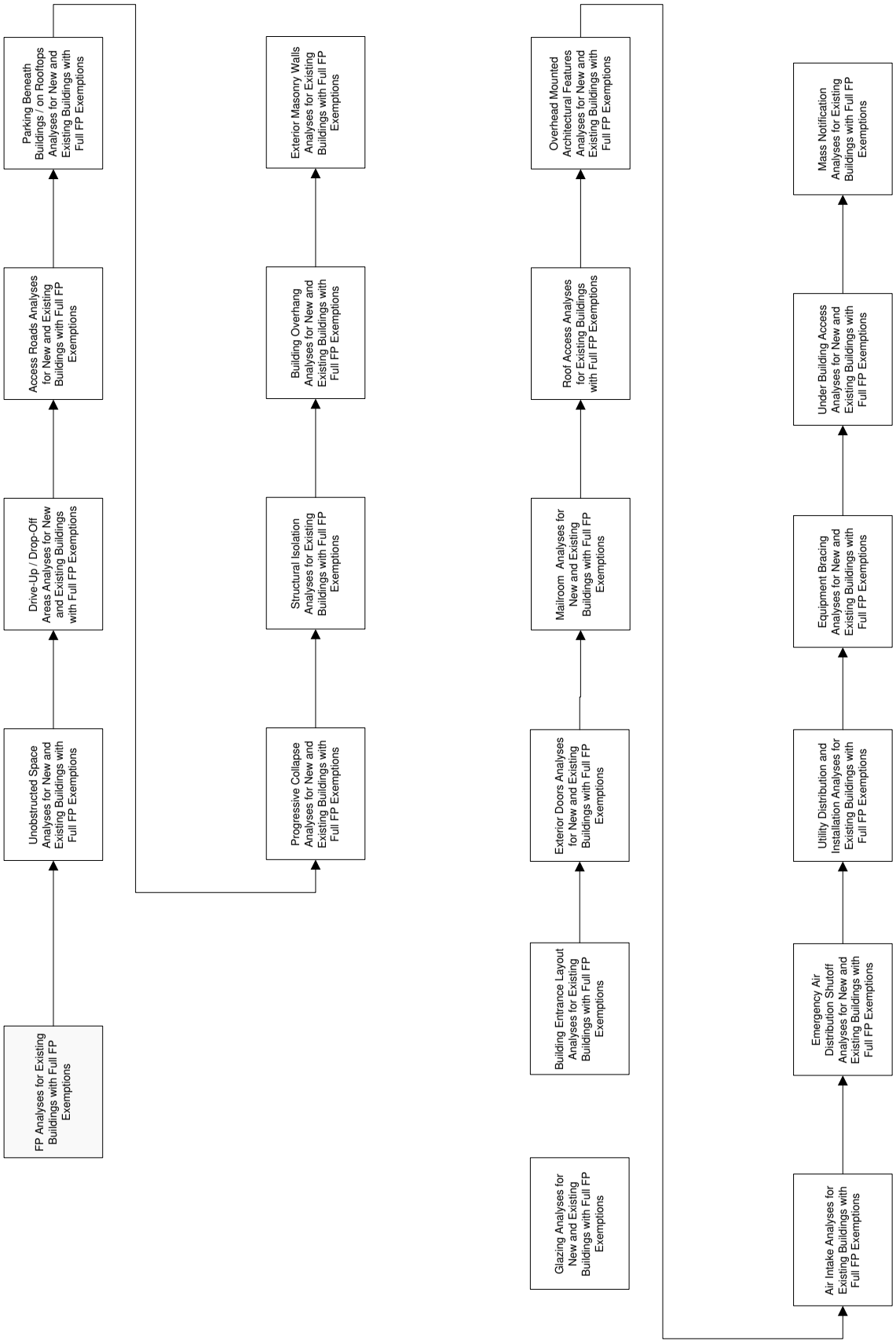

Standoff Distances Analyses for New Buildings with Full FP Exemptions
Does area have a Controlled Perimeter
Yes
No
Is building category billeting or primary gathering building
Yes
No
Recommend Minimum Standoff Distance from controlled perimeter to be 45 meters
Recommend Minimum Standoff Distance from parking/roadways to be 25 meters
Recommend Minimum Standoff Distance from parking/roadways to be 45 meters
Recommend Minimum Standoff Distance from controlled perimeter to be 25 meters
Recommend Standoff Distance from parking/ roadways to be 10 meters
Recommend Minimum Standoff Distance from parking/roadways to be 25 meters
Recommend Minimum Standoff Distance from trash containers to be 25 meters
Recommend Minimum Standoff Distance from trash containers to be 10 meters
Recommend Minimum Separation Distance from adjacent buildings to be 10 meters
FP Analyses for New Buildings with Full FP Exemptions

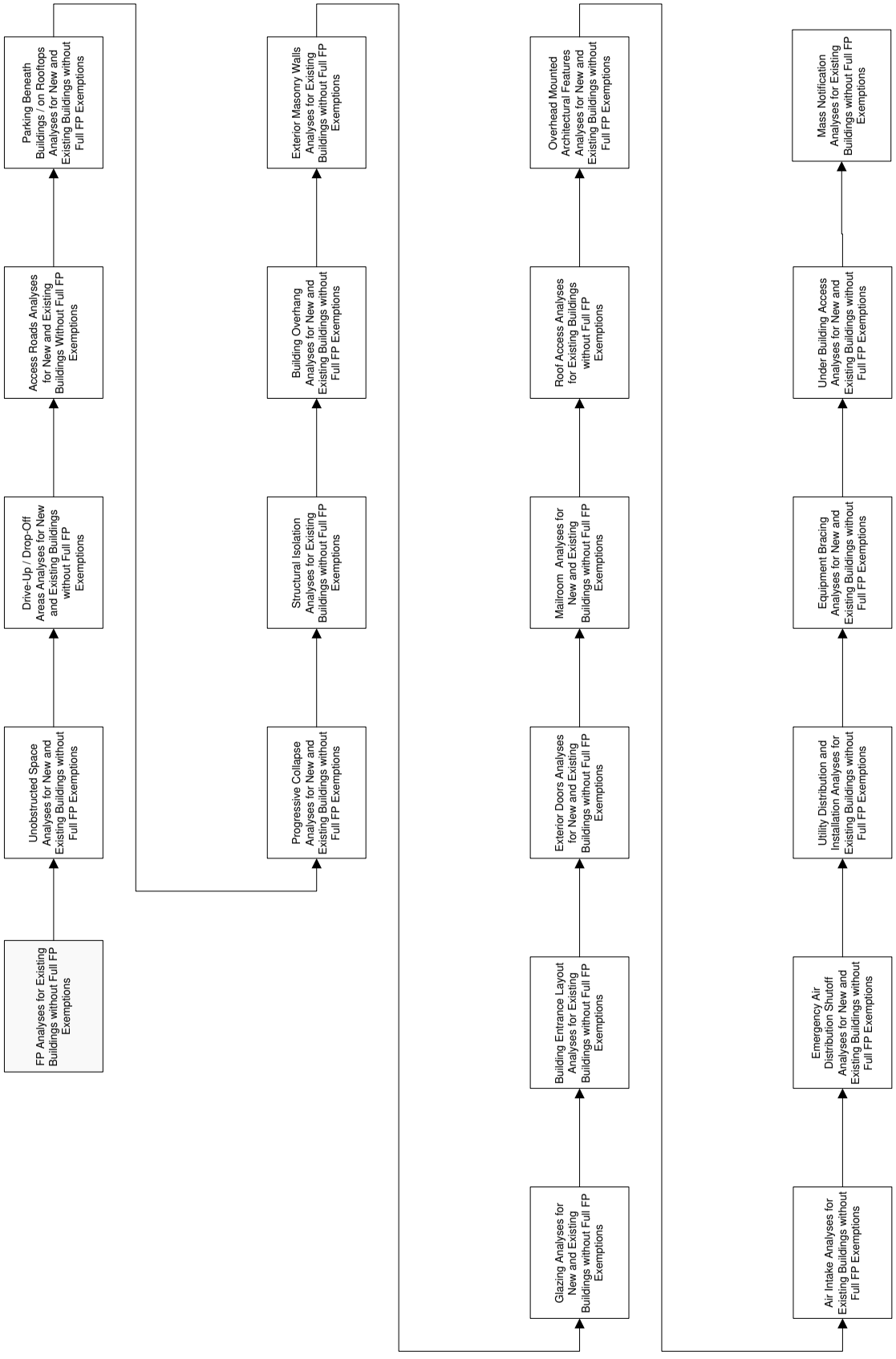

Standoff Distances Analyses for Existing Buildings with Parking and Roadway Standoff Exemptions
Does area have a Controlled Perimeter
No
Yes
is building category billeting or primary gathering building
Yes
is Minimum Standoff Distance of 45 meters from controlled perimeter being met
Yes
No
Minimum Standoff Distance from controlled perimeter shall be met through hardening measures and based on analyses (25 meters min.)
is Minimum Standoff Distance of 25 meters from parking/ roadways being met
Yes
No
Recommend meeting Minimum Standoff Distance of 25 meters from parking/roadways or harden structures using measures listed in Note 3
is building category billeting or primary gathering building
Yes
No
is Minimum Standoff Distance of 45 meters from parking/ roadways being met
No
Recommend meeting Minimum Standoff Distance of 45 meters from parking/roadways or harden structures using measures listed in Note 3
is Minimum Standoff Distance of 25 meters from controlled perimeter be met
No
Yes
Minimum Standoff Distance from controlled perimeter shall be met through hardening measures and based on analyses (10 meters min.)
is Minimum Standoff Distance of 10 meters from parking/ roadways being met
Yes
No
Recommend meeting Minimum Standoff Distance of 10 meters from parking/roadways or harden structures using measures listed in Note 3
is Minimum Standoff Distance of 25 meters from parking/ roadways be met
No
Yes
Recommend meeting Minimum Standoff Distance of 25 meters from parking/roadways or harden structures using measures listed in Note 3
can Minimum Standoff Distance of 25 meters from trash containers be met
Yes
Minimum Standoff Distance from trash containers shall be 25 meters (Conv. construction without hardening)
No
can minimum standoff distance of 10 meters to trash container be met and required container hardening measures based on analyses be accomplished
Yes
Minimum Standoff Distance from trash containers shall be 10 meters based on container hardening measures and analyses
No
secure trash enclosures to preclude placement of objects into enclosure by unauthorized personnel
can Minimum Standoff Distance of 10 meters from trash containers be met
Yes
Minimum Standoff Distance from trash containers shall be 10 meters (Conv. construction without hardening)
No
secure trash enclosures to preclude placement of objects into enclosure by unauthorized personnel
FP Analyses for Existing Buildings without Full FP Exemptions
Note 3
Where possible, move parking and roadways away from building in accordance with minimum standoff distance requirements
Apply structural retrofits and other hardening measures to meet existing standoff distance, if practical
Establish access control to portions of parking areas that are closer than the required standoff distance to ensure unauthorized vehicles are not allowed closer than the required standoff distance. For primary gathering buildings and billeting, if access control is provided to prevent unauthorized parking within the required standoff distance, controlled parking may be permitted as close as 10 meters (33 feet) without hardening or analysis.
Eliminate parking on roadways within the required standoff distances
For existing family housing with 13 or more units per building within a controlled perimeter or where there is access control to the parking area, parking within the required standoff distances may be allowed where designated parking spaces are assigned for specific residents or residences.


Standoff Distances Analyses for Existing Buildings with Full FP Exemptions
Does area have a Controlled Perimeter
No
Yes
is building category billeting or primary gathering building
Yes
Recommend Minimum Standoff Distance of 45 meters from controlled perimeter be met
is Minimum Standoff Distance of 25 meters from parking/ roadways being met
No
Recommend meeting Minimum Standoff Distance of 25 meters from parking/roadways or harden structures using measures listed in Note 3
Yes
is building category billeting or primary gathering building
Yes
is Minimum Standoff Distance of 45 meters from parking/ roadways being met
No
Recommend meeting Minimum Standoff Distance of 45 meters from parking/roadways or harden structures using measures listed in Note 3
No
is Minimum Standoff Distance of 25 meters from parking/ roadways be met
No
Recommend meeting Minimum Standoff Distance of 25 meters from parking/roadways or harden structures using measures listed in Note 3
Yes
No
Recommend Minimum Standoff Distance of 25 meters from controlled perimeter be met
is Minimum Standoff Distance of 10 meters from parking/ roadways being met
No
Recommend meeting Minimum Standoff Distance of 10 meters from parking/roadways or harden structures using measures listed in Note 3
Yes
Yes
Recommend Minimum Standoff Distance from trash containers to be 10 meters
Yes
Recommend Minimum Standoff Distance from trash containers to be 25 meters
FP Analyses for Existing Buildings with Full FP Exemptions
Note 3
Where possible, move parking and roadways away from building in accordance with minimum standoff distance requirements
Apply structural retrofits and other hardening measures to meet existing standoff distance
Establish access control to portions of parking areas that are closer than the required standoff distance to ensure unauthorized vehicles are not allowed closer than the required standoff distance. For primary gathering buildings and billeting, if access control is provided to prevent unauthorized parking within the required standoff distance, controlled parking may be permitted as close as 10 meters (33 feet) without hardening or analysis.
Eliminate parking on roadways within the required standoff distances
For existing family housing with 13 or more units per building within a controlled perimeter or where there is access control to the parking area, parking within the required standoff distances may be allowed where designated parking spaces are assigned for specific residents or residences.


FP Analyses for New Buildings with Full FP Exemptions
Unobstructed Space Analyses for New and Existing Buildings with Full FP Exemptions
Drive-Up / Drop-Off Areas Analyses for New and Existing Buildings with Full FP Exemptions
Access Roads Analyses for New and Existing Buildings with Full FP Exemptions
Parking Beneath Buildings / on Rooftops Analyses for New and Existing Buildings with Full FP Exemptions
Progressive Collapse Analyses for New and Existing Buildings with Full FP Exemptions
Structural Isolation Analyses for New Buildings with Full FP Exemptions
Building Overhang Analyses for New and Existing Buildings with Full FP Exemptions
Exterior Masonry Walls Analyses for New Buildings with Full FP Exemptions
Glazing Analyses for New and Existing Buildings with Full FP Exemptions
Building Entrance Layout Analyses for New Buildings with Full FP Exemptions
Exterior Doors Analyses for New and Existing Buildings with Full FP Exemptions
Mailroom Analyses for New and Existing Buildings with Full FP Exemptions
Roof Access Analyses for New Buildings with Full FP Exemptions
Overhead Mounted Architectural Features Analyses for New and Existing Buildings with Full FP Exemptions
Air Intake Analyses for New Buildings with Full FP Exemptions
Emergency Air Distribution Shutoff Analyses for New and Existing Buildings with Full FP Exemptions
Utility Distribution and Installation Analyses for New Buildings with Full FP Exemptions
Equipment Bracing Analyses for New and Existing Buildings with Full FP Exemptions
Under Building Access Analyses for New and Existing Buildings with Full FP Exemptions
Mass Notification Analyses for New Buildings with Full FP Exemptions


FP Analyses for New Buildings without Full FP Exemptions
Unobstructed Space Analyses for New and Existing Buildings without Full FP Exemptions
Drive-Up / Drop-Off Areas Analyses for New and Existing Buildings without Full FP Exemptions
Access Roads Analyses for New and Existing Buildings Without Full FP Exemptions
Parking Beneath Buildings / on Rooftops Analyses for New and Existing Buildings without Full FP Exemptions
Progressive Collapse Analyses for New and Existing Buildings without Full FP Exemptions
Structural Isolation Analyses for New Buildings without Full FP Exemptions
Building Overhang Analyses for New and Existing Buildings without Full FP Exemptions
Exterior Masonry Walls Analyses for New Buildings without Full FP Exemptions
Glazing Analyses for New and Existing Buildings without Full FP Exemptions
Building Entrance Layout Analyses for New Buildings without Full FP Exemptions
Exterior Doors Analyses for New and Existing Buildings without Full FP Exemptions
Mailroom Analyses for New and Existing Buildings without Full FP Exemptions
Roof Access Analyses for New Buildings without Full FP Exemptions
Overhead Mounted Architectural Features Analyses for New and Existing Buildings without Full FP Exemptions
Air Intake Analyses for New Buildings without Full FP Exemptions
Emergency Air Distribution Shutoff Analyses for New and Existing Buildings without Full FP Exemptions
Utility Distribution and Installation Analyses for New Buildings without Full FP Exemptions
Equipment Bracing Analyses for New and Existing Buildings without Full FP Exemptions
Under Building Access Analyses for New and Existing Buildings without Full FP Exemptions
Mass Notification Analyses for New Buildings without Full FP Exemptions


FP Analyses for Existing Buildings with Full FP Exemptions
Unobstructed Space Analyses for New and Existing Buildings with Full FP Exemptions
Drive-Up / Drop-Off Areas Analyses for New and Existing Buildings with Full FP Exemptions
Access Roads Analyses for New and Existing Buildings with Full FP Exemptions
Parking Beneath Buildings / on Rooftops Analyses for New and Existing Buildings with Full FP Exemptions
Progressive Collapse Analyses for New and Existing Buildings with Full FP Exemptions
Structural Isolation Analyses for Existing Buildings with Full FP Exemptions
Building Overhang Analyses for New and Existing Buildings with Full FP Exemptions
Exterior Masonry Walls Analyses for Existing Buildings with Full FP Exemptions
Glazing Analyses for New and Existing Buildings with Full FP Exemptions
Building Entrance Layout Analyses for Existing Buildings with Full FP Exemptions
Exterior Doors Analyses for New and Existing Buildings with Full FP Exemptions
Mailroom Analyses for New and Existing Buildings with Full FP Exemptions
Roof Access Analyses for Existing Buildings with Full FP Exemptions
Overhead Mounted Architectural Features Analyses for New and Existing Buildings with Full FP Exemptions
Air Intake Analyses for Existing Buildings with Full FP Exemptions
Emergency Air Distribution Shutoff Analyses for New and Existing Buildings with Full FP Exemptions
Utility Distribution and Installation Analyses for Existing Buildings with Full FP Exemptions
Equipment Bracing Analyses for New and Existing Buildings with Full FP Exemptions
Under Building Access Analyses for New and Existing Buildings with Full FP Exemptions
Mass Notification Analyses for Existing Buildings with Full FP Exemptions


FP Analyses for Existing Buildings without Full FP Exemptions
Unobstructed Space Analyses for New and Existing Buildings without Full FP Exemptions
Drive-Up / Drop-Off Areas Analyses for New and Existing Buildings without Full FP Exemptions
Access Roads Analyses for New and Existing Buildings Without Full FP Exemptions
Parking Beneath Buildings / on Rooftops Analyses for New and Existing Buildings without Full FP Exemptions
Progressive Collapse Analyses for New and Existing Buildings without Full FP Exemptions
Structural Isolation Analyses for Existing Buildings without Full FP Exemptions
Building Overhang Analyses for New and Existing Buildings without Full FP Exemptions
Exterior Masonry Walls Analyses for Existing Buildings without Full FP Exemptions
Glazing Analyses for New and Existing Buildings without Full FP Exemptions
Building Entrance Layout Analyses for Existing Buildings without Full FP Exemptions
Exterior Doors Analyses for New and Existing Buildings without Full FP Exemptions
Mailroom Analyses for New and Existing Buildings without Full FP Exemptions
Roof Access Analyses for Existing Buildings without Full FP Exemptions
Overhead Mounted Architectural Features Analyses for New and Existing Buildings without Full FP Exemptions
Air Intake Analyses for Existing Buildings without Full FP Exemptions
Emergency Air Distribution Shutoff Analyses for New and Existing Buildings without Full FP Exemptions
Utility Distribution and Installation Analyses for Existing Buildings without Full FP Exemptions
Equipment Bracing Analyses for New and Existing Buildings without Full FP Exemptions
Under Building Access Analyses for New and Existing Buildings without Full FP Exemptions
Mass Notification Analyses for Existing Buildings without Full FP Exemptions

Unobstructed Space Analyses for New and Existing Buildings without Full FP Exemptions

Require

- Building Perimeter. Ensure that obstructions within 10 meters (33 feet) of inhabited buildings or portions thereof do not allow for concealment from observation of explosive devices 150 mm (6 inches) or greater in height. This does not preclude the placement of site furnishings or plantings around buildings. It only requires conditions such that any explosive devices placed in that space would be observable by building occupants. For existing buildings where the standoff distances for parking and roadways have been established at less than 10 meters in accordance withpara. B-1.1.2.2, the unobstructed space may be reduced to be equivalent to that distance.
- Electrical and Mechanical Equipment. The preferred location of electrical and mechanical equipment such as transformers, air-cooled condensers, and packaged chillers is outside the unobstructed space or on the roof. However this equipment can be placed within the unobstructed space as long the equipment provides no opportunity for concealment of explosive devices.
- Equipment Enclosures. If walls or other screening devices with more than two sides are placed around electrical or mechanical equipment within the unobstructed space, enclose the equipment on all four sides and the top. Openings in screening materials and gaps between the ground and screens or walls making up an enclosure will not be greater than 150 mm (6 inches). Secure any surfaces of the enclosures that can be opened so that unauthorized personnel cannot gain access through them.

Unobstructed Space Analyses for New and Existing Buildings with Full FP Exemptions

Recommend

- Building Perimeter. Ensure that obstructions within 10 meters (33 feet) of inhabited buildings or portions thereof do not allow for concealment from observation of explosive devices 150 mm (6 inches) or greater in height. This does not preclude the placement of site furnishings or plantings around buildings. It only requires conditions such that any explosive devices placed in that space would be observable by building occupants. For existing buildings where the standoff distances for parking and roadways have been established at less than 10 meters in accordance withpara. B-1.1.2.2, the unobstructed space may be reduced to be equivalent to that distance.
- Electrical and Mechanical Equipment. The preferred location of electrical and mechanical equipment such as transformers, air-cooled condensers, and packaged chillers is outside the unobstructed space or on the roof. However this equipment can be placed within the unobstructed space as long the equipment provides no opportunity for concealment of explosive devices.
- Equipment Enclosures. If walls or other screening devices with more than two sides are placed around electrical or mechanical equipment within the unobstructed space, enclose the equipment on all four sides and the top. Openings in screening materials and gaps between the ground and screens or walls making up an enclosure will not be greater than 150 mm (6 inches). Secure any surfaces of the enclosures that can be opened so that unauthorized personnel cannot gain access through them.

Drive-Up / Drop-Off Areas Analyses for New and Existing Buildings without Full FP Exemptions

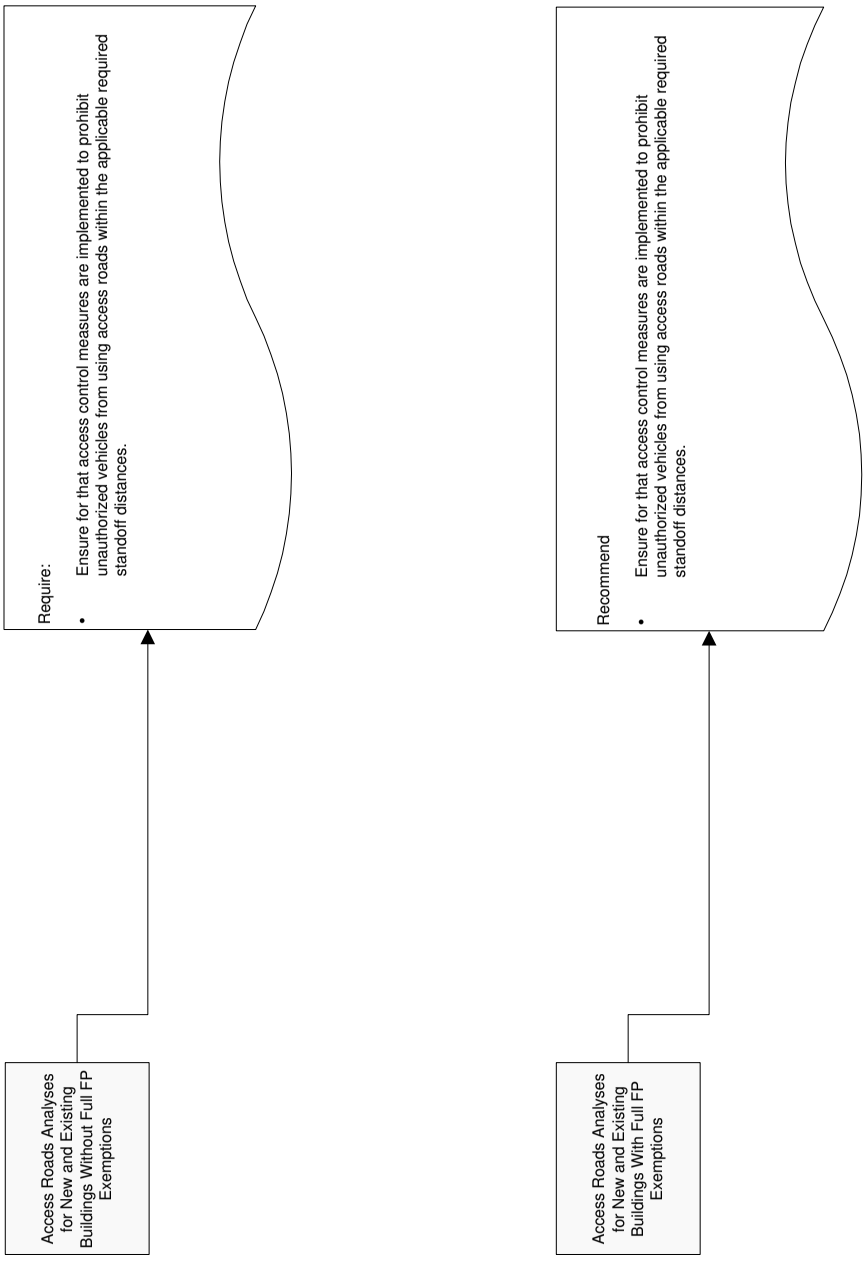
Require

- Marking. Where operational or safety considerations require drive-up or drop-off areas or drive through lanes near buildings, ensure those areas or lanes are clearly defined and marked and that their intended use is clear to prevent parking of vehicles in those areas.
- Unattended Vehicles. Do not allow unattended vehicles in drive-up or drop-off areas or drive through lanes.
- Location. Do not allow drive-through lanes or drive-up/drop-off to be located under any inhabited portion of a building.

Drive-Up / Drop-Off Areas Analyses for New and Existing Buildings with Full FP Exemptions

Recommend

- Marking. Where operational or safety considerations require drive-up or drop-off areas or drive through lanes near buildings, ensure those areas or lanes are clearly defined and marked and that their intended use is clear to prevent parking of vehicles in those areas.
- Unattended Vehicles. Do not allow unattended vehicles in drive-up or drop-off areas or drive through lanes.
- Location. Do not allow drive-through lanes or drive-up/drop-off to be located under any inhabited portion of a building.


Access Roads Analyses for New and Existing Buildings Without Full FP Exemptions
Require:
• Ensure for that access control measures are implemented to prohibit unauthorized vehicles from using access roads within the applicable required standoff distances.
Access Roads Analyses for New and Existing Buildings With Full FP Exemptions
Recommend
• Ensure for that access control measures are implemented to prohibit unauthorized vehicles from using access roads within the applicable required standoff distances.

Parking Beneath Buildings / on Rooftops Analyses for New and Existing Buildings without Full FP Exemptions

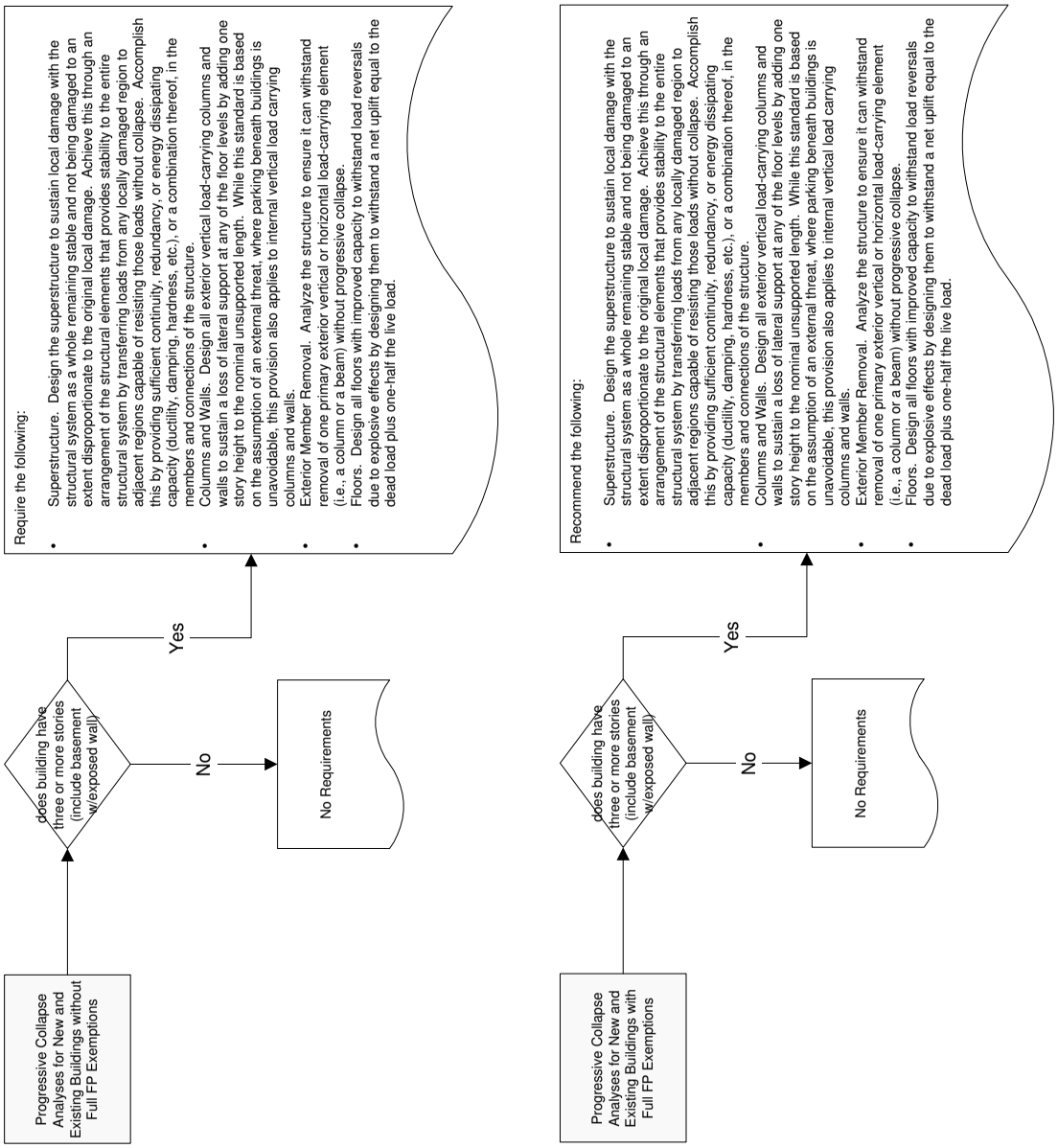
Require that parking beneath inhabited buildings or on rooftops of inhabited buildings be eliminated. Where very limited real estate makes such parking unavoidable, the following measures must be incorporated into the design for new buildings or mitigating measures must be incorporated into existing buildings to achieve an equivilent level of protection.

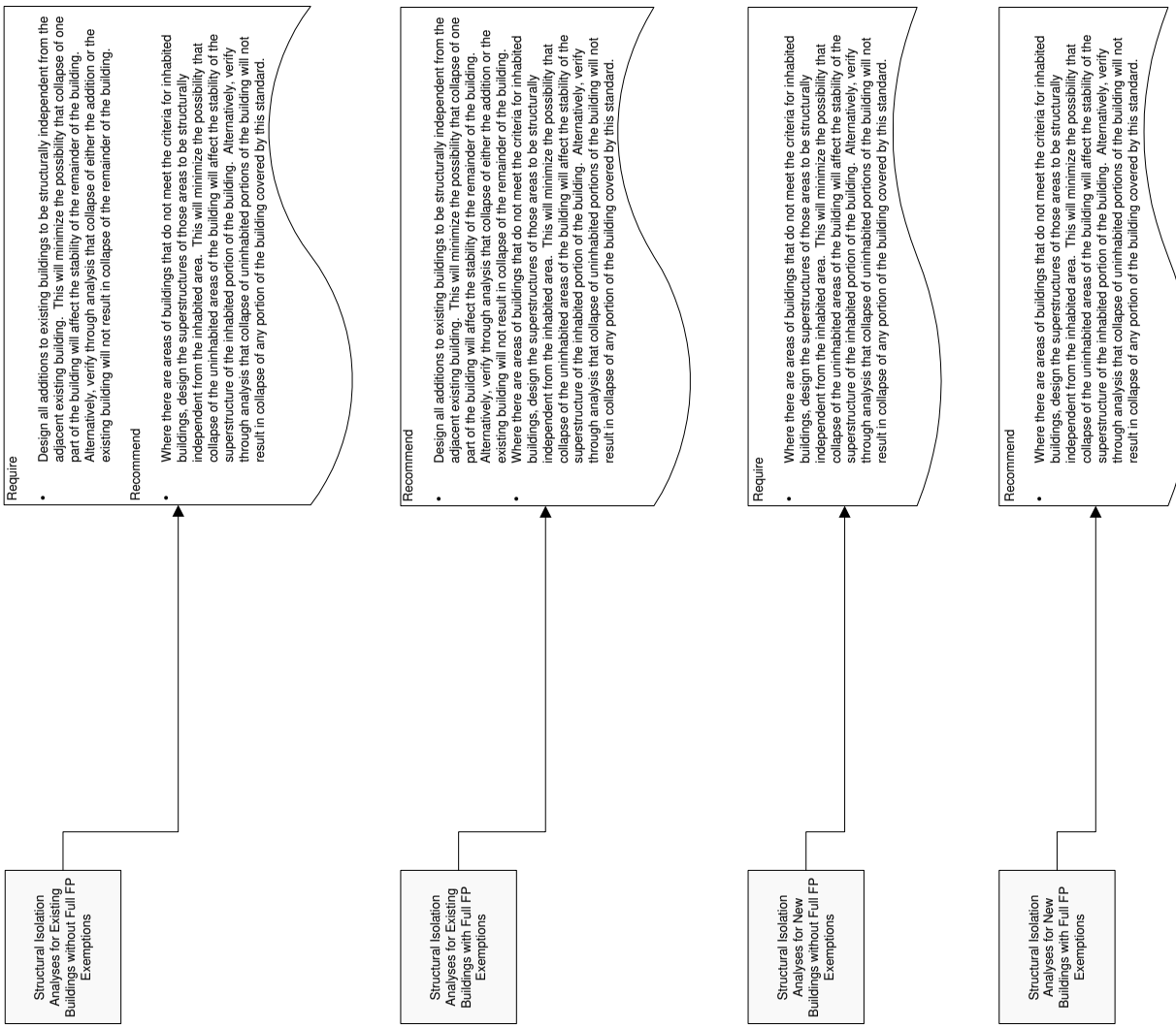
- Access Control. Ensure that access control measures are implemented to prohibit unauthorized personnel and vehicles from entering parking areas.
- Structural Elements. Ensure that the floors beneath or roofs above inhabited areas and all other adjacent supporting structural elements will not fail from the detonation in the parking area of an explosive equivalent to explosive weight II in Table B-1.
- Progressive Collapse. All structural elements within and adjacent to the parking area will be subject to all progressive collapse provisions of Standard 7 except that the exterior member removal provision will also apply to interior vertical or horizontal load carrying elements. Apply those provisions based on an explosive equivalent to explosive weight II in Table B-1.
-

Parking Beneath Buildings / on Rooftops Analyses for New and Existing Buildings with Full FP Exemptions

Recommend that parking beneath inhabited buildings or on rooftops of inhabited buildings be eliminated. Where very limited real estate makes such parking unavoidable, the following measures are recommended for incorporation into the design for new buildings or as mitigation measures for incorporation into existing buildings to achieve an equivilent level of protection.

- Access Control. Ensure that access control measures are implemented to prohibit unauthorized personnel and vehicles from entering parking areas.
- Structural Elements. Ensure that the floors beneath or roofs above inhabited areas and all other adjacent supporting structural elements will not fail from the detonation in the parking area of an explosive equivalent to explosive weight II in Table B-1.
- Progressive Collapse. All structural elements within and adjacent to the parking area should be subject to all progressive collapse provisions of Standard 7 except that the exterior member removal provision should also apply to interior vertical or horizontal load carrying elements. Apply those provisions based on an explosive equivalent to explosive weight II in Table B-1.


Progressive Collapse Analyses for New and Existing Buildings without Full FP Exemptions
does building have three or more stories (include basement w/exposed wall)
No
No Requirements
Yes
Require the following:
• Superstructure. Design the superstructure to sustain local damage with the structural system as a whole remaining stable and not being damaged to an extent disproportionate to the original local damage. Achieve this through an arrangement of the structural elements that provides stability to the entire structural system by transferring loads from any locally damaged region to adjacent regions capable of resisting those loads without collapse. Accomplish this by providing sufficient continuity, redundancy, or energy dissipating capacity (ductility, damping, hardness, etc.), or a combination thereof, in the members and connections of the structure.
• Columns and Walls. Design all exterior vertical load-carrying columns and walls to sustain a loss of lateral support at any of the floor levels by adding one story height to the nominal unsupported length. While this standard is based on the assumption of an external threat, where parking beneath buildings is unavoidable, this provision also applies to internal vertical load carrying columns and walls.
• Exterior Member Removal. Analyze the structure to ensure it can withstand removal of one primary exterior vertical or horizontal load-carrying element (i.e., a column or a beam) without progressive collapse.
• Floors. Design all floors with improved capacity to withstand load reversals due to explosive effects by designing them to withstand a net uplift equal to the dead load plus one-half the live load.
•
Progressive Collapse Analyses for New and Existing Buildings with Full FP Exemptions
does building have three or more stories (include basement w/exposed wall)
No
No Requirements
Yes
Recommend the following:
• Superstructure. Design the superstructure to sustain local damage with the structural system as a whole remaining stable and not being damaged to an extent disproportionate to the original local damage. Achieve this through an arrangement of the structural elements that provides stability to the entire structural system by transferring loads from any locally damaged region to adjacent regions capable of resisting those loads without collapse. Accomplish this by providing sufficient continuity, redundancy, or energy dissipating capacity (ductility, damping, hardness, etc.), or a combination thereof, in the members and connections of the structure.
• Columns and Walls. Design all exterior vertical load-carrying columns and walls to sustain a loss of lateral support at any of the floor levels by adding one story height to the nominal unsupported length. While this standard is based on the assumption of an external threat, where parking beneath buildings is unavoidable, this provision also applies to internal vertical load carrying columns and walls.
• Exterior Member Removal. Analyze the structure to ensure it can withstand removal of one primary exterior vertical or horizontal load-carrying element (i.e., a column or a beam) without progressive collapse.
• Floors. Design all floors with improved capacity to withstand load reversals due to explosive effects by designing them to withstand a net uplift equal to the dead load plus one-half the live load.
•


Structural Isolation Analyses for Existing Buildings without Full FP Exemptions
Require
• Design all additions to existing buildings to be structurally independent from the adjacent existing building. This will minimize the possibility that collapse of one part of the building will affect the stability of the remainder of the building. Alternatively, verify through analysis that collapse of either the addition or the existing building will not result in collapse of the remainder of the building.
Recommend
• Where there are areas of buildings that do not meet the criteria for inhabited buildings, design the superstructures of those areas to be structurally independent from the inhabited area. This will minimize the possibility that collapse of the uninhabited areas of the building will affect the stability of the superstructure of the inhabited portion of the building. Alternatively, verify through analysis that collapse of uninhabited portions of the building will not result in collapse of any portion of the building covered by this standard.
Structural Isolation Analyses for Existing Buildings with Full FP Exemptions
Recommend
• Design all additions to existing buildings to be structurally independent from the adjacent existing building. This will minimize the possibility that collapse of one part of the building will affect the stability of the remainder of the building. Alternatively, verify through analysis that collapse of either the addition or the existing building will not result in collapse of the remainder of the building.
• Where there are areas of buildings that do not meet the criteria for inhabited buildings, design the superstructures of those areas to be structurally independent from the inhabited area. This will minimize the possibility that collapse of the uninhabited areas of the building will affect the stability of the superstructure of the inhabited portion of the building. Alternatively, verify through analysis that collapse of uninhabited portions of the building will not result in collapse of any portion of the building covered by this standard.
Structural Isolation Analyses for New Buildings without Full FP Exemptions
Require
• Where there are areas of buildings that do not meet the criteria for inhabited buildings, design the superstructures of those areas to be structurally independent from the inhabited area. This will minimize the possibility that collapse of the uninhabited areas of the building will affect the stability of the superstructure of the inhabited portion of the building. Alternatively, verify through analysis that collapse of uninhabited portions of the building will not result in collapse of any portion of the building covered by this standard.
Structural Isolation Analyses for New Buildings with Full FP Exemptions
Recommend
• Where there are areas of buildings that do not meet the criteria for inhabited buildings, design the superstructures of those areas to be structurally independent from the inhabited area. This will minimize the possibility that collapse of the uninhabited areas of the building will affect the stability of the superstructure of the inhabited portion of the building. Alternatively, verify through analysis that collapse of uninhabited portions of the building will not result in collapse of any portion of the building covered by this standard.

Building Overhang Analyses for New and Existing Buildings without Full FP Exemptions

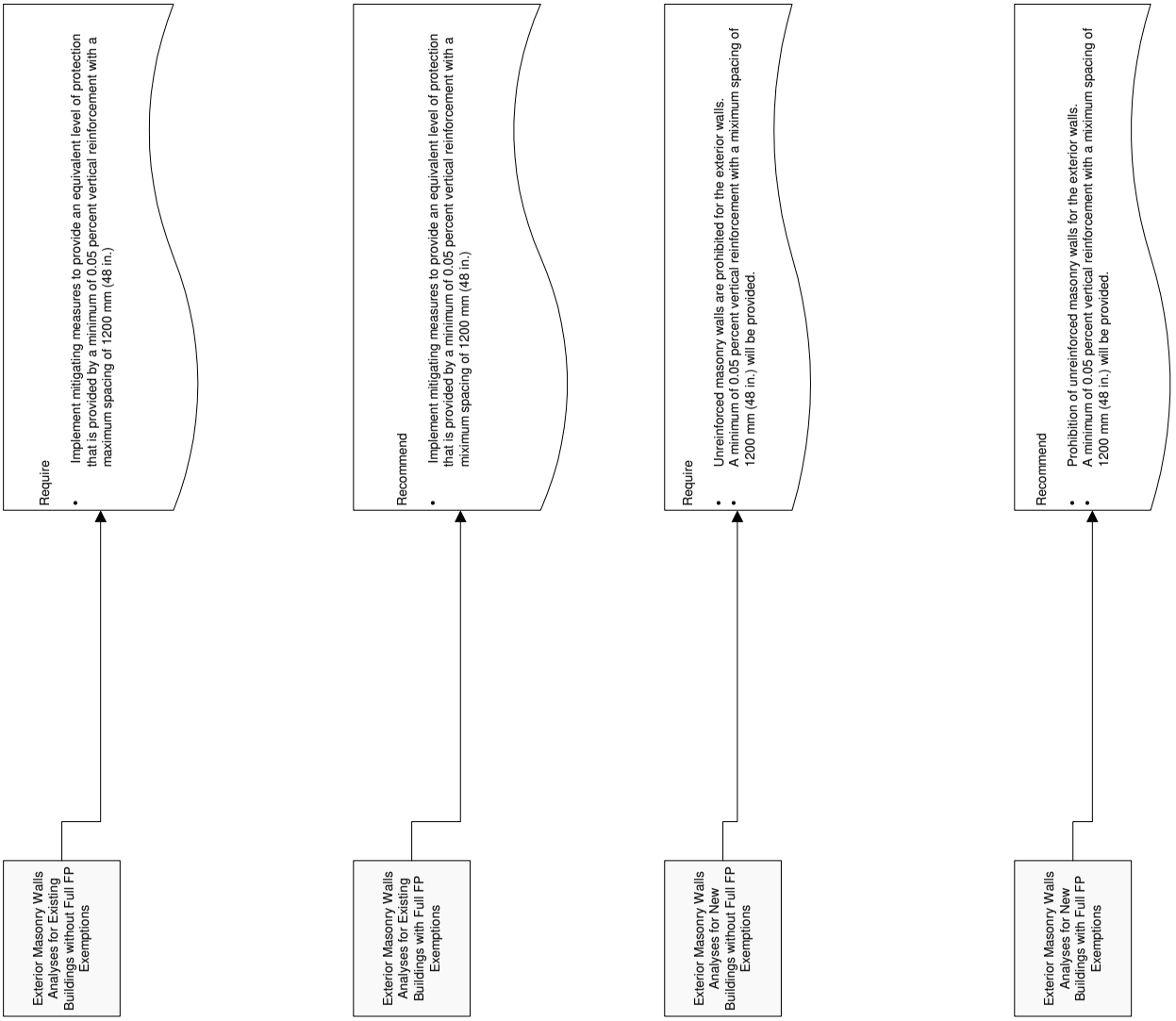
Avoid building overhangs with inhabited spaces above them where people could gain access to the area underneath the overhang. Where such overhangs must be used, require the incorporation of the following measures into the design for new buildings or for existing buildings require the incorporation of mitigating measures to achieve an equivalent level of protection.

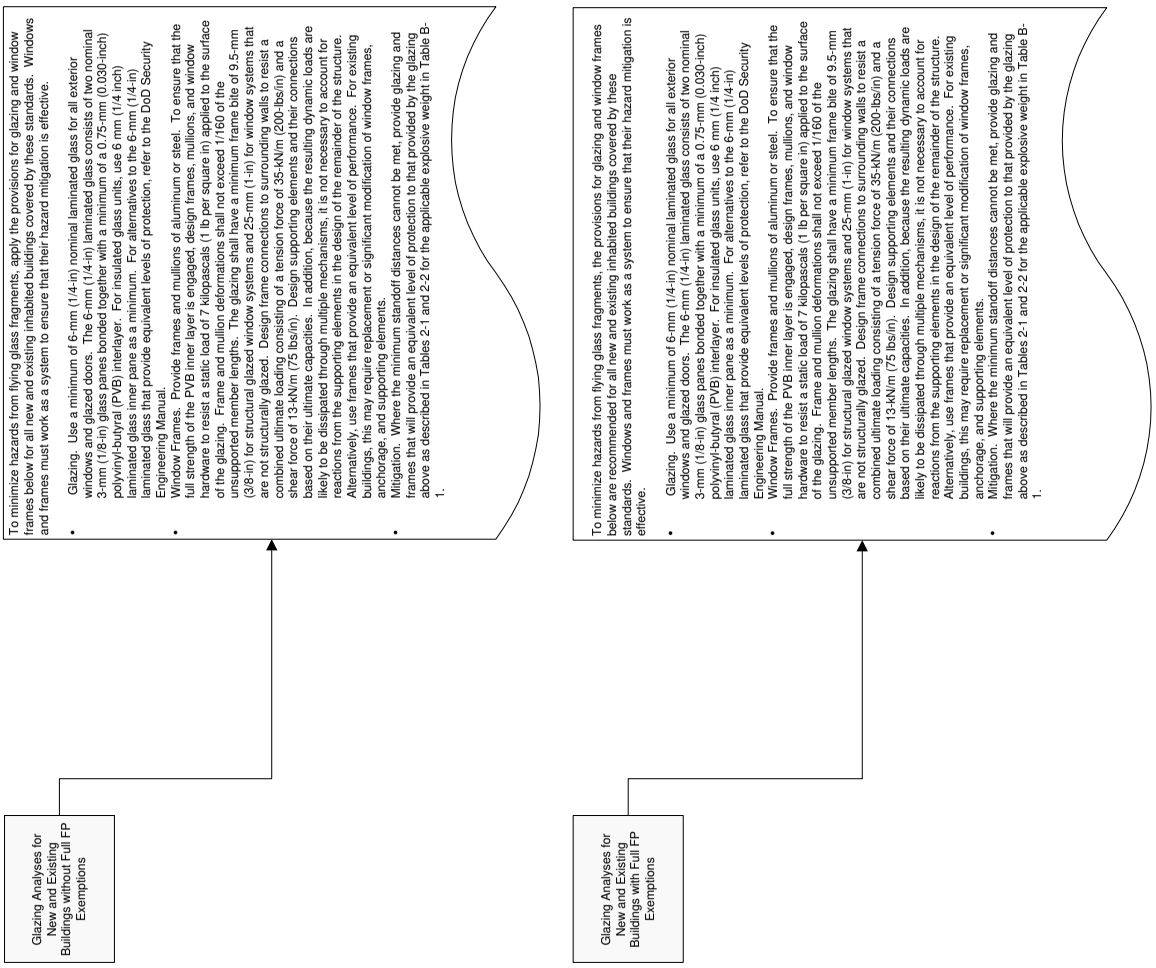
- Parking and Roadway Restrictions. Ensure that there are no roadways or parking areas under overhangs.
- Floors. Ensure that the floors beneath inhabited areas will not fail from the detonation underneath the overhang of an explosive equivalent to explosive weight II where there is a controlled perimeter and explosive weight I for an uncontrolled perimeter. Explosive weights I and II are identified in Table B-1.
- Superstructure. The progressive collapse provisions of Standard 7, including the provision for loss of lateral support for vertical load carrying elements, will include all structural elements within and adjacent to the overhang.

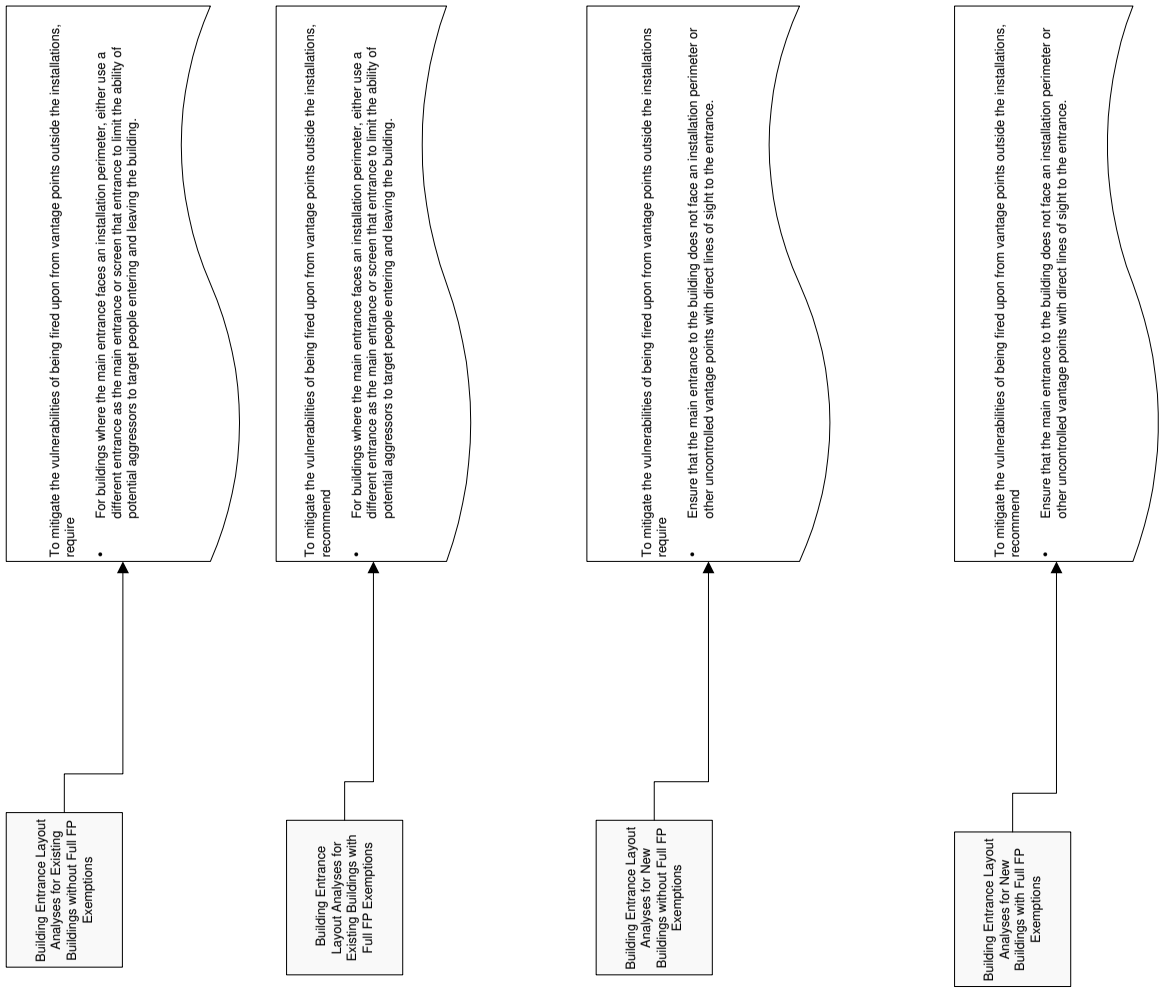
Building Overhang Analyses for New and Existing Buildings with Full FP Exemptions

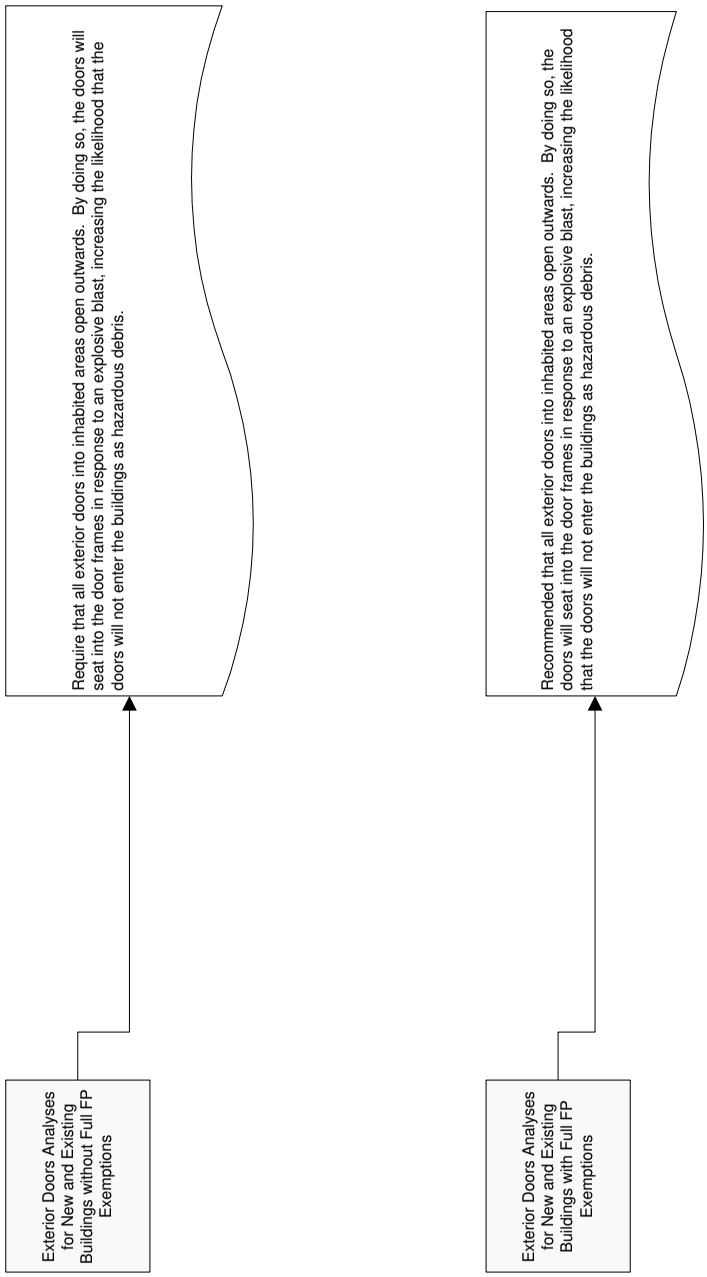
Avoid building overhangs with inhabited spaces above them where people could gain access to the area underneath the overhang. Where such overhangs must be used, recommend the incorporation of the following measures into the design for new buildings or for existing buildings recommend the incorporation of mitigating measures to achieve an equivalent level of protection.

- Parking and Roadway Restrictions. Ensure that there are no roadways or parking areas under overhangs.
- Floors. Ensure that the floors beneath inhabited areas will not fail from the detonation underneath the overhang of an explosive equivalent to explosive weight II where there is a controlled perimeter and explosive weight I for an uncontrolled perimeter. Explosive weights I and II are identified in Table B-1.
- Superstructure. The progressive collapse provisions of Standard 7, including the provision for loss of lateral support for vertical load carrying elements, will include all structural elements within and adjacent to the overhang.


Exterior Masonry Walls Analyses for Existing Buildings without Full FP Exemptions
Require
Implement mitigating measures to provide an equivalent level of protection that is provided by a minimum of 0.05 percent vertical reinforcement with a maximum spacing of 1200 mm (48 in.)
Exterior Masonry Walls Analyses for Existing Buildings with Full FP Exemptions
Recommend
Implement mitigating measures to provide an equivalent level of protection that is provided by a minimum of 0.05 percent vertical reinforcement with a mixiumum spacing of 1200 mm (48 in.)
Exterior Masonry Walls Analyses for New Buildings without Full FP Exemptions
Require
Unreinforced masonry walls are prohibited for the exterior walls.
A minimum of 0.05 percent vertical reinforcement with a mixiumum spacing of 1200 mm (48 in.) will be provided.
Exterior Masonry Walls Analyses for New Buildings with Full FP Exemptions
Recommend
Prohibition of unreinforced masonry walls for the exterior walls.
A minimum of 0.05 percent vertical reinforcement with a mixiumum spacing of 1200 mm (48 in.) will be provided.


Glazing Analyses for New and Existing Buildings without Full FP Exemptions
To minimize hazards from flying glass fragments, apply the provisions for glazing and window frames below for all new and existing inhabited buildings covered by these standards. Windows and frames must work as a system to ensure that their hazard mitigation is effective.
Glazing. Use a minimum of 6-mm (1/4-in) nominal laminated glass for all exterior windows and glazed doors. The 6-mm (1/4-in) laminated glass consists of two nominal 3-mm (1/8-in) glass panes bonded together with a minimum of a 0.75-mm (0.030-inch) polyvinyl-butyral (PVB) interlayer. For insulated glass units, use 6 mm (1/4 inch) laminated glass inner pane as a minimum. For alternatives to the 6-mm (1/4-in) laminated glass that provide equivalent levels of protection, refer to the DoD Security Engineering Manual.
Window Frames. Provide frames and mullions of aluminum or steel. To ensure that the full strength of the PVB inner layer is engaged, design frames, mullions, and window hardware to resist a static load of 7 kilopascals (1 lb per square in) applied to the surface of the glazing. Frame and mullion deformations shall not exceed 1/160 of the unsupported member lengths. The glazing shall have a minimum frame bite of 9.5-mm (3/8-in) for structural glazed window systems and 25-mm (1-in) for window systems that are not structurally glazed. Design frame connections to surrounding walls to resist a combined ultimate loading consisting of a tension force of 35-kN/m (200-lbs/in) and a shear force of 13-kN/m (75 lbs/in). Design supporting elements and their connections based on their ultimate capacities. In addition, because the resulting dynamic loads are likely to be dissipated through multiple mechanisms, it is not necessary to account for reactions from the supporting elements in the design of the remainder of the structure. Alternatively, use frames that provide an equivalent level of performance. For existing buildings, this may require replacement or significant modification of window frames, anchorage, and supporting elements.
Mitigation. Where the minimum standoff distances cannot be met, provide glazing and frames that will provide an equivalent level of protection to that provided by the glazing above as described in Tables 2-1 and 2-2 for the applicable explosive weight in Table B-1.
Glazing Analyses for New and Existing Buildings with Full FP Exemptions
To minimize hazards from flying glass fragments, the provisions for glazing and window frames below are recommended for all new and existing inhabited buildings covered by these standards. Windows and frames must work as a system to ensure that their hazard mitigation is effective.
Glazing. Use a minimum of 6-mm (1/4-in) nominal laminated glass for all exterior windows and glazed doors. The 6-mm (1/4-in) laminated glass consists of two nominal 3-mm (1/8-in) glass panes bonded together with a minimum of a 0.75-mm (0.030-inch) polyvinyl-butyral (PVB) interlayer. For insulated glass units, use 6 mm (1/4 inch) laminated glass inner pane as a minimum. For alternatives to the 6-mm (1/4-in) laminated glass that provide equivalent levels of protection, refer to the DoD Security Engineering Manual.
Window Frames. Provide frames and mullions of aluminum or steel. To ensure that the full strength of the PVB inner layer is engaged, design frames, mullions, and window hardware to resist a static load of 7 kilopascals (1 lb per square in) applied to the surface of the glazing. Frame and mullion deformations shall not exceed 1/160 of the unsupported member lengths. The glazing shall have a minimum frame bite of 9.5-mm (3/8-in) for structural glazed window systems and 25-mm (1-in) for window systems that are not structurally glazed. Design frame connections to surrounding walls to resist a combined ultimate loading consisting of a tension force of 35-kN/m (200-lbs/in) and a shear force of 13-kN/m (75 lbs/in). Design supporting elements and their connections based on their ultimate capacities. In addition, because the resulting dynamic loads are likely to be dissipated through multiple mechanisms, it is not necessary to account for reactions from the supporting elements in the design of the remainder of the structure. Alternatively, use frames that provide an equivalent level of performance. For existing buildings, this may require replacement or significant modification of window frames, anchorage, and supporting elements.
Mitigation. Where the minimum standoff distances cannot be met, provide glazing and frames that will provide an equivalent level of protection to that provided by the glazing above as described in Tables 2-1 and 2-2 for the applicable explosive weight in Table B-1.


Building Entrance Layout Analyses for Existing Buildings without Full FP Exemptions
To mitigate the vulnerabilities of being fired upon from vantage points outside the installations, require
For buildings where the main entrance faces an installation perimeter, either use a different entrance as the main entrance or screen that entrance to limit the ability of potential aggressors to target people entering and leaving the building.
Building Entrance Layout Analyses for Existing Buildings with Full FP Exemptions
To mitigate the vulnerabilities of being fired upon from vantage points outside the installations, recommend
For buildings where the main entrance faces an installation perimeter, either use a different entrance as the main entrance or screen that entrance to limit the ability of potential aggressors to target people entering and leaving the building.
Building Entrance Layout Analyses for New Buildings without Full FP Exemptions
To mitigate the vulnerabilities of being fired upon from vantage points outside the installations require
Ensure that the main entrance to the building does not face an installation perimeter or other uncontrolled vantage points with direct lines of sight to the entrance.
Building Entrance Layout Analyses for New Buildings with Full FP Exemptions
To mitigate the vulnerabilities of being fired upon from vantage points outside the installations, recommend
Ensure that the main entrance to the building does not face an installation perimeter or other uncontrolled vantage points with direct lines of sight to the entrance.


Exterior Doors Analyses for New and Existing Buildings without Full FP Exemptions
Require that all exterior doors into inhabited areas open outwards. By doing so, the doors will seat into the door frames in response to an explosive blast, increasing the likelihood that the doors will not enter the buildings as hazardous debris.
Exterior Doors Analyses for New and Existing Buildings with Full FP Exemptions
Recommended that all exterior doors into inhabited areas open outwards. By doing so, the doors will seat into the door frames in response to an explosive blast, increasing the likelihood that the doors will not enter the buildings as hazardous debris.

Mailroom Analyses for New and Existing Buildings without Full FP Exemptions

The following measures address the location of rooms to which mail is delivered or in which mail is handled in new and existing inhabited buildings. The measures involve limiting collateral damage and injuries and facilitating future upgrades to enhance protection should they become necessary.

Require

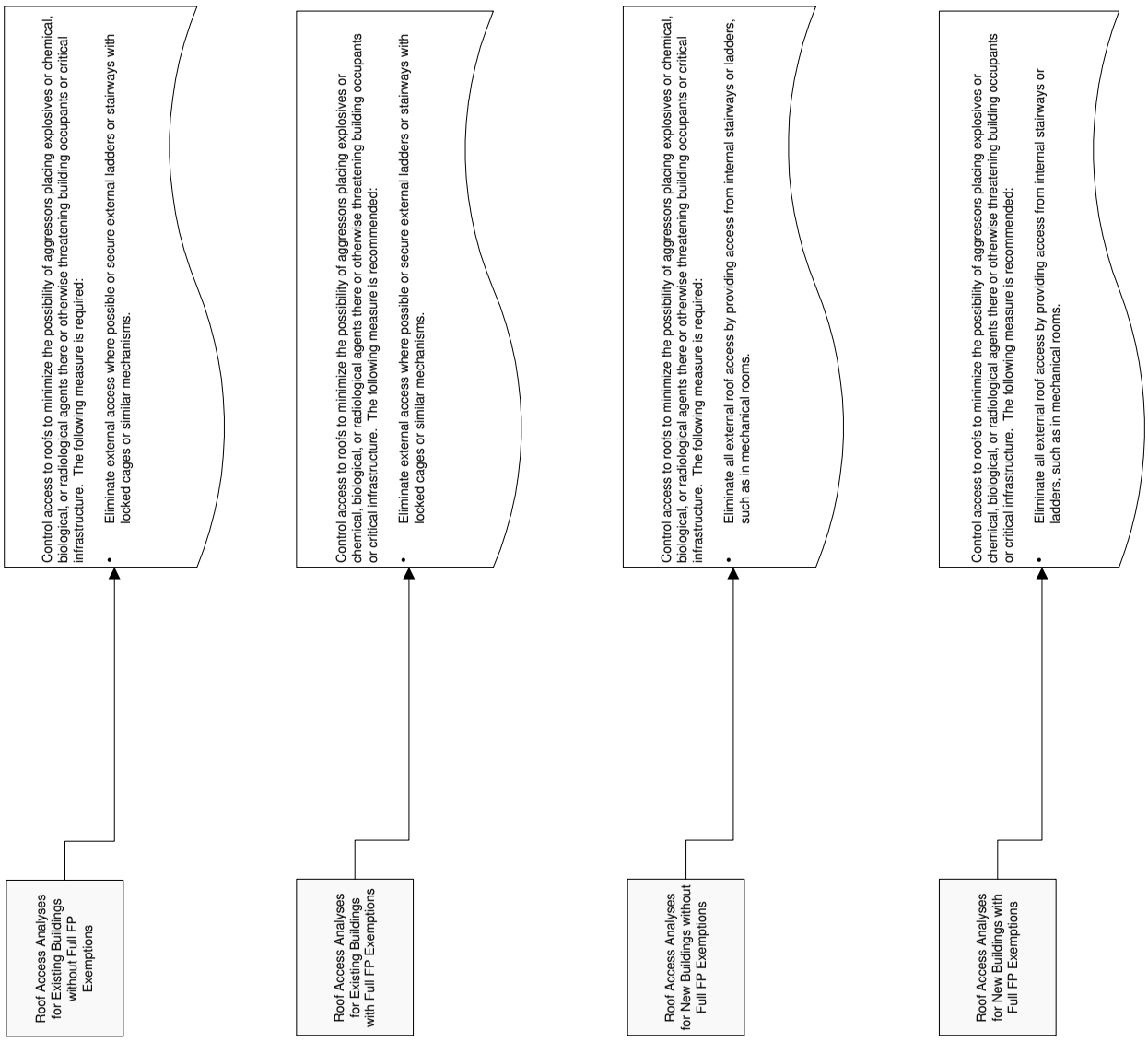
- Location. Where a building must have a mailroom, locate that mailroom on the perimeter of the building. By locating the mailroom on the building perimeter there is an opportunity to modify it in the future if a mail bomb threat is identified. Where mailrooms are located in the interior of buildings, few retrofit options are available for mitigating the mail bomb threat.
- Proximity. Locate mailrooms as far from heavily populated areas of the building and critical infrastructure as possible. This measure will minimize injuries and damage if a mail bomb detonates in the mailroom. Further, it will reduce the potential for wider dissemination of hazardous agents. These apply where the mailroom is not specifically designed to resist those threats.
- Sealing. To limit migration into buildings of airborne chemical, biological, and radiological agents introduced into mailrooms, ensure that mailrooms are well sealed between their envelopes and other portions of the buildings in which they are located. Ensure the mailroom walls are of full height construction that fully extends and is sealed to the undersides of the roofs, to the undersides of any floors above them, or to hard ceilings (i.e. gypsum wallboard ceiling.) Sealing should include visible cracks, the interface joints beween walls and ceilings/roofs and all wall and ceiling/roof penetrations. Doors will have weather stripping on all four edges.

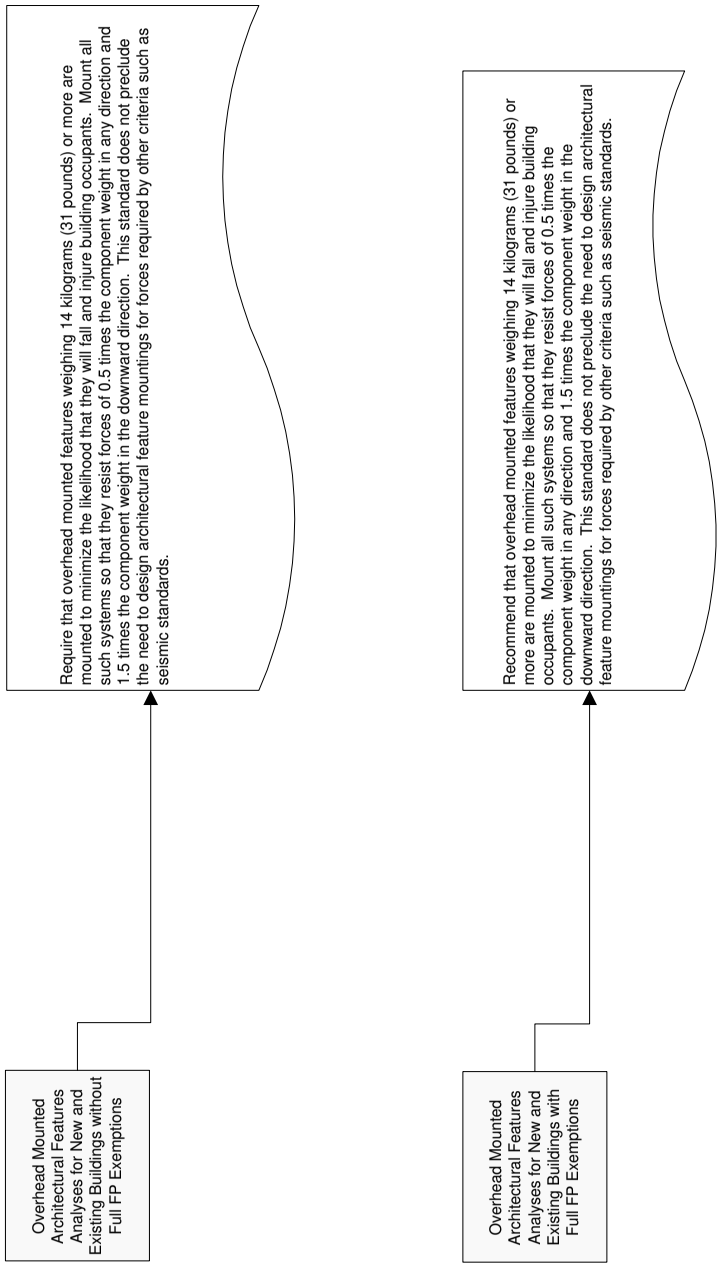
Mailroom Analyses for New and Existing Buildings with Full FP Exemptions

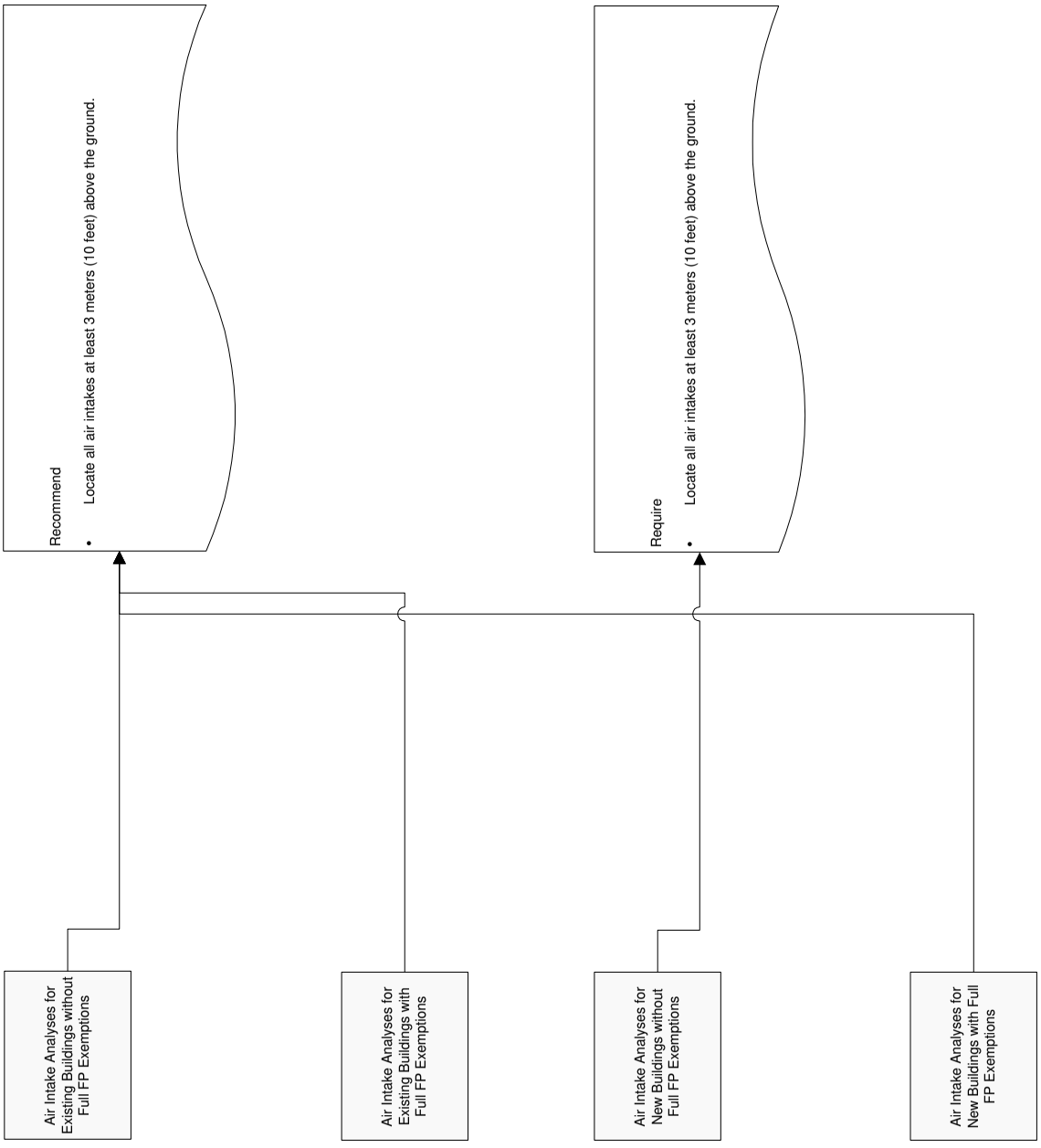
The following measures address the location of rooms to which mail is delivered or in which mail is handled in new and existing inhabited buildings. The measures involve limiting collateral damage and injuries and facilitating future upgrades to enhance protection should they become necessary.

Recommend

- Location. Where a building must have a mailroom, locate that mailroom on the perimeter of the building. By locating the mailroom on the building perimeter there is an opportunity to modify it in the future if a mail bomb threat is identified. Where mailrooms are located in the interior of buildings, few retrofit options are available for mitigating the mail bomb threat.
- Proximity. Locate mailrooms as far from heavily populated areas of the building and critical infrastructure as possible. This measure will minimize injuries and damage if a mail bomb detonates in the mailroom. Further, it will reduce the potential for wider dissemination of hazardous agents. These apply where the mailroom is not specifically designed to resist those threats.
- Sealing. To limit migration into buildings of airborne chemical, biological, and radiological agents introduced into mailrooms, ensure that mailrooms are well sealed between their envelopes and other portions of the buildings in which they are located. Ensure the mailroom walls are of full height construction that fully extends and is sealed to the undersides of the roofs, to the undersides of any floors above them, or to hard ceilings (i.e. gypsum wallboard ceiling.) Sealing should include visible cracks, the interface joints beween walls and ceilings/roofs and all wall and ceiling/roof penetrations. Doors will have weather stripping on all four edges.


Roof Access Analyses for Existing Buildings without Full FP Exemptions
Control access to roofs to minimize the possibility of aggressors placing explosives or chemical, biological, or radiological agents there or otherwise threatening building occupants or critical infrastructure. The following measure is required:
• Eliminate external access where possible or secure external ladders or stairways with locked cages or similar mechanisms.
Roof Access Analyses for Existing Buildings with Full FP Exemptions
Control access to roofs to minimize the possibility of aggressors placing explosives or chemical, biological, or radiological agents there or otherwise threatening building occupants or critical infrastructure. The following measure is recommended:
• Eliminate external access where possible or secure external ladders or stairways with locked cages or similar mechanisms.
Roof Access Analyses for New Buildings without Full FP Exemptions
Control access to roofs to minimize the possibility of aggressors placing explosives or chemical, biological, or radiological agents there or otherwise threatening building occupants or critical infrastructure. The following measure is required:
• Eliminate all external roof access by providing access from internal stairways or ladders, such as in mechanical rooms.
Roof Access Analyses for New Buildings with Full FP Exemptions
Control access to roofs to minimize the possibility of aggressors placing explosives or chemical, biological, or radiological agents there or otherwise threatening building occupants or critical infrastructure. The following measure is recommended:
• Eliminate all external roof access by providing access from internal stairways or ladders, such as in mechanical rooms.


Overhead Mounted Architectural Features Analyses for New and Existing Buildings without Full FP Exemptions
Require that overhead mounted features weighing 14 kilograms (31 pounds) or more are mounted to minimize the likelihood that they will fall and injure building occupants. Mount all such systems so that they resist forces of 0.5 times the component weight in any direction and 1.5 times the component weight in the downward direction. This standard does not preclude the need to design architectural feature mountings for forces required by other criteria such as seismic standards.
Overhead Mounted Architectural Features Analyses for New and Existing Buildings with Full FP Exemptions
Recommend that overhead mounted features weighing 14 kilograms (31 pounds) or more are mounted to minimize the likelihood that they will fall and injure building occupants. Mount all such systems so that they resist forces of 0.5 times the component weight in any direction and 1.5 times the component weight in the downward direction. This standard does not preclude the need to design architectural feature mountings for forces required by other criteria such as seismic standards.


Recommend
• Locate all air intakes at least 3 meters (10 feet) above the ground.
Require
• Locate all air intakes at least 3 meters (10 feet) above the ground.
Air Intake Analyses for Existing Buildings without Full FP Exemptions
Air Intake Analyses for Existing Buildings with Full FP Exemptions
Air Intake Analyses for New Buildings without Full FP Exemptions
Air Intake Analyses for New Buildings with Full FP Exemptions

Emergency Air Distribution Shutoff Analyses for New and Existing Buildings without Full FP Exemptions

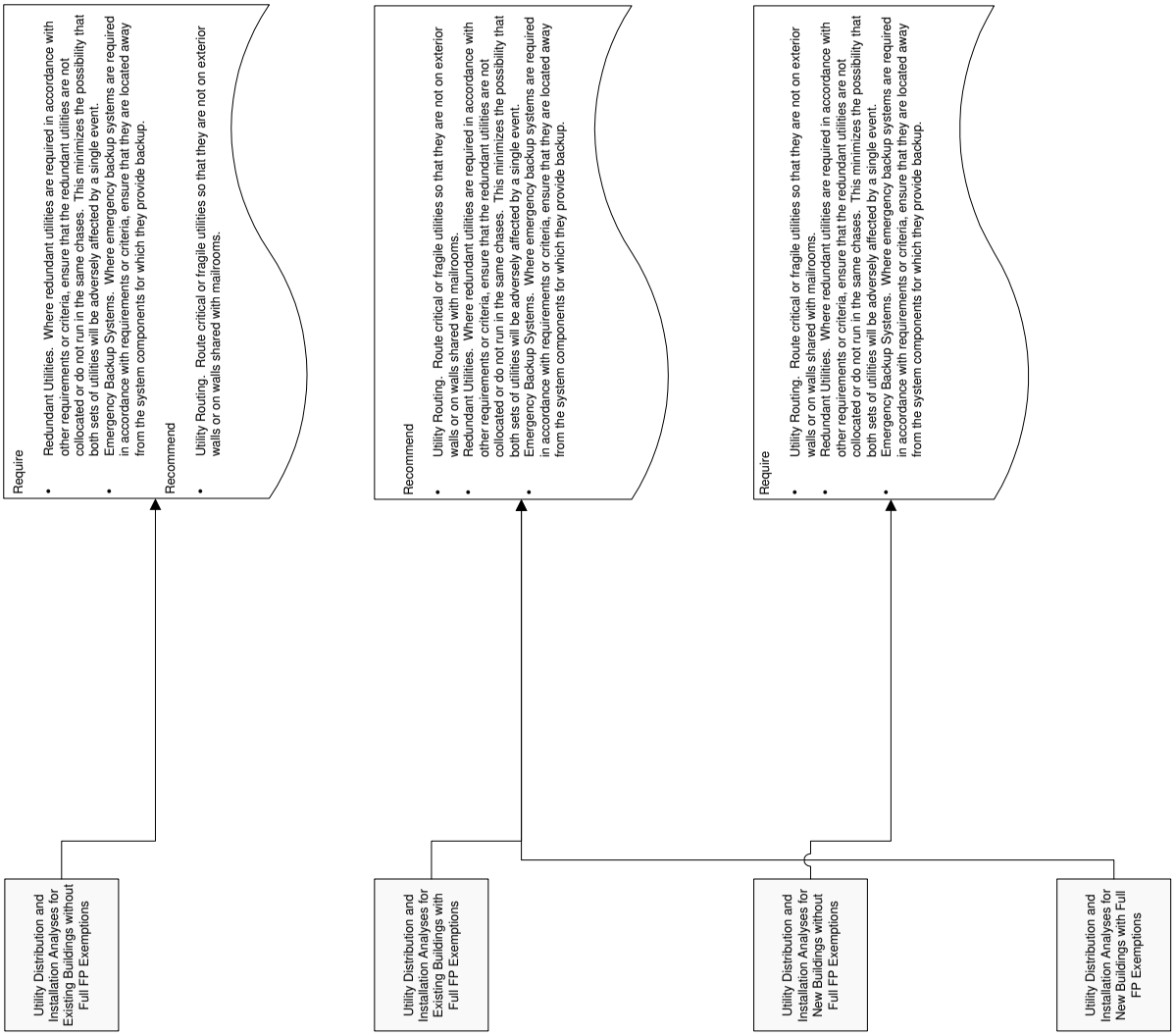
Require

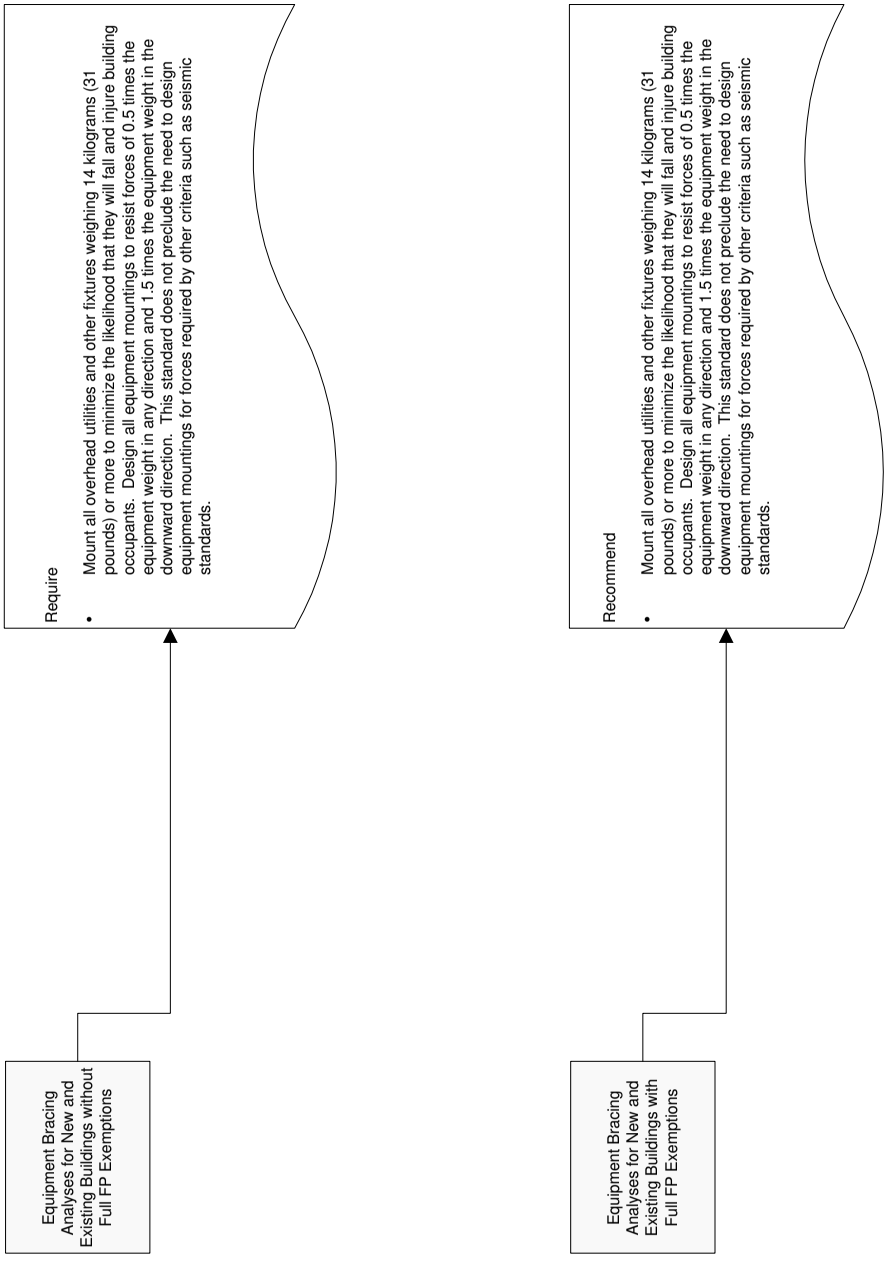
- Provide an emergency shutoff switch in the HVAC control system that can immediately shut down air distribution throughout the building, except where interior pressure and airflow control would more efficiently prevent the spread of airborne contaminants and/or ensure the safety of egress pathways. Locate the switch (or switches) to be easily accessible by building occupants. Providing such a capability will allow the facility manager or building security manager to limit the distribution of airborne contaminants that may be introduced into the building.

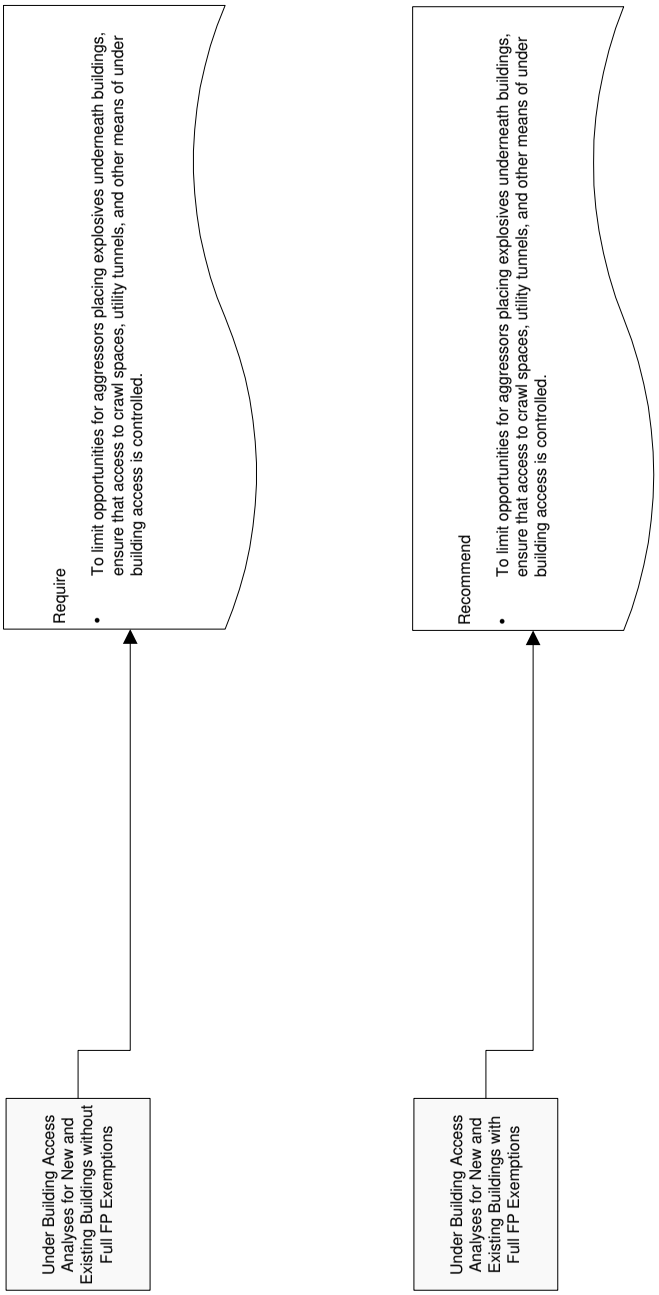
Emergency Air Distribution Shutoff Analyses for New and Existing Buildings with Full FP Exemptions

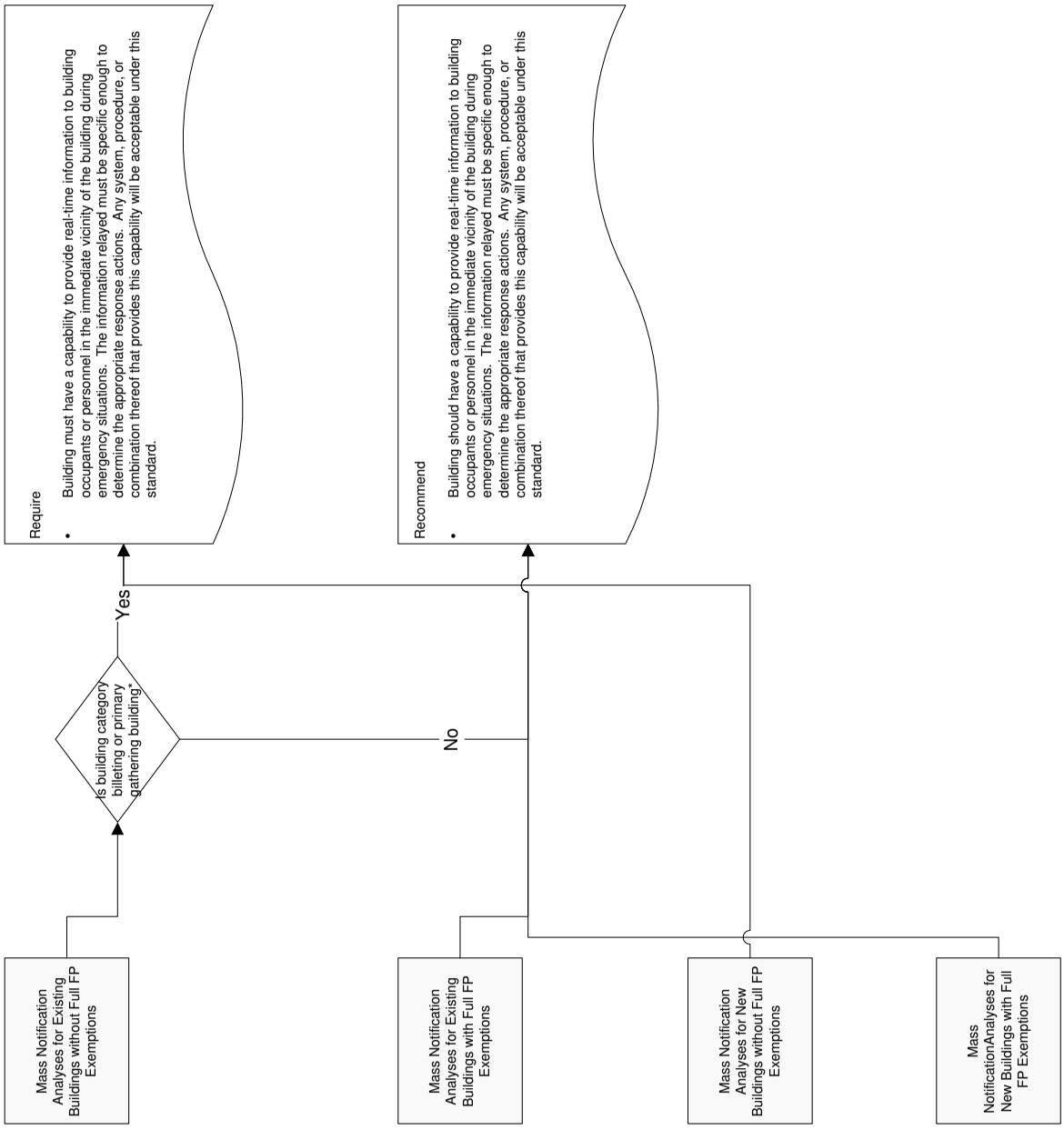
Recommend

- Provide an emergency shutoff switch in the HVAC control system that can immediately shut down air distribution throughout the building, except where interior pressure and airflow control would more efficiently prevent the spread of airborne contaminants and/or ensure the safety of egress pathways. Locate the switch (or switches) to be easily accessible by building occupants. Providing such a capability will allow the facility manager or building security manager to limit the distribution of airborne contaminants that may be introduced into the building.
-


Require
Redundant Utilities. Where redundant utilities are required in accordance with other requirements or criteria, ensure that the redundant utilities are not collocated or do not run in the same chases. This minimizes the possibility that both sets of utilities will be adversely affected by a single event.
Emergency Backup Systems. Where emergency backup systems are required in accordance with requirements or criteria, ensure that they are located away from the system components for which they provide backup.
Recommend
Utility Routing. Route critical or fragile utilities so that they are not on exterior walls or on walls shared with mailrooms.
Recommend
Utility Routing. Route critical or fragile utilities so that they are not on exterior walls or on walls shared with mailrooms.
Redundant Utilities. Where redundant utilities are required in accordance with other requirements or criteria, ensure that the redundant utilities are not collocated or do not run in the same chases. This minimizes the possibility that both sets of utilities will be adversely affected by a single event.
Emergency Backup Systems. Where emergency backup systems are required in accordance with requirements or criteria, ensure that they are located away from the system components for which they provide backup.
Require
Utility Routing. Route critical or fragile utilities so that they are not on exterior walls or on walls shared with mailrooms.
Redundant Utilities. Where redundant utilities are required in accordance with other requirements or criteria, ensure that the redundant utilities are not collocated or do not run in the same chases. This minimizes the possibility that both sets of utilities will be adversely affected by a single event.
Emergency Backup Systems. Where emergency backup systems are required in accordance with requirements or criteria, ensure that they are located away from the system components for which they provide backup.
Utility Distribution and Installation Analyses for Existing Buildings without Full FP Exemptions
Utility Distribution and Installation Analyses for Existing Buildings with Full FP Exemptions
Utility Distribution and Installation Analyses for New Buildings without Full FP Exemptions
Utility Distribution and Installation Analyses for New Buildings with Full FP Exemptions


Equipment Bracing Analyses for New and Existing Buildings without Full FP Exemptions
Require
• Mount all overhead utilities and other fixtures weighing 14 kilograms (31 pounds) or more to minimize the likelihood that they will fall and injure building occupants. Design all equipment mountings to resist forces of 0.5 times the equipment weight in any direction and 1.5 times the equipment weight in the downward direction. This standard does not preclude the need to design equipment mountings for forces required by other criteria such as seismic standards.
Equipment Bracing Analyses for New and Existing Buildings with Full FP Exemptions
Recommend
• Mount all overhead utilities and other fixtures weighing 14 kilograms (31 pounds) or more to minimize the likelihood that they will fall and injure building occupants. Design all equipment mountings to resist forces of 0.5 times the equipment weight in any direction and 1.5 times the equipment weight in the downward direction. This standard does not preclude the need to design equipment mountings for forces required by other criteria such as seismic standards.


Under Building Access Analyses for New and Existing Buildings without Full FP Exemptions
Require
• To limit opportunities for aggressors placing explosives underneath buildings, ensure that access to crawl spaces, utility tunnels, and other means of under building access is controlled.
Under Building Access Analyses for New and Existing Buildings with Full FP Exemptions
Recommend
• To limit opportunities for aggressors placing explosives underneath buildings, ensure that access to crawl spaces, utility tunnels, and other means of under building access is controlled.


Require
Building must have a capability to provide real-time information to building occupants or personnel in the immediate vicinity of the building during emergency situations. The information relayed must be specific enough to determine the appropriate response actions. Any system, procedure, or combination thereof that provides this capability will be acceptable under this standard.
Recommend
Building should have a capability to provide real-time information to building occupants or personnel in the immediate vicinity of the building during emergency situations. The information relayed must be specific enough to determine the appropriate response actions. Any system, procedure, or combination thereof that provides this capability will be acceptable under this standard.
Yes
No
Is building category billeting or primary gathering building?
Mass Notification Analyses for Existing Buildings without Full FP Exemptions
Mass Notification Analyses for Existing Buildings with Full FP Exemptions
Mass Notification Analyses for New Buildings without Full FP Exemptions
Mass NotificationAnalyses for New Buildings with Full FP Exemptions

Glazing Analyses for Glazing Replacement Project
To minimize hazards from flying glass fragments, apply the provisions for glazing and window frames below for all Windows or Doors Glazing Replacement projects. Windows and frames must work as a system to ensure that their hazard mitigation is effective.
• Glazing. Use a minimum of 6-mm (1/4-in) nominal laminated glass for all exterior windows and glazed doors. The 6-mm (1/4-in) laminated glass consists of two nominal 3-mm (1/8-in) glass panes bonded together with a minimum of a 0.75-mm (0.030-inch) polyvinyl-butyral (PVB) interlayer. For insulated glass units, use 6 mm (1/4 inch) laminated glass inner pane as a minimum. For alternatives to the 6-mm (1/4-in) laminated glass that provide equivalent levels of protection, refer to the DoD Security Engineering Manual.
• Window Frames. Provide frames and mullions of aluminum or steel. To ensure that the full strength of the PVB inner layer is engaged, design frames, mullions, and window hardware to resist a static load of 7 kilopascals (1 lb per square in) applied to the surface of the glazing. Frame and mullion deformations shall not exceed 1/160 of the unsupported member lengths. The glazing shall have a minimum frame bite of 9.5-mm (3/8-in) for structural glazed window systems and 25-mm (1-in) for window systems that are not structurally glazed. Design frame connections to surrounding walls to resist a combined ultimate loading consisting of a tension force of 35-kN/m (200-lbs/in) and a shear force of 13-kN/m (75 lbs/in). Design supporting elements and their connections based on their ultimate capacities. In addition, because the resulting dynamic loads are likely to be dissipated through multiple mechanisms, it is not necessary to account for reactions from the supporting elements in the design of the remainder of the structure. Alternatively, use frames that provide an equivalent level of performance. For existing buildings, this may require replacement or significant modification of window frames, anchorage, and supporting elements.
• Mitigation. Where the minimum standoff distances cannot be met, provide glazing and frames that will provide an equivalent level of protection to that provided by the glazing above as described in Tables 2-1 and 2-2 for the applicable explosive weight in Table B-1.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (From - To)
11-2004	Final	

4. TITLE AND SUBTITLE
Facility Composer Design Wizards:
A Method for Extensible Codified Design Logic Based on Explicit Facility Criteria

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Van J. Woods, Susan D. Nachtigall, Beth A. Brucker, and Awilda Andrillion

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER
LK6K75-N

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
U.S. Army Engineer Research and Development Center (ERDC)
Construction Engineering Research Laboratory (CERL)
PO Box 9005
Champaign, IL 61826-9005

8. PERFORMING ORGANIZATION REPORT NUMBER
ERDC/CERL TR-04-22

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Engineer Research and Development Center (ERDC)
Construction Engineering Research Laboratory (CERL)
PO Box 9005
Champaign, IL 61826-9005

10. SPONSOR/MONITOR'S ACRONYM(S)
CEERD-CV-T

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES
Copies are available from the National Technical Information Service, 5285 Port Royal Road, Springfield, VA 22161.

14. ABSTRACT

Government design criteria is commonly captured in the form of design guides, regulations, technical manuals, and web pages, but not in a computable format. Current design systems provide no way to directly interact with a specific criterion, or to efficiently extend the functionality of an application to directly support criteria usage. Consequently, designers must ensure that all applicable criteria are identified and satisfied, or that a large customized application is developed. Custom systems are slow to develop and change, and difficult to update. Such systems do allow data modularization, but do not provide modular functionality—the ability to support customized methods or algorithms that perform useful operations on the data. The *Facility Composer* suite of tools supports the capturing and tracking of facility criteria and requirements, planning and design charrettes, and associated planning and design analyses. *Facility Composer* addresses many of the problems associated with the decentralized, non-computationally explicit, ad-hoc definition, distribution, and utilization of design criteria. This report described work undertaken to provide a set of the most commonly used tools as part of the core features in *Facility Composer*, and also to provide a means for modularized extensibility.

15. SUBJECT TERMS

fort future	facility composer	software
facility planning	facility design	

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Susan D. Nachtigall
Unclassified	Unclassified	Unclassified	SAR	128	19b. TELEPHONE NUMBER (include area code) 217-373-4579