CA Gen

Host Encyclopedia Subsetting User Guide

Release 8.5

CA Technologies Product References

This document references the following CA Technologies products:

- CA Gen

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At http://ca.com/support, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at http://ca.com/docs.

Contents

Chapter 1: About Subsetting

Host Encyclopedia Subsetting

The following list describes host encyclopedia subsetting:

- Definition of part of a model that a user wants to change on the toolset.
- Checkout of the objects that are requested in the definition, which the software accomplishes by:
  - Making a temporary copy of those objects and transferring them to the toolset.
  - Tracking and resolving conflicts between users trying to change the same object.

A subset is a temporary workset that is created from a model.

Subsetting provides a means of asking for the parts of a model to accomplish a task. Subsetting has been designed to maximize the ability to share objects without conflicting updates.

Benefits

Subsetting benefits users by:

- Speeding development and maintenance
- Minimizing the amount of data that is checked out to perform toolset tasks
- Reducing transfer time by reducing volume of data transfers
- Allowing concurrent development from a single model

Concepts

Three key concepts underlie subsetting:

- Subset definition
- Expansion
- Protection

Subset Definition

Subset definition is the process of selecting objects from a model to include in a subset.

Object

As used here, an object is simply part of a model. The various objects are represented in figures in the same way they are represented in CA Gen diagrams.

To define a subset, select the objects (scoping objects) related to the task you want to accomplish.

Scoping Object

A scoping object is an object that you can select to include in a subset. To be selectable, objects must already exist in the model, and they must be of certain types. Not all types of CA Gen objects are selectable.

Some of the most common types of scoping objects are entity types, procedures, screens, action blocks, databases, or load modules. For a complete list of scoping object types, see the Scoping Object Types section.

Selecting a scoping object causes a set of more objects to be included in the subset during the checkout. The additional objects are related in some way to the scoping object.

For example, if you select an entity type, on checkout the subset includes (among other objects) the entity types:

- Description
- Identifier
- Attributes

If you select a procedure step, on checkout the subset includes (among other objects):

- The related procedure
- Entity types that are referenced by the procedure step's import and export views
- Business system defaults that the procedure step uses

In a few cases, a scoping object consists of a single item, the scoping object itself. An example is an external object.

Factors that Affect Subset Content

The precise set of objects that a scoping object delivers to the toolset depends on:

- Scoping an object
- Protection granted the scoping object
- Expansion that is specified for the scoping object
- Subset type
- Maturity of the model

For example, if an entity type has no attributes, the subset contains no attributes.

Scoping the root subject area before transformation includes all entity types. Scoping the same root subject area after transformation includes the database also.

When creating a subset, you first select scoping objects and then specify their protection and expansion.

Scoping Object Types

For a list of valid types of scoping objects, see the following table:

Object Type	Description
ACTIVITY CLUSTER	Natural business system
BATCH JOB	Batch job
BATCH JOB STEP	Batch job step
BUSINESS AREA	Business area
BUSINESS SYSTEM	Business system
COMMON ACTION BLOCK	Common, default and derivation ads
COMPONENT IMPLEMENTATION	Component implementation
COMPONENT MODEL	Component model
COMPONENT SPECIFICATION	Component specification
CONFIGURATION INSTANCE	Configuration Instance
CRITICAL SUCCESS	Critical success factor
CURRENT DATA	Current database or data store
CURRENT INFO. SYSTEM	Current information system

Object Type	Description
CUSTOM PROXIES	Custom Proxies
DATA CLUSTER	Natural data store
DATA TABLE	Data table definition
DATABASE	Database definition
DIALECT	Dialect
ENTITY	Entity type
ENVIRONMENT	Environment
EXIT STATE	Exit state
EXTERNAL OBJECT	External object
FACILITY	Computing or communication facility
FUNCTION	Function definition
GOAL	Goal
INFORMATION NEED	Information need
INTERFACE TYPE	Interface type
INTERFACE TYPE MODEL	Interface type model
LOCATION	Location of business assets
MATRIX	ISP matrix
NAVIGATION DIAGRAM	Navigation diagram
OBJECTIVE	Objective
ONLINE LOAD MODULE	Online load module
OPERATIONS LIBRARY	Operations library
ORGANIZATIONAL UNIT	Root organizational unit
PERFORMANCE MEASURE	Performance measure
PROCEDURE	Procedure
PROCEDURE STEP	Procedure step
PROCESS	Process definition
SCREEN	Screen
SCROLL AMOUNT VALUE	Scroll amount value
SERVER MANAGER	Server manager
SPECIFICATION TYPE	Specification type

Object Type	Description
STORAGE GROUP	Storage group
STRATEGY	Strategy
SUBJECT AREA	Subject area
SYSTEM CLASS	System defined object class
TACTIC	Tactic
TECH DESIGN DEFAULT	Technical design default
TEMPLATE	System screen template
TRANS OPERATION	Transaction operation
TYPEMAP	Typemap
USER CLASS	User defined object class
USER OBJECT	User defined objects
WEB SERVICE DEFINITION	Web service definition
WINDOW LOAD MODULE	Window load module
WORK/SYSTEM ATTR SET	Work and system attribute set
z/OS LIBRARY	z/OS library

Note: System Class and ISP Objects are used to create matrices.

Expansion

Expansion is the extent to which more objects for a scoping object are included in the subset. During a subset checkout, the software expands each scoping object into a set of required and optional component objects.

Expansion Level

The expansion level of a scoping object controls the extent to which other objects are included.

At the end of subset definition, you are presented with a list of selected objects, each with default expansion level and protection level that is assigned to it. You can accept the default expansion level for all types of scoping objects, or you can specify different expansion level options.

Expansion Level Options

Valid expansion levels are:

- Default
- Short
- Full
- B
- A

Every scoping object has a default expansion level, but not all have the other expansion levels. For example, a procedure step has all five levels, but an exit state has only default expansion.

Default expansion contains more objects than short expansion and fewer objects than full expansion.

Expansions A and B, which occur in only a few types of scoping objects, usually contain more objects than short expansion and always fewer than full.

As you define a subset, you specify an expansion level; select the smallest expansion that is effective for your task.

Expansion Level Options by Object Type

For a list of valid expansion options for the various types of scoping objects, see the following table.

Scoping Object Type	**Short**	**B**	**A**	**Default**	**Full**
Batch Job	X		X	X	X
Batch Job Step	X	X	X	X	X
Business System	X			X	X
Common Action Blocks (Common, Default and Derivation)				X	X
Component Implementation	X			X	X
Component Model				X	X
Component Specification	X			X	X
Configuration Instance				X	X

Scoping Object Type	Short	B	A	Default	Full
Custom Proxies	X			X	X
Data Table	X			X	
Database	X			X	
Dialect				X	
Entity	X			X	X
Exit State				X	
External Object				X	
Function	X			X	X
Interface Type	X			X	X
Interface Type Model				X	X
ISP Objects				X	
Matrix				X	
Navigation Diagram				X	X
Online Load Module	X	X	X	X	X
Operations Library	X			X	X
Organizational Unit				X	
Procedure	X		X	X	X
Procedure Step	X	X	X	X	X
Process	X			X	X
Screen				X	
Scroll Amount Value				X	
Server Manager	X	X	X	X	X
Specification Type	X			X	X
Storage Group				X	
Subject Area	X			X	
System Class	X			X	
Tech Design Default				X	
Template	X			X	
Trans Operation	X			X	
Typemap				X	

Scoping Object Type	Short	B	A	Default	Full
User Object				X	
User Class				X	
Web Service Definition	X			X	X
Window Load Module	X	X	X	X	X
Work/System Attr Set	X			X	
z/OS Library	X			X	X

Object Expansion

Expansion of Scoping Objects

Expansion of scoping objects within a subset cascades from one level of scoping object to another:

- When the chosen scoping object expands, it often includes other, lower-level scoping objects.

  For example, Procedures expand to include Procedure Steps, which are also scoping objects in their own right.

- Next, the lower-level scoping objects included in the original scoping object expand.

  For example, Procedure Steps can expand to include Action Blocks, another scoping object.

- If the lower-level of scoping objects include any lower-level scoping objects, they expand next.

  For example, the Action Block of a Procedure Step can expand to include other (Used) action blocks.

Such cascading expansions can create a large subset, something we recommend that you avoid by selecting the lowest possible level of scoping object. For example, do not select a Procedure scoping object if choosing an Action Block can accomplish your objective.

Identifying Embedded Scoping Objects

To determine whether a scoping object contains lower-level scoping objects that can accomplish your objective equally well:

- Familiarize yourself with the list of valid scoping objects.

- Preview the expansion of scoping objects when creating or modifying a subset. During both creation and modification procedures, you can see all the lower-level objects that the scoping object includes (including any other scoping objects).

Note: Scoping objects that are contained by other scoping objects receive default expansion unless otherwise noted. If an expanded object is requested twice, it receive the highest protection and highest expansion requested.

Default Set of Expansion Objects

Every subset includes a default set of required objects. You can think of these objects as model defaults instead of business system defaults.

Some default scoping objects are not expanded when brought down as part of the default set. They are identified and added directly to the subset if no other scoped object has already caused them to be included.

Object Category	Object	Expansion
Root objects	Root subject area for the model	Not expanded
	Root function for the model and its data view set	Not expanded
	Root organizational unit, if one exists	Not expanded
System-supplied functions and dialects	System-supplied functions for time, date, and so on.	Default
	Dialects	Default
Global exit states	Global exit states	Default
	Messages and bilingual text for global exit states	Default
Work attribute set and system attributes	System supplied work attribute set and system attributes	Default
	Bilingual text for prompts and system attributes	Default
Scroll amount values	Scroll amount values in the model	Default
	Bilingual text for scroll amount values	Default
Technical design, database, and storage groups	Default database used by technical design object, if database exists	Not expanded

Object Category	Object	Expansion
	Default storage groups used by the technical design, if storage groups exist	Not expanded

Expansion Tables

The expansion tables provide the following information for each scoping object:

- Type of scoping object
- Valid subset types
- Valid expansion levels with
- Expansion objects
- Default protection for each expansion object shown

To determine the correct scoping object type, subset type, expansion, and protection when creating a subset; use the expansion tables.

Protection

Protection is the level of authorization you have on an object. Protection maintains the integrity of the model and prevents inadvertent or conflicting changes.

Protection Options

Each object in a subset has one of four protection options:

- Delete
- Modify (also known as an update)
- Access (also known as reference)
- Read

The protection options and what you can do with them follow:

Delete-Delete an object and references to it, if all references to the object are included in the subset.

Modify-Modify an object and its components, including properties. Add or delete associations to the object if the deletion does not imply the deletion of the object.

For example, modify on an entity type lets you add or change attributes or change the description of the entity type or its attributes but does not allow deletion of the attribute.

Access-Add or delete referencing associations to the object and its components.

For example, access on an entity type lets you add an information view of the entity type.

Read-Read an object and generate its components.

Protection Hierarchy

Protection levels are hierarchical and are from highest to lowest:

1. Delete
2. Modify
3. Access
4. Read

A user with Delete protection on an object also has Modify, Access, and Read privileges on the object.

The first user to request a Delete or Modify protection level on an object at checkout is granted it. The subsequent users requesting the checked out object with Modify or Delete receive the object with downgraded protection.

Effect of Protection on Subsequent Users

The protection that is granted on an object that has been checked out from the Host Encyclopedia determines the protection levels that are granted on the object on subsequent checkouts:

If a scoping object is checked out with	and a subsequent checkout requests	this checkout will be granted
Delete	Delete, Modify, Access, or Read	Read
Modify	Delete, Modify, or Access	Access
Modify	Read	Read
Access	Delete or Modify	Modify
Access	Access	Access

If a scoping object is checked out with	and a subsequent checkout requests	this checkout will be granted
Access	Read	Read
Read	Delete	Delete unless your subset is missing any references to the object
Read	Modify	Modify
Read	Access	Access
Read	Read	Read

Note: Only one person at a time can check out an object with Delete or Modify protection.

When someone has checked out an object with Delete protection, all other users requesting a copy of that object receive only Read protection.

When someone has checked out an object with Modify protection, all other users are restricted to Access or Read protection on that object.

Effect of Checkout on Protection

Unless a subset is checked out, its protection requests have no effect on other users. For example, if you request Delete protection on Procedure Step A in your subset definition but your subset is not checked out, other subsets containing Procedure A are not affected. Others can check out the procedure step to delete, modify, access, or read it.

Protection Granted

When you create a subset, the protections you specify for the scoping objects are only requests. The protection that is granted depends on the protection logic described next.

Protection for Scoping Objects

1. User requests protection on a scoping object when defining a subset.
2. User checks out the subset from the Host Encyclopedia to the toolset.
3. During a subset checkout, the software looks at the protection that is requested for the scoping object and checks the Host Encyclopedia to see whether anyone has already checked out the object.
4. If the scoping object is not checked out, the software usually grants the protection level requested. (See an exception in the following table.)

The following table shows subsetting protection that is granted to available scoping objects:

If you request an available scoping object with this protection	You get the scoping object with this protection
Delete	Delete Exception: True as long as your subset contains all references to the object you wish to delete. Otherwise, protection may be downgraded to Modify.
Modify	Modify
Access	Access
Read	Read

Protection Downgrades for Scoping Objects

If a scoping object in the subset you are checking out is included in a checked out subset, the protection level you request for that object is downgraded if there is a conflict between this protection level and that already granted to the object in the checked out subset.

The six combinations causing conflict follow:

If you request a scoping object with this protection	But the object is checked out with this protection	You get the object with this downgraded protection
Delete	Delete	Read
Delete	Modify	Access
Delete	Access	Modify
Modify	Delete	Read
Modify	Modify	Access
Access	Delete	Read

Protection for Expansion Objects

Expansion objects are the additional objects that the expansion of the scoping object includes in a subset during a checkout.

The protection that is granted to expansion objects determines what you and others can do to the object while you have it checked out. Usually, expansion objects are assigned protection that is based on three things:

- Relationship of the expansion object to the scoping object
- Protection level of the scoping object
- Expansion option of the scoping object

Protection for Default Set

For the default protection for the default set of expansion objects-the objects automatically included in every subset, see the table at the end of this section.

Note: Default protection noted for each component of the default set is granted unless the object is also brought into the subset by directly scoping the object or by the expansion of some other scoping object.

Default Protection Overridden

If a default set object is directly scoped or brought in as an expansion object, the default protection that is shown in the table is overridden. Instead, the default set object receives the protection that is granted to the scoping object or expansion object. For example:

- If you scope a dialect directly, the protection that is requested for the dialect is probably granted instead of the Access protection automatically granted dialects in the default set.
- If you scope an entity type that is directly contained by the root subject area, the root subject area comes into the subset as part of the entity type's expansion rather than as part of the default set.
- If you scope an action diagram with Read protection, the subset brings in entity types with Read protection. However, if the same entity type is scoped with Modify, the entity type receives Modify protection (assuming no conflicts).
- The protection granted the root subject area when it is part of the entity type's expansion depends on the protection granted the scoped entity type.

The following table shows default set of objects (with protections) included in every subset:

Object Category	Object	Protection
Root objects	Root subject area for the model	Access
	Root function for the model and its data view set	Access

Object Category	Object	Protection
	Root organizational unit, if one exists	Access
System-supplied functions and dialects	System-supplied functions for time, date, and so on.	Access
	Dialects	Access
Global exit states	Global exit states	Access
	Messages and bilingual text for global exit states	Read
Work attribute set and system attributes	System supplied work attribute set and system attributes	Access
	Bilingual text for prompts and system attributes	Read
Scroll amount values	Scroll amount values in the model	Access
	Bilingual text for scroll amount values	Read
Technical design, database, and storage groups	Default database used by technical design object, if database exists	Access
	Default storage groups used by the technical design, if storage groups exist	Access

Expansion Objects and Their Protection

Expansion objects are the additional objects that the expansion of a scoping object adds to a subset during a checkout.

The relationship between a scoping object and an expansion object determines the standard protection level that applies to the expansion object.

Most Common Types

The following types of correspondence are:

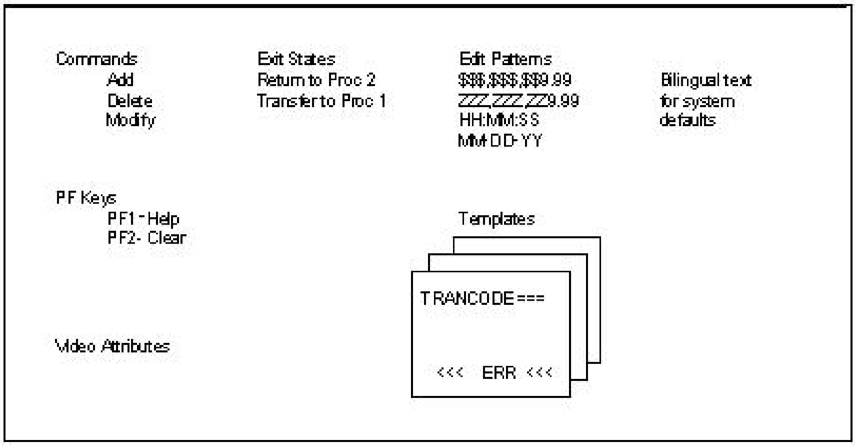
- **Non-shared component object**-A non-shared component expansion object is integral to the scoping object and the scoping object owns it. No other scoping object has an interest in it or can change it in any way. It belongs exclusively to the scoping object it is a component of, even if that scoping object is embedded in another scoping object.

  Examples of non-shared objects are action statements, bilingual text for system default and screen objects, and local or work views.

  If you request Modify protection on a scoping object that has wholly owned subsidiary objects, these non-shared components are upgraded to Delete protection. For example, an action block with Modify protection gets its action statements with Delete.

- **Shared component object**-The shared component expansion object is a part of the scoping object, a part of the basic definition of the scoping object.

  For example, system defaults are a part of a business system and, therefore, are shared components.



- **Shared companion object-**The shared companion object is not a part of the scoping object, but some part of the subset requires it.

  For example, the entity types that are used in information views are frequently shared companion objects-if the entity types are specified, or referenced, in earlier stages of view maintenance. These entity types are also named referenced entity types.

Shared companion objects to a scoped entity type include:

- All entity types that have relationships with the scoped entity type
- All entity types that have relationships that are used to identify:
  - The scoped entity type
  - Shared companion entity types

Correspondence Between Scoping and Expansion Objects Example

Expansion	Shared Components	Shared Companions	Non-Shared Components
Short	All system defaults	None	All bilingual text for system defaults and screen objects
Default	All system defaults All procedures and procedure steps	None	All dialog flows All bilingual text for system defaults and screen objects
Full	All system defaults All procedures and procedure steps All action diagrams All screens and/or windows	All referenced entity types and their parent subject areas	All dialog flows All bilingual text for system defaults and screen objects

Less Common Types

Two less common types of scoping objects are:

- Scoping objects without expansion objects.

  For example, scoping an External Object brings in only the scoped external object. Scoping a Dialect brings in only the dialect but not any text in that language.

- Scoping objects without shared companion objects

  For example, scoping a scroll amount value brings in only that scroll amount plus any bilingual text for the value, if any exists.

Protection Granted

The following table shows that the protection granted the expansion object that is based on:

- Its relationship to the scoping object

- The protection granted the scoping object

If the type of expansion object is	and the scoping object has this protection	the expansion object is granted this protection
Shared component	Delete	Delete
Shared component	Modify	Modify
Shared component	Access	Access
Shared component	Read	Read
Shared companion	Delete	Access
Shared companion	Modify	Access
Shared companion	Access	Access
Shared companion	Read	Read
Non-shared component	Delete	Delete
Non-shared component	Modify	Delete
Non-shared component	Access	Read
Non-shared component	Read	Read

Subset Types

Subset Type is an option that you specify when defining a subset. Subset Type refers to the stage of development the model is in when you create a subset. Its purpose is to give the expansion necessary to perform Construction tasks without requiring the user to specifically scope more objects.

By specifying the subset type, you can alter the expansion of the subset. The three subset type options are:

- Design
- Unit test
- System test

Design Phase Subset Type

Design is the default subset type. This option applies to all tasks performed outside the Construction Toolset. Design subset types provide the normal expansion of each scoping object.

Unit Test Subset Type

Select unit test when generating code for local testing of Construction Toolset tasks. In most development life cycles, unit testing comes before system testing or full integration testing.

Unit test subsets provide:

- All Data Structure Diagram (DSD) Components-Records, database, TD action block for any entities in the subset
- Limited the Packaging-Related components for any procedure steps that are found in the subset

Unit Test does not provide fully expanded load modules for procedure steps.

Types of Scoping Objects Used

When defining subsets for either unit test or system test, only the following scoping object types change their expansion:

- Entity type
- Subject area
- Procedure
- Procedure step
- Batch job
- Batch job step
- Online load module
- Windowed program
- Server Manager

Direct and Indirect Scoping

The treatment of these objects changes any time one of them is found during expansion, not only when they are directly scoped.

For example, if a view of an action block references an entity type in a unit test expansion, the entity type is indirectly scoped and obeys all the rules for unit test of an entity type.

Load Module Packaging

Load module packaging consists of the following types:

- Online or batch (more commonly known as block-mode, terminal-based, or 3270)

- Window
- Cooperative (for the Client/Server applications)

Component Packaging

These packages the Operations Libraries (not load modules) for the Component Model.

Strategy for Online Packaging

Online packaging (terminal-based, block mode, or batch) is created on the Host Encyclopedia and can be checked out and modified on the toolset for unit testing. All changes to packaging on the toolset are uploaded to the Host Encyclopedia.

Subsetting Work Flow in a Team Environment

The Subsetting work flow illustrates:

- The creation of a model containing several entity types
- Definition of subsets to modify and delete entity types at the toolset
- Successful interaction between subsets
- A protection downgrade

Sample Scenario

The following scenario involving two people illustrates the tasks in the work flow, each creating a subset definition and checking out their respective subsets simultaneously.

1. John creates the model ORDER_PROCESSING on his toolset. The model contains three entity types:
   - ORDER
   - ORDER_LINE
   - MISTAKE
2. John checks in ORDER_PROCESSING from his toolset to the encyclopedia.
3. John defines a subset on the encyclopedia to let him modify the entity type ORDER and to delete the entity type MISTAKE.
4. John checks out his subset (named FIRST) from his toolset.
5. Mary defines a subset on the encyclopedia to let her modify the entity type ORDER_LINE.
6. Mary checks out her subset (named SECOND) from her toolset.

7. John and Mary work on their respective subsets on their toolsets.
8. On his toolset, John checks in his subset.
9. On her toolset, Mary checks in her subset.
10. John checks out subset FIRST from his toolset, which copies a portion of the ORDER_PROCESSING model.
11. Mary changes her subset definition to try to check out the entity type ORDER with Delete usage.
12. Mary checks out her subset but finds that she has only Access to the entity type ORDER because John already had ORDER with Modify.

System Test Subset Type

Select system test when producing code to be installed with code from other subsets of the same model. Select system test to use executables that are produced with executables from other system test subsets of the model or with the entire model.

System test subsets are supersets of the same scoping objects in a unit test subset. Referential integrity triggers generated from a system test subset are also valid for use with other code produced from system test subsets of that model or the entire model. Selecting system test produces a much larger subset than selecting unit test. Use system test after successful unit testing.

Note: In some environments, you can select Unit Test when mixing modules from different subsets. For example, you are able to pull executables from other subsets of the same model from the LAN and then pick up modules from the local environment. In all cases, select System Test when you generate modules with dialog flows.

Both the unit test and system test options change the expansion of the subset for a number of scoping objects. They allow different expansion of key items (entity types, records, procedure steps) depending on the part of Construction you are in.

Chapter 2: Creating a Subset

You create subsets using Task Tables. These tables are easier to use because they are task oriented and reduce the need to use the tables in Expansion Tables to create a subset.

Before creating a subset, spend a few minutes learning about subset tasks and the subset task tables.

Plan a Subset

Before creating a subset, spend a few minutes learning about subset tasks and the subset task tables.

Subset Planning Using Task Tables

A subset task is any action that you can accomplish using the objects within a subset. A subset usually includes enough objects to support the execution of multiple tasks.

The task tables contain recommended definitions for over 200 subset tasks.

Use the Task Tables

The task tables do not itemize all objects that are included in a subset, nor do they give special information about the expansion of objects, their limitations, or other special data. In most cases, however, they represent a shortcut to the information you create your subset.

Create a Subset for General Use

1. Determine whether one of the General Subset Definitions suits your need. If this is the first subset being created for the current phase of development, consider the following subsetting task definitions are typical for:
   - Analysis
   - Design (first subset)
   - Procedure step maintenance
   - Workstation construction
2. If these subsets lack the detail that your task requires, see the procedure that follows in General Subset Definitions section.

3. Use the subset definition appropriate for your development phase as a guide, and create a subset. See the steps that are outlined in the Add Subset Definition section.

Create a Subset for a Specific Task

1. Consider the task that you want to accomplish.
2. Locate the corresponding Detailed Subset Task Definition.

   The subset type organizes the detailed subset task definitions in the task tables:

   - Design
     The tasks that are defined for the Design type subsets are listed under these headings:
     - Planning
     - Analysis
     - Design
       (which includes data modeling tasks before and after transformation)
     - Internal Design
       (which includes subset definitions for initial Transformation and Intelligent Retransformation)
   - Unit Test
   - System Test

   Note: The tasks that are defined for Unit Test and System Test subsets are listed under Construction. If the task tables do **not** contain the task that you need, see the procedure for Using the Expansion Tables.

3. Jot down the recommended scoping object and its recommended expansion and protection.

   P designates recommended protection, and the available options are:

   - D (for Delete)
   - M (for Modify)
   - A (for Access)
   - R (for Read)

E designates recommended expansion, and the available options are:

- S (for Short)
- D (for Default)
- F (for Full)
- B (valid only for Batch Job Step, Procedure Step, Online Load Module, Server Manager, and Window Load Module)
- A (valid only for the object types that are listed for B previously, plus Batch Job and Procedure)

See the Add Subset Definition procedure for step-by-step instructions.

Example of Task Table Usage

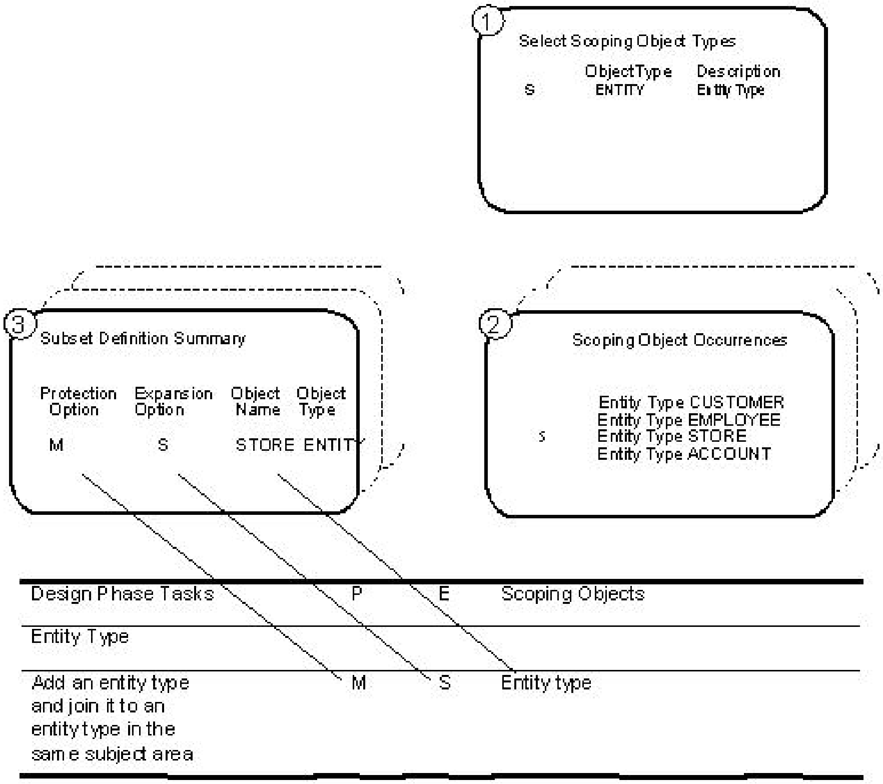
1. List the tasks you want to accomplish.

   An example of a detailed task is: Add the entity type PRODUCT to the subject area RETAIL OUTLET and join it to the entity type STORE.

2. Locate the corresponding task definition.

Design Phase Tasks	P	E	Scoping Objects
Entity Type-Add an entity type and join it to an entity type in the same subject area	M	S	Entity Type

3. Use the task definition as a guide in defining the subset you need. The following figure shows the correspondence between the task table data and how you specify it on the panels for defining a subset.



Design Phase Tasks	P	E	Scoping Objects
Entity Type			
Add an entity type and join it to an entity type in the same subject area	M	S	Entity type

Subset Planning for Task Not in Task Tables

Use the expansion tables when you cannot find an appropriate task entry in the task tables.

The expansion tables provide the following information for each scoping object:

- Type of scoping object
- Valid subset types
- Valid expansion levels with:
  - Expansion objects (objects that are brought in by expanding the scoping object)
  - Default protection for each expansion object shown

- A reference to the explanation of default expansion

To determine the correct scoping object type, subset type, expansion, and protection when creating a subset; use the expansion tables.

Using the Expansion Tables

1. Consider the task that you want to accomplish with the subset.
   - If you want to delete views, see the Plan a Subset to Delete Views section.
   - If your task is to delete an object, consider recommendations in Plan Alternative to Subsetting for Object Deletion. If an alternative is not feasible and your task involves deleting shared objects, see the Plan a Subset to Delete Shared Objects section.
   - For all other tasks, identify the type of objects that are most closely related to the task.
2. Determine whether the objects required for the tasks are scoping objects only, scoping objects that are also expansion objects, or expansion objects only. See the “About Subsetting” chapter in this guide. If your task requires modifying only high-level scoping objects, you do not need the expansion tables. Proceed with creating the subset and scope the objects with the default protection and expansion.
3. For the objects that are expansion objects, note the types of scoping objects that include it. For the considerations on selecting the best one for your task, see the Choose the Right Scoping Object section.

   For example, if you are working on screens and the Screen scoping object type includes the desired object, check the expansion table for Screen first.
4. Refer to the correct expansion table to identify the expansion level of the scoping object that includes the expansion object. If the object you need is included in multiple expansions, note the protection of the expansion object at each expansion and make note of the lowest one adequate for your task.

   For the considerations, see the Select the Lowest Expansion Level that Contains Needed Objects section.
5. Evaluate the default protection that is associated with the objects you need.

   For the considerations, see the Determine Whether to Accept or Change Default Protection section.
6. With your list of the required scoping objects, and the expansion and protection you plan to request, see the Add Subset Definition procedure for step-by-step instructions.

Choose the Right Scoping Object

Many scoping objects expand to include the same CA Gen objects. We recommend that you **not** automatically select the first type of scoping object that includes the object and protection you want.

Try to identify the scoping object and expansion that are best for the task. For example, if you want to include all system defaults in your subset, including all exit states, scope by business system (the most inclusive choice) at short expansion. However, if you want to work only on an exit state message, scope by exit state, which is the smallest unit whose expansion is suitable for the task and the one with the least potential of interfering with other users.

Using the expansion tables requires some practice, but if you use the suggested procedure, you quickly learn to select the proper scoping objects for most tasks.

Plan a Subset to Delete Views

Deleting the views requires Delete protection on the process, procedure step, or action block. However, even if these objects are marked INCOMPLETE in the Expansion Conflict Report, you still be able to delete the unwanted views.

- Scoping a process, procedure step, or action block with Delete protection allows you to delete:
  - Local, entity action, and import/export views that are not passed on flows
  - Attributes from any import or export views
  - You can get Delete protection on the views without being able to delete the procedure step.
- To delete an export view, you must scope all flowed-to procedure steps where the view is matched on the flow.
- To delete an import view, you must scope all flowed-from procedure steps where the view is matched on the flow.

Plan a Subset to Delete Shared Objects

The different parts of the model shares all scoping objects. Some non-scoping objects, like information views, can also be shared. Sharing objects makes it easier to develop and maintain applications, but it also complicates subsetting:

To delete a shared object in a subset:

- You must be granted Delete protection for the proper scoping object.
- The subset must include all references to the object.

Note: A preferred alternative to deleting objects in a subset is to use the Model Management function, delete, or rename objects within model.

To delete these shared objects		Obtain delete protection on these scoping objects	Note
Attribute	(See Entity Type, Entity Subtype, and Attribute)		1
Action Block	Common	the action block itself	
	Process Action Block	owning process	
	Procedure Step Action Block	owning procedure step	
Business System and Business System Defaults	Business System	business system (fully expanded)	
	Business System Defaults	business system	
	Bilingual Texts	business system	
	Edit Pattern	business system	
	Error Field Properties	business system	
	Literal Field Properties	business system	
	Normal Field Properties	business system	
	Prompt Field Properties	business system	
	System-Wide PF Keys	business system	
Command and Command Synonym	Command/Command Synonym	business system	
Database, Data table, and Link Table	Database	database	
	Data Table	data records and associated entities	
	Link Table	data records and associated entity type	
Entity Type, Entity Subtype, and Attribute	Entity Type	entity type	
	Entity Subtype	entity type	
	Attribute	entity type	
Entity View	(See View)		

To delete these shared objects		Obtain delete protection on these scoping objects	Note
	Exit State	all its USEing action blocks	
External Object	External Object	function or process where used	
Function and Process	Function	function	
	Process	process	
Group View	(See View)		
Link Table	(See Database)		
Procedure Step	Procedure Step	procedure or procedure step	
Process	(See Function and Process)		
Prompt and Prompt Usage	Prompt	procedure steps, entity type	2
	Prompt Usage	procedure step	
Relationship Membership	Relationship Membership	both entity types	
Screen and Template	(See Window)		
	Screen	screen	
	Template	template	3
View	... in Analysis	function, process, or action block	
	... in Design	procedure, procedure step or action block	
Window	Window	procedure step that contains it	

Note:Before an attribute can be deleted from an entity type, it must be removed from any information views. To identify the information views involved, generate the Host Encyclopedia Delete Prevention Report. To delete a prompt, you Modify protection for all screens that use it and Delete protection for the owning entity type. Before you can delete a template, all screens that use it must be in the subset.

Plan Alternative to Subsetting for Object Deletion

The shared objects are easier to delete on the host, for example, when the required scoping object creates an large subset or many objects that are required to be in the subset are missing at checkout.

When the deletion in a subset is undesirable, use Delete or Rename Objects, an option on the Host Encyclopedia Model Management menu. This function lets you delete objects at the host level, without defining and checking out a subset.

The same rules that govern the deletion on the workstation govern deletion on the Host Encyclopedia. For example, you cannot delete an object that is checked out with any protection other than Read.

Select the Lowest Expansion Level That Contains Needed Objects

The figures in this section show the three expansions of a business system:

- Short
- Default
- Full

The figures also illustrate:

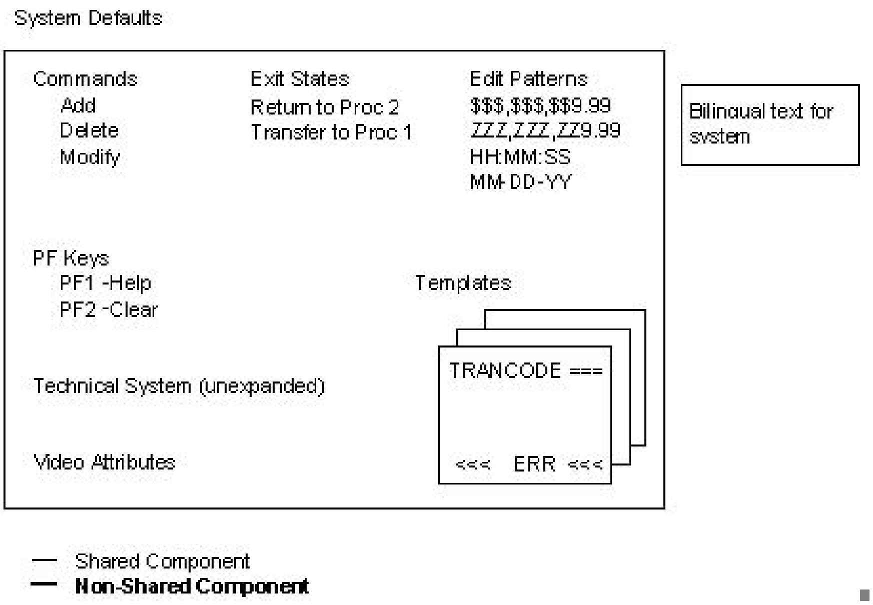
- The different expansions of the same scoping object contain different combinations of expansion objects.
- Some scoping objects contain other scoping objects.
- The default and full expansions of a business system contain other scoping objects, such as procedures and procedure steps.

The following examples illustrate how to compare the short, default, and full expansions of the Business System scoping object.

Example: Short Expansion

Short expansion of Business System includes only the system defaults and associated bilingual text.

Note: Because business system defaults affect many other parts of the model, they are difficult to change once they are defined. Therefore, define your defaults carefully before beginning the Dialog Flow Diagram (DLG), screen design or window design, and Procedure Action Diagrams.

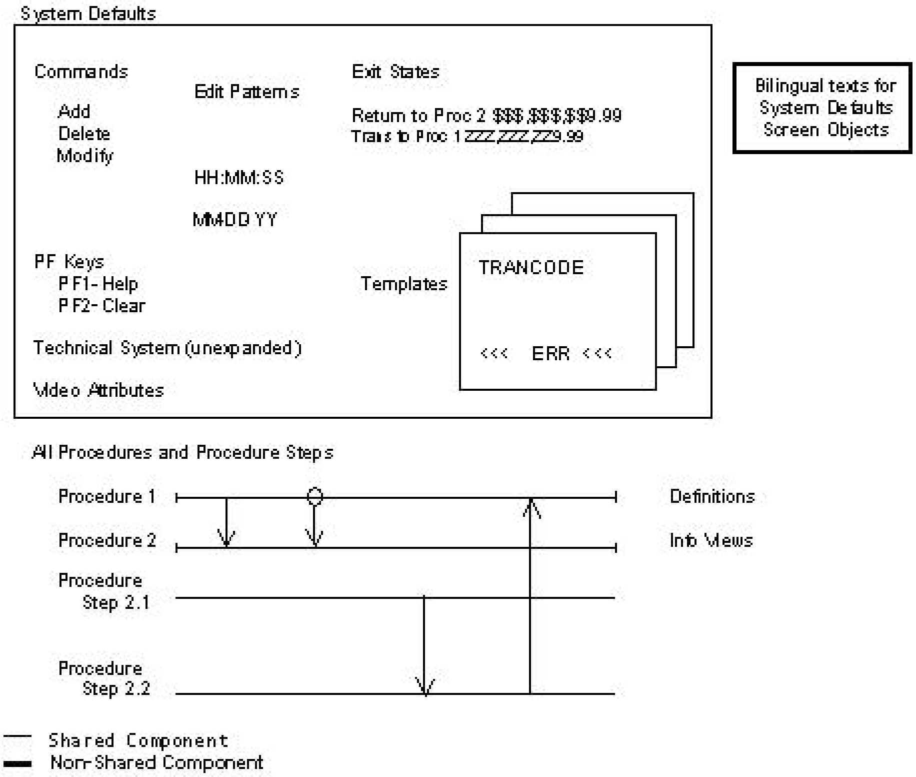


The following table shows expansion objects in short expansion of business system:

Shared Components	Shared Companions	Non-Shared Components
All system defaults (commands, PF keys, video attributes, exit states, edit patterns, and templates)	None	All bilingual text for system defaults

Example: Default Expansion

Default expansion of Business System is powerful. It includes all the objects in the short expansion and adds large groups of shared objects, that is, all procedures, all procedure steps, all dialog flows, and views. If you select this expansion, check out the subset, and you are granted Modify protection, no one else can modify any of these items.



The following table shows expansion objects in default expansion of business system:

Shared Components	Shared Companions	Non-Shared Components
All system defaults	None	All dialog flows
All procedures and procedure steps (unexpanded)		All bilingual text for system defaults and screen objects

Example: Full Expansion

Full expansion of Business System includes all the objects of short expansion and default expansion and adds still more objects. If you specify Business System and full expansion with Delete protection, then check out the subset, the full expansion prevents others from using any of the objects that are listed in the table except to read them.

The full expansion of a business system contains at least four scoping object types as shared components and at least two scoping object types as shared companions.

Note: Do not use full expansion of Business System unless you are the only analyst working on the entire business system or you delete a widely used system default.

The following table shows expansion objects in full expansion of business system:

Shared Components	Shared Companions	Non-Shared Components
All system defaults All procedures and procedure steps All action diagrams All screens and/or windows	All referenced entity types and their parent subject areas	All dialog flows All bilingual text for system defaults and screen objects

Determine Whether to Accept or Change Default Protection

In reading an expansion table, you can identify possible tasks for the expansion objects by examining their default protection.

The following table shows the decision table for changing scoping object default protection:

If the expansion object shows this default protection ...	Then keep the scoping object's default protection (Modify) to ...	Or change the scoping object's protection to ...
Modify	Modify the expansion object	Delete the expansion object Access the expansion object Read the expansion object
Access	Reference the expansion object (build an association to it)	Read the expansion object
Read	Read the expansion object	N/A
Delete (relatively few items)	Delete the expansion object	N/A

The following examples suggest how to approach interpreting the default protection column (the last column) of the expansion tables.

Example: Default Protection of Expansion Object Is Modify

If the default protection for an expansion object is Modify, you can use the associated scoping object to define a subset to modify, delete, access, or read the expansion object. For example, you can scope on Business System to modify, delete, access, and read system defaults. See the note for important qualifications on deleting an object marked Modify in the expansion tables.

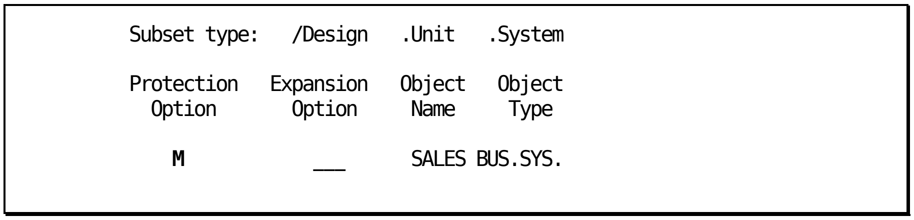
The following table shows when the default protection of expansion object is Modify:

This Business System expansion	Includes the following additional objects	with this protection
Short	System Defaults and Technical System	
	System defaults (PF keys, commands, exit states, templates, video attributes, and edit patterns).	Modify
	Unexpanded technical system.	Modify

Note: In deleting any object in a subset, include all references to that object in the subset. Because the short expansion of Business System includes only the system defaults and not the many objects that reference them, short expansion is not suitable for a deletion. For the deletion of system defaults, you must have full expansion of the Business System to bring in all usages.

To Modify Expansion Objects Marked Modify

1. Leave the default protection option of the associated scoping object at M (Modify) on the Subset Definition Summary:



```
Subset type:   /Design   .Unit   .System

Protection    Expansion   Object   Object
  Option        Option     Name     Type

    M            ___       SALES BUS.SYS.
```

2. Specify the smallest appropriate expansion option (here, S for Short).

```
           Subset type:   /Design   .Unit   .System

           Protection   Expansion   Object   Object
             Option       Option     Name     Type

               M            S          SALES BUS.SYS.
```

The short expansion is appropriate because the expansion table for Business System indicates that you can modify system defaults at short expansion.

Note: Unless you change the expansion option from blank (which represents the default expansion), your subset definition specifies default expansion of the scoping object. Default expansion create a larger subset than required.

To Delete Objects Marked Modify

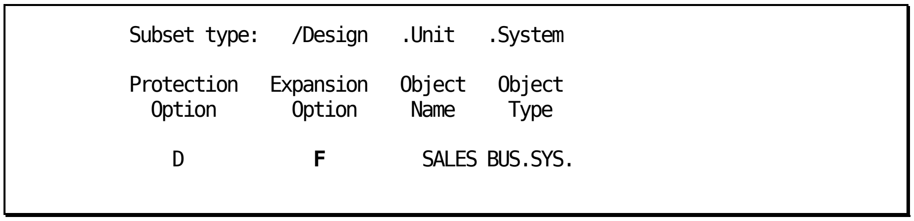
Before defining a subset to delete an object, *always* check the task tables. By skimming the task table for Business System Defaults, you can see that deletion of the most commonly used system defaults requires full expansion of the business system to bring in all usages.

To delete objects marked Modify in an expansion table, specify Delete as the protection option during subset definition.

1. Change the default protection option of the associated scoping object from M (Modify) to D (Delete) on the Subset Definition Summary:

```
           Subset type:   /Design   .Unit   .System

           Protection   Expansion   Object   Object
             Option       Option     Name     Type

               D            ___       SALES BUS.SYS.
```

2. Specify the appropriate expansion option. For example, to delete system defaults, change the expansion option from blank (default) to F (Full).



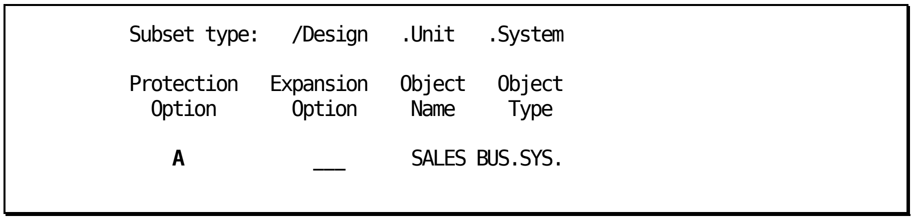
```
Subset type:   /Design   .Unit   .System

Protection   Expansion   Object   Object
  Option      Option     Name     Type

    D           F        SALES BUS.SYS.
```

Note: If you have Delete protection on an object marked Modify in an expansion table, you also have Modify protection on the object. Other users can obtain only Read protection on the object and cannot modify the object or even reference it so long as you have the subset that is checked out.

To Access or Read Objects Marked Modify

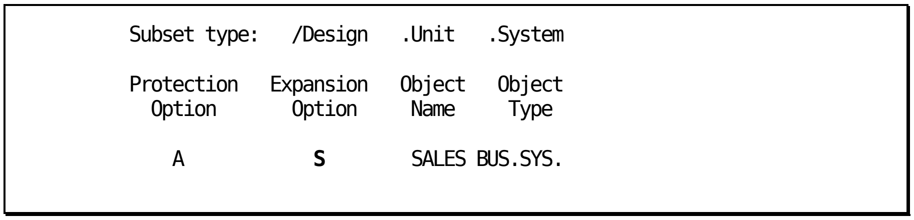
To reduce the protection of an object marked Modify, you specify Access or Read as the protection option during subset definition. However, see the following note:

1. Change the default protection option of the scoping object from M (Modify) to A (Access) or R (Read) on the Subset Definition Summary:



```
Subset type:   /Design   .Unit   .System

Protection   Expansion   Object   Object
  Option      Option     Name     Type

    A           ___     SALES BUS.SYS.
```

2. Specify the smallest appropriate expansion option for the task at hand. For example, to access system defaults, change the expansion option from blank (default) to S (Short).



```
Subset type:   /Design   .Unit   .System

Protection  Expansion  Object  Object
  Option      Option    Name    Type

    A           S        SALES BUS.SYS.
```

Note: Objects marked Modify in one expansion table are often marked Access in other tables. This indicates that they are included as part of the shared companions of another scoping object. By using one of these other scoping objects with Modify protection, you can add associations to the desired shared companion objects. As a rule, rather than reduce the protection of an object, look for scoping objects that already offer Access protection. They frequently include other expansion objects that you also need. For example, scoping procedures, procedure steps, screens, or templates give you their business system defaults with Access protection.

Example: Default Protection of Expansion Object Is Access

If the default protection for an expansion object is Access, the scoping object automatically includes the expansion object at Access protection if you are granted at least Modify protection on the scoping object. You can add new associations to the Access-level objects, but you cannot change anything about them.

For example, if you scope on Procedure Step at short expansion, you are given only the views of the action block of the scoped procedure step-without the action block statements. Because the body of the action block is not given for this expansion level, its protection is reduced to Access. You want to use this expansion for view maintenance.

The following table shows when the default protection of expansion object is Access:

This Procedure Step expansion	Includes the following additional objects for Design	with this protection
Short Only	Unexpanded Action Block, Windows and Dialog Boxes	
	Unexpanded action block of the procedure step (views only).	Access
	Unexpanded (empty) windows and dialog boxes.	Access

To Access Objects Marked Access

1. Leave the default protection option of the associated scoping object at M (Modify) on the Subset Definition Summary:

```
Subset type:   /Design   .Unit   .System

Protection   Expansion   Object   Object
  Option       Option     Name     Type

    M           ___       ADD PStep 1
```

2. Specify the smallest appropriate expansion option. In the sample, it is S (Short).

```
Subset type:   /Design   .Unit   .System

Protection   Expansion   Object   Object
  Option       Option     Name     Type

    M            S        SALES BUS.SYS.
```

The short expansion is appropriate, because short expansion of Procedure Step gives you access to action block views.

Example: Default Protection of Expansion Object Is Delete

If the default protection for an expansion object is Delete, the scoping object totally owns the object and can automatically include the expansion object at Delete protection, if you are granted at least Modify protection on the scoping object.

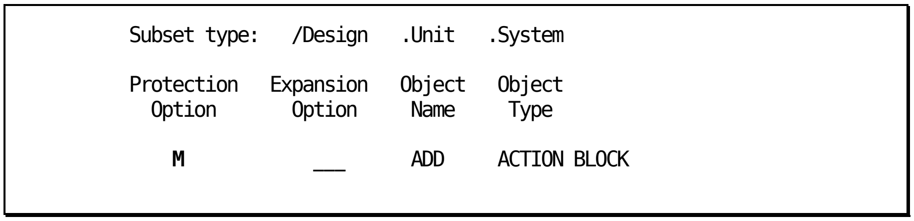
For example, if you scope an action block with Default or Full expansion, you can delete action statements of the action block.

The following table shows when the default protection of expansion object is Delete:

This Action Block expansion	**Includes the following additional objects**	**with this protection**
Default and Full	Action Statements	Delete
	All action statements in the action block.	

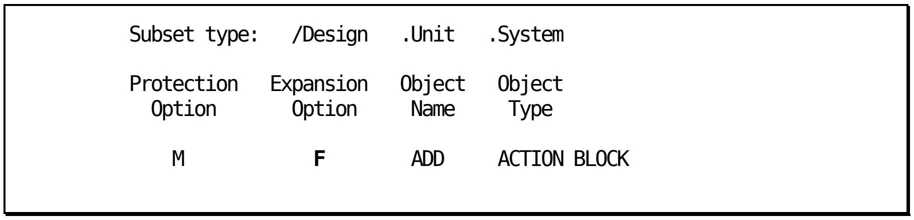
To Delete Objects Marked Delete

1. Leave the default protection option of the associated scoping object at M (Modify) on the Subset Definition Summary.



```
Subset type:   /Design   .Unit   .System

Protection   Expansion   Object   Object
  Option       Option     Name     Type

    M           ___        ADD     ACTION BLOCK
```

2. Specify the smallest appropriate expansion option. In the sample, we use blank (default) and F (Full).



```
Subset type:   /Design   .Unit   .System

Protection   Expansion   Object   Object
  Option       Option     Name     Type

    M           F          ADD     ACTION BLOCK
```

Full expansion is appropriate if you also want Delete protection on any action blocks USEd by this action block. Default expansion is appropriate if you want only to reference the views of USEd action blocks.

Example: Default Protection of Expansion Object Is Read

If the default protection for an expansion object is Read, the scoping object in question is bringing in some other scoping object as a shared companion item with Access protection and probably with short expansion.

For example, if you receive Modify or Delete protection on a screen or template, you get Read on the bilingual text for its system defaults. You get Read because the business system that owns the defaults is in the subset with Access protection.

If you specify the Unit Test or System Test subset type for a scoping object, it can include the Read object either directly or indirectly.

For instance, if you scope on Procedure Step with Unit Test, you can read the records, database, of the entity types that are referenced by the views of the procedure step's action block.

If a view of an action block references an entity type in a unit test expansion, the entity type obeys all the unit test rules for an entity type.

This Entity Type expansion	Includes the following additional objects for Unit Test	with this protection
Short and Default	Data table and Databases	
	Fully expanded data table for the transformed entity type.	Read
	The database and technical design for the data table.	Read
	TD Action Blocks, DBRMs and Implementation Logic	
	Unexpanded TD action block for the scoped entity type and table.	Read
	DBRM and implementation logic units for the TD action blocks.	Read

Create a Subset Definition

Subset Definition Guidelines

- Know what you plan to do with a subset before you define it. Create each subset to support a specific set of toolset tasks.
- Define the smallest subset possible.
- Smaller subsets result in:
  - Lower cost for checkout
  - Reduced protection conflicts between users of a model
  - Minimum impact of current access to the encyclopedia
- Minimize potential conflict over objects that may be shared across subsets by requesting the minimum protection and expansion that is required to accomplish your task.
- After creating a subset, review the expansion and protection of each scoping object from the Subset Definition Summary.
  - Remove unnecessary objects.
  - If you have scoped both the procedure and the procedure step, remove the procedure if possible, or if not, remove the individual procedure step that is already part of the procedure expansion.

  - Lower the protection levels where possible.
  - Never request Delete protection for a scoping object unless you delete a component that requires this level of protection. Any other subset that needs the object can get it with only Read protection.
  - Avoid requesting Modify protection when Access works too.
  - Lower the expansion levels, where possible.
  - For example, Scope only the needed procedure step instead of the whole procedure.
  - To maintain a procedure step action diagram, scope the procedure step.
  - To maintain the dialog flows between two or more procedure steps, scope only those steps with short expansion and subset type Design.
  - Avoid requesting full expansion when it is not required for the task.

- If an object you delete is selectable through the Delete Object function, use this means of deleting the object rather than scoping the object with delete protection in a subset.
- If the object you delete is not selectable through the Delete Object function, determine how to define your subset for this object deletion by running the Delete report from the Host Encyclopedia Reports menu named Object Cross Reference Reports. This report lists the objects that you must include in your subset with the object you want to delete.
- To delete an object and references to it, explicitly select for inclusion all references to the object in the subset. Otherwise, you receive a message that the subset is incomplete. Do not assume that asking for Delete protection of an object automatically brings in all usages of the object.
- If you create a reference to an object, be sure that you request Access protection, not only Read.

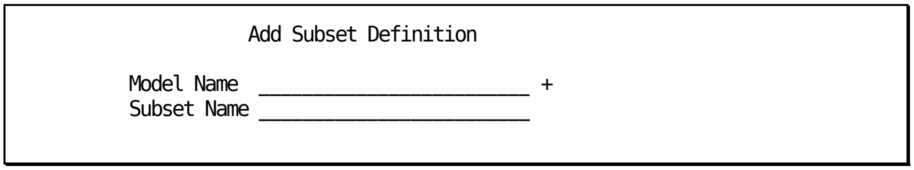
  For example, to create an action diagramming information view, you must have Access protection of an entity type. You cannot create views when the entity type has Read protection.

Subset Definition Procedure

Subset definitions are requests and do not ensure the object protection that you specify.

Add Subset Definition

1. From the Main Menu, select Host Encyclopedia Functions, Subset Management, Add Subset Definition or enter 1.4.1.
2. Enter the name of an existing model or press F4 to use the Model name selection list.

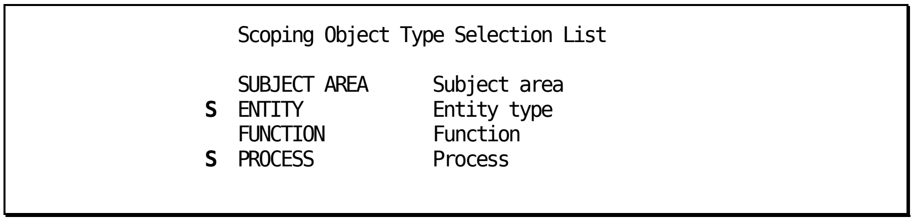


```
                   Add Subset Definition

         Model Name  _________________________  +
         Subset Name _________________________
```

Note: You can scroll or use F (Find) or L (Locate) commands on all selection lists. Enter any character except period (.) by the item you want and press Enter.

3. Enter the name of the subset you want to create and press Enter.
4. For each scoping object you want included in your subset, enter a forward slash (/) next to the desired object type and press Enter. An “S” appears for each object you select. Press F8 (Down) and F7 (Up) to scroll and review the complete list.

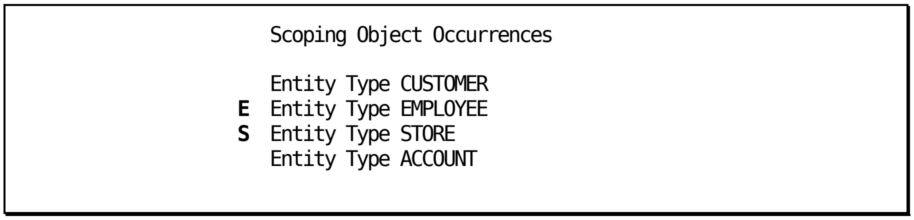
   Note: Always scope on the lowest-level object possible. For example, when possible scope a Procedure Step instead of a Procedure. The shared companions of a Procedure Step are limited, whereas the shared companions of a Procedure include all procedures flowed to and from, their action blocks, and so forth.



```
                 Scoping Object Type Selection List

                 SUBJECT AREA      Subject area
              S  ENTITY            Entity type
                 FUNCTION          Function
              S  PROCESS           Process
```

5. Enter a forward slash (/) next to the desired objects and press Enter twice.

   The list of scoping object occurrences can occupy multiple screens, depending on the type of object and the size of your model. You can repeat selecting until you have selected all scoping objects for your subset or until you exit from the function by pressing F3 (End).

```
   Scoping Object Occurrences

   Entity Type CUSTOMER
 E Entity Type EMPLOYEE
 S Entity Type STORE
   Entity Type ACCOUNT
```

   Note: Avoid duplicate scoping. For example, once you create a view of an entity type in an action block, do not scope the entity type in the subset that is used to modify the action block, unless it is to modify the entity type.
   Also, do **not** include attributes in views because you need them some day. It is easier to add views later than to delete them.

For a complete list of scoping object types, see the following table:

Scoping Object Type	Description
ACTIVITY CLUSTER	Natural business system
BATCH JOB	Batch job
BATCH JOB STEP	Batch job step
BUSINESS AREA	Business area
BUSINESS SYSTEM	Business system
COMMON ACTION BLOCK	Common, Default, and Derivation ADs
COMPONENT IMPLEMENTATION	Component implementation
COMPONENT MODEL	Component model
COMPONENT SPECIFICATION	Component specification
CONFIGURATION INSTANCE	Configuration Instance
CRITICAL SUCCESS	Critical success factor
CURRENT DATA	Current database or data store
CURRENT INFO. SYSTEM	Current information system
CUSTOM PROXIES	Custom Proxies
DATA CLUSTER	Natural data store
DATA TABLE	Data table definition

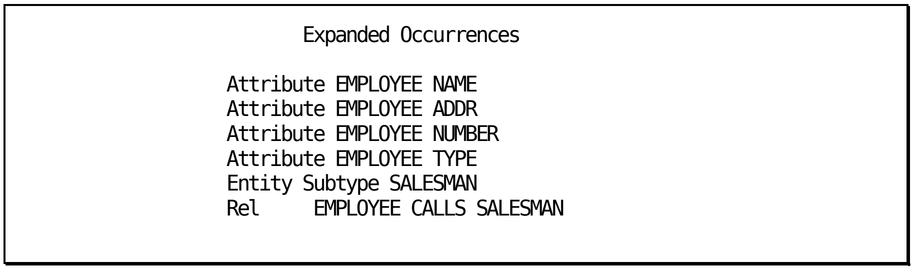
Scoping Object Type	Description
DATABASE	Database definition
DIALECT	Dialect
ENTITY	Entity type
ENVIRONMENT	Environment
EXIT STATE	Exit state
EXTERNAL OBJECT	External object
FACILITY	Computing or communication facility
FUNCTION	Function definition
GOAL	Goal
INFORMATION NEED	Information need
INTERFACE TYPE	Interface type
INTERFACE TYPE MODEL	Interface type model
LOCATION	Location of business assets
MATRIX	ISP matrix
NAVIGATION DIAGRAM	Navigation diagram
OBJECTIVE	Objective
ONLINE LOAD MODULE	Online load module
OPERATIONS LIBRARY	Operations library
ORGANIZATIONAL UNIT	Root organizational unit
PERFORMANCE MEASURE	Performance measure
PROCEDURE	Procedure
PROCEDURE STEP	Procedure step
PROCESS	Process definition
SCREEN	Screen
SCROLL AMOUNT VALUE	Scroll amount value
SERVER MANAGER	Server manager
SPECIFICATION TYPE	Specification type
STORAGE GROUP	Storage group
STRATEGY	Strategy
SUBJECT AREA	Subject area

Scoping Object Type	Description
SYSTEM CLASS	System defined object class
TACTIC	Tactic
TECH DESIGN DEFAULT	Technical design default
TEMPLATE	System screen template
TRANS OPERATION	Transaction operation
USER CLASS	User defined object class
USER OBJECT	User defined objects
WEB SERVICE DEFINITION	Web service definition
WINDOW LOAD MODULE	Window load module
WORK/SYSTEM ATTR SET	Work and system attribute set
z/OS LIBRARY	z/OS library

(Optional) Expanding Scoping Objects

1. To see any subordinate objects, enter E by the scoping object and press Enter.
2. To select only certain subordinates, enter S by the desired items and press Enter twice.

   Note: The partial selection applies only to subordinates that are scoping objects.



```
                    Expanded Occurrences

          Attribute EMPLOYEE NAME
          Attribute EMPLOYEE ADDR
          Attribute EMPLOYEE NUMBER
          Attribute EMPLOYEE TYPE
          Entity Subtype SALESMAN
          Rel     EMPLOYEE CALLS SALESMAN
```

Tailoring the Subset Definition to the Task

1. If testing, space out the slash (/) at Design phase and select Unit Test or System Test. Otherwise, accept the default.

```
                       Subset Definition Summary

                 Subset type:     /Design     .Unit     .System

                 Protection   Expansion   Object   Object
                  Option       Option      Name     Type

                   M            __          STORE ENTITY
                   M            __          ADD PROCESS
                   M            __          DEL PROCESS
```

2. For each displayed object that is not required by your task, enter D on the line in front of the object's protection option. For the recommendations, compare the displayed list of objects with the Task Table for the tasks you plan to perform with this subset.

   Note: Review subset summaries to remove redundant or duplicate scoping objects (embedded scoping objects). If you scoped an object that was included by the expansion of another object because you request more protection or expansion of that object, you can now remove the object that was included in the initial expansion.

3. For each object that requires a protection level different from that displayed, make the needed change. For the recommendations, compare the displayed protection for each object with the recommended protection in the relevant task table. Protection options are:

   - D (Required to Delete the object)
   - M (Required to Modify the object)
   - A (Required to Access, reference, or build an association to the object)
   - R (Required to Read the object)

4. For each object that requires an expansion level different from that displayed, make the needed change. For the recommendations, compare the displayed expansion for each object with the expansion recommended in the relevant task table.

To display a prompt stating the supported expansions for any given object, enter a forward slash (/) in the object's Expansion Option field. Options include Default (blank) and one or more of the following:

- S (Short, includes the minimum number of related objects, and are selected when sufficient for your task)
- A (defined for certain tasks that are associated with the selected object types as described in the task tables)
- B (defined for certain tasks that are associated with the selected object types as described in the task tables)
- F (Full, includes all related objects, that is, objects for all usage of the subset type and are selected only when required)

5. When you have reviewed the subset definition and have made all desired changes, press Enter twice to confirm and accept it.

Chapter 3: Checking a Subset In or Out of the Encyclopedia

Initiate Subset Checkout from the Host Encyclopedia

Checking out a subset transfers a copy of the objects in the subset definition to the local encyclopedia on your workstation. The objects in the subset definition are marked in the Host Encyclopedia as checked out to your TSO user id, not your workstation.

To check out and change a subset, you need access to the subset, but not necessarily to the model. If you do not have this authority, ask the model administrator or encyclopedia administrator for it.

You can check out subsets that are created on the Host Encyclopedia to a Toolset from the Toolset or from the Host Encyclopedia.

Subset Checkout Guideline

- Check out a subset only as you need it.
- To determine how the successful checkout is currently, generate the Expansion Conflict Report.

Subset Checkout Procedure

The following procedure describes where downloading is initiated from the Host Encyclopedia.

To download a subset to a toolset manually, follow these steps:

1. From the Main Menu, select Host Encyclopedia Functions, Subset Management, Download, or enter 1.4.13.
2. Identify the model containing the subset and the subset to download. To download all objects in the model, enter ALL as the subset name.
3. Enter the model release for Software version. If you do nott know, enter 1.3.5 from the Main Menu to display the Model Statistics panel, where the last entry is model release.
4. Accept the default transaction file name to use userid.IEF.TRAN as the name of the data set to be created at download or enter a different name, where your TSO user ID is automatically used as the prefix.

5. Accept the default Extract option, No, to download the subset to the workstation.
6. Enter the codepage.

   Note: For a listing of supported codepages, see the *CA Gen Client Server Encyclopedia User Guide*.

Note: If you do not know what it should be, bring up the Encyclopedia Client and detail an existing model using the same language. The codepage is displayed as Language Code. This is the identifier for the language (for example, English, French, Japanese) on the destination platform (for example, UNIX, Windows), or the platform where the encyclopedia resides.

Also, look at a model that already resides on the workstation. The model properties show the current version, tool, and codepage.

1. Accept online as the execution mode or change to batch. Press Enter.
   - If Online, initiates the download process. The message “IEF OK” indicates the successful completion.
   - If Batch, displays JCL. Submit job. The “Batch download was successful” message indicates a successful completion.
   - To specify JCL, use option 5.2.1 from the Main Menu.
2. Transfer (as binary or data) the resulting file, userid.ief.tran, from the Host Encyclopedia to the toolset model directory on your workstation; name the file to be transferred “CHECKOUT.TRN.”

   For example, the path to the file in drive d for subset1 would be:

   d:\subset1.ief\checkout.trn
3. Open the subset:

   a. Select Model, Encyclopedia, Check Out a Model. Uncheck the checked box for Perform file transfers for encyclopedia options or verify that it is not selected.

   b. Enter the Model Name, Subset Name, and Local Name as described in online help. Be sure to enter <subset> as the local name.

   For example, if the subset name were subset1, you would enter subset1 as the local name. The name that you enter identifies the <subset>.ief directory where the checkout.trn file resides.

Recover from Scoping Errors

If scoping errors are such that you modify your subset definition, and check it out again, consider overriding your current checkout instead of checking the subset back in with no changes.

Identify Protection Downgrades

By downgrading an object's protection, the checkout function prevents conflicts between subsets that share the object.

You can identify protection downgrades in these ways:

- View the Model Retrieval Status Report on the workstation after checkout.

  This report is generated during a checkout as part of checkout.trn. For the details, see the Online Help.

- Generate the Expansion Conflict Report on the host before or after checkout. Else, submit SQL to Check Protection Level. This SQL lists all named objects in an expanded subset and shows the protection level received.

SQL to Check Protection Level

```
SELECT OT_MNEMONIC, SE_CKO_STATUS, PROP_CHAR_VAL
FROM
DMDL, SOBJ, DSUBID, DSUBEX, DPRP
WHERE
MODEL_NAME = 'CBS SRVS SURVEY SYSTEM-DEVL'
AND S_MODEL_ID = MODEL_ID
AND S_SUBSET_NAME = 'SUBJECT AREAS'
AND SE_SUBSET_ID = S_SUBSET_ID
AND OT_OBJ_CODE = SE_OBJ_TYPE
AND PROP_OBJ_ID = SE_OBJ_ID
AND PROP_TYPE_CODE = 224
ORDER BY OT_MNEMONIC, PROP_CHAR_VAL
```

Sample Report from Protection Level Check

```
OT_MNEMONIC SE_CKO_STATUS PROP_CHAR_VAL
ACBLKTD U E1777310
ACBLKTD U E1777327
ACBLKTD U E1777330
ACBLKTD U SRVDGQU1
ACBLKTD U SRVDPRS1
ACBLKTD U SRVDQSL1
ACBLKTD U SRVDVRS1
ACBLKTD D SRVJGQS1
ACBLKTD D SRVJGQS2
ACBLKTD D SRVJINS1
ATTRUSR U EXTERNAL_NORM
ATTRUSR U FIELD_NAME
ATTRUSR A INPUT_STRING
ATTRUSR A INPUT_STRING
ATTRUSR A JULIAN_DATE
ATTRUSR U LABEL
ENTRYPNT D TOPIC0X1
ENTRYPNT D VALDEMX1
ENTRYPNT D VALPHRX1
ENTRYPNT D VALRSPX1
ENTVW A EXPORT
ENTVW A EXPORT
ENTVW A EXPORT
ENTVW A EXPORT
ENTVW A EXPORT
ENTVW A IMPORT
EXSTATE A ACTION_BAR_IS_INVALID
EXSTATE A CREATE_OK
EXSTATE A DELETE_OK
EXSTATE A DISPLAY_OK
EXSTATE A FLOW_TO_MENU
EXSTATE A INVALID_COMMAND
EXSTATE A PROCESSING_STARTED
```

Make an Incomplete Subset Complete

If the expansion of any object is marked incomplete, you have requested Delete on that object, but the expanded subset fails to include all references to the object.

To fix this problem, you can modify the subset definition or, in many cases, delete the incomplete object on the Host Encyclopedia.

- Override or check in the subset before modifying.
- Modify the subset definition.

- Generate the Host Encyclopedia Delete Report to identify the missing references. Then, add the missing objects to the subset, using Modify Subset Definition.
- When many objects are missing, select the next option.
- Delete the incomplete object on the Host Encyclopedia.
- Delete all references to the incomplete object, using subsets. Then use Delete or Rename Objects, an option on the Model Management menu. This function lets you delete most objects at the Host Encyclopedia level, without defining and checking out a single, sometimes massive subset.

Note: Some objects, like views, cannot be deleted on the Host Encyclopedia. Often, the users need Delete protection on an action diagram to delete a view.

The following table shows an example: subset is missing needed objects:

Task	**To correct how Procedure Step A updates a particular field**
What was scoped	Procedure Step A, default expansion and protection.
Resulting subset	The resulting subset contains Procedure Step A without the action block that updates the field. The subset can be used to correct the call to the action block but not to correct the logic in the action block.
What to do next	Check in subset or override checkout status. Determine why scoping for the second task is incorrect by checking the action block, tasks entry in the index. The index reference for action block tasks points to the table for the Procedure Action Diagram. That table suggests scoping an action block, default expansion, Modify protection, to change or delete statements in an action block. Use the Subset Definition Summary function to specify the action block, not the procedure step, at default expansion, Modify protection. Check out the subset again.

More information:

Change a Subset Definition (see page 71)

Obtain Downgraded Objects at Requested Protection

After identifying protection downgrades, you want to:

- Wait for the conflicting subsets to be checked in. Meanwhile, if you have checked out your subset:
  - Check it in so that you can try again.
  - Override the subset checkout status of your subset (if you checked it out). If you are a member of a group user ID and the group is the model administrator, you can also override the checkout or can ask the model administrator to override the checkout.
- Ask the user of a conflicting subset to make the downgraded objects available to you. To make the objects available, the user must check in the subset and:
  - Wait for you to finish working with the objects.
  - Change the conflicting subset's definition to lower the protection that is requested on the downgraded objects so that the new protection requests are not in conflict with your requests.
  - Asking the current user to check in or change a subset definition is justified if the person has requested Delete protection on an object when Modify or Access would serve too.
- If the current user is unavailable, consult the model administrator about changing the checkout user ID for that person's subset.
- After the change of checkout user ID, the person whose TSO ID is substituted can check the subset in, thus making the subset's objects available for other users. For more information, see the Change Checkout User ID for Subset section.

The following table shows an example about the subset's objects have insufficient protection:

Task	**To delete the Import View from Procedure Step C**
What was scoped	Procedure Step C, default expansion and Delete protection.
Resulting subset	The resulting subset contains Procedure Step C but without Delete protection. The import view of Procedure Step C cannot be deleted.

Task	To delete the Import View from Procedure Step C
What to do next	Check in subset or override checkout status. Determine why scoping for view deletion is incorrect. To delete an import view (entity view), you must scope the process or procedure step with full expansion and Delete protection and every process or procedure step that calls it, with expansion A and Modify protection. Run the Delete report to identify every process and procedure step that calls the view. To delete a view, your subset must contain all references to the view. The Objects Preventing Deletion report helps you define a subset that contains all references to the object. Use the Subset Definition Summary function to: Change the expansion of Procedure Step C to Full. Scope every process and procedure step that references the import view, with expansion A. Modify protection. Check out the subset again.

Return Objects Not Needed for Current Task

When you get many shared objects in your subset that you, do not need for your current task, it is helpful in a teaming environment to check in the subset right away, rescope it, and check it out again.

The following table shows that an example about the subset contains more objects than necessary:

Task	To correct the spelling of the company name on a Screen Template.
What was scoped	Screen Template, default expansion and default protection.
Resulting subset	The resulting subset contains all screens that use the template with Modify protection. Some team members are upset because they cannot work on the screens they need.

Task	To correct the spelling of the company name on a Screen Template.
What to do next	Upload the subset with check in as soon as possible to relinquish control of everyone's screens. Determine why the screens were included, by checking the expansion table for Template at default. Short expansion of the template is what you need. It lets you change fields on the template without monopolizing all the screens that use the template. Use the Subset Definition Summary function to specify short expansion for the scoped template. Check out the subset again. With short expansion, your subset receives the template and all its screen items without tying up the screens that use the template.

The following table shows expansion table for template high-level:

This Template expansion	Includes the following additional objects for all subset types	with this default protection
Short and Default	Screen Template and Bilingual Text	
Short and Default	Business System and System Defaults	
Default	Screens, Procedure Steps, Action Blocks, Windows, and Procedures	
Default	Referenced Entity Types and Parent Subject Areas	

Use the Checked Out Subset

- When you create a new procedure or action block by copying a skeleton, clean up any extra views before checking in the subset. It is easier to add views than to scope to delete import and export views once an action block is used or a procedure step is incorporated into the Dialog Flow Diagram.
- Avoid clicking OK on toolset panels when you have not changed anything. Clicking OK without changing anything creates false object changes and increases the size of updates to the Host Encyclopedia. If you have not changed anything, press CANCEL instead.

Change Checkout Status or User

If you determine that you do not want to work on the subset in the state that you received it, it is less costly to override the checkout than to check the subset back in.

To allow multiple users to work on a subset while it is checked out, you can change the checkout userid. This function is also useful if you want to transfer responsibility of a checked out subset from one developer to another.

Change Checkout User ID for a Subset

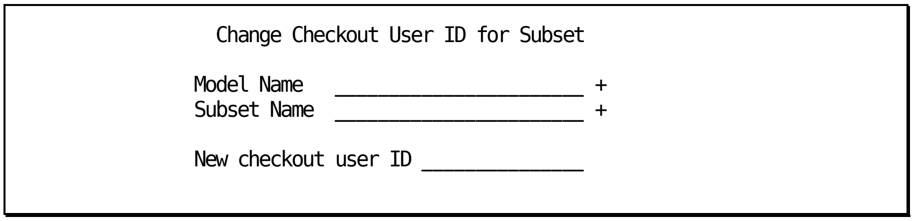
Change Checkout User ID for Subset changes the user ID to which a subset is checked out. It also lets users working with multiple TSO IDs check in a subset under a different user ID than was used for the checkout. The user whose ID you substitute must have access to the subset.

When you can change the checkout user ID to allow someone else to work on your subset, give them the .DAT files.

Encyclopedia administrators or model administrators use this function if a subset is checked out and they want someone else to check it back in.

Follow these steps:

1. From the Main Menu, select Host Encyclopedia Functions, Subset Management, Change Checkout User ID for Subset or enter 1.4.9.



```
                 Change Checkout User ID for Subset

               Model Name    ______________________ +
               Subset Name   ______________________ +

               New checkout user ID ______________
```

2. Enter the Host Encyclopedia name of the model.
3. To use the model name selection list instead, tab to the model name and press F4. (See the note on selection lists.)
4. Enter the subset name and press Enter, or tab to the subset name and press F4 to use the subset selection list.
5. Enter the user ID of the person to whom you want the subset that is checked out. Press Enter.

Override Checkout Status for a Subset

Override Checkout Status for Subset changes a subset's checkout status from checked out to checked in and makes another checkout of the subset possible. This transaction requires no confirmation.

All work that is done on the workstation since the last checkout or update is lost. All work that is previously updated to the Host Encyclopedia will not be rolled back.

You can override the checkout status for a subset if you:

- Log on with the ID used to check out the subset.
- Log on as the model administrator (individual or group ID) and change the checkout ID to your ID.
- Log on as the encyclopedia administrator.

If the target subset is one you are working on, use override checkout status followed by a new checkout when you have made changes to the checked out subset that you do not want to check in, and you want to roll back to the status of your subset at last update.

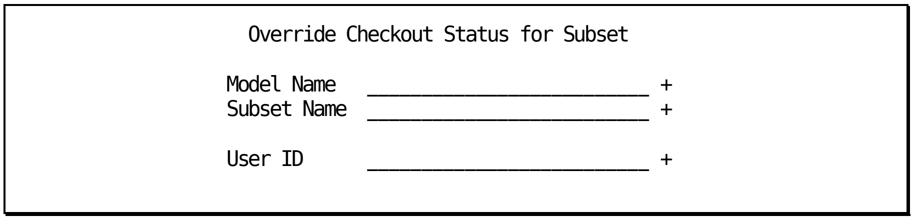
Note: The subset on the toolset is not updated to Read Only. As a result, changes can still be made but cannot be applied to the Host Encyclopedia:

- If the target subset is one you have only checked out but received an incomplete subset, override the checkout status, modify the subset definition, and check it out again.
- If you are the encyclopedia administrator, use override checkout status to check in all checked out subsets for any important objective that requires all subsets or models to be checked in.
- If you are the model administrator and the current user is unable to check in the subset, you can release the subset from checked-out status without risking updating the model with untested changes.

Note: To undo the effect of an override, create a new model from the subset and use Version Control to migrate the changes back to model originally.

Follow these steps:

1. From the Main Menu, select Host Encyclopedia Functions, Subset Management, Override checkout status for subset or enter 1.4.10.



```
                     Override Checkout Status for Subset

                    Model Name    _________________________ +
                    Subset Name   _________________________ +

                    User ID       _________________________ +
```

2. Enter the Host Encyclopedia name of the model.

   To use the model name selection list instead, tab to the model name and press F4.

3. Enter the subset name and press Enter, or tab to the subset name and press F4 to use the subset selection list.

4. Enter the TSO user ID of the person to whom the subset is now checked out. Press Enter.

Check In a Subset

To update a model on the Host Encyclopedia after working on one of its subsets, use Update And Check In Model or Update But Do Not Check In.

- Update And Check In Model returns changes to the Host Encyclopedia and surrenders control of the objects in the subset.
- Update But Do Not Check In sends changes to the Host Encyclopedia but keeps the workstation control of the objects in the subset.

Both functions are options on the Model\Encyclopedia drop-downs. Both are documented in online help.

If you check out a subset and add objects to it on the workstation, then Update But Do Not Check In, your subset maintains Delete protection and Full expansion of the new objects. No one else can use the objects. To share the new objects with other users, use Update and Check In Model. The addition of objects on the workstation also alters your subset definition.

When you check in the subset, the new objects remain in the subset definition with Modify protection and Default expansion. Before you check out the subset again, review the subset definition for added objects to determine whether the new objects still require scoping and whether the expansion and protection are appropriate.

Note: As a rule, after adding a procedure, modify the subset definition to scope on the procedure step instead of the procedure.

Subset Check In Guidelines

- Update frequently using Update But Do Not Check In, which lets you continue working where you left off.
- Update and Check In Model at least weekly. If you do not check in the model at regular intervals, a check in error occur.
- If you check out a subset with Delete protection for a scoping object, delete the object quickly and check the subset back in.
- Check in the subset promptly after completing your task to make objects you have created in a subset available to others and relinquish control of an object to whoever need it.
- Change the checkout user ID to allow a specific individual or group to continue working on the subset.

Handle System-Renamed Objects

When a subset is checked in, conflicts occur when new or renamed objects in the subset are given the same name as an existing object in the model.

Update and Check in Model or Update But Do Not Check In detects any duplication of object names and prevents the duplicate object from reaching the Host Encyclopedia by renaming the object from the subset. It renames the new, duplicate object by adding a unique object ID number to the name of the new object.

For example, if the duplicate object is an entity type that is named PAYMENT and its object ID number is 45678, the renamed entity type is PAYMENT-45678.

When you encounter renamed objects, examine them and the original objects to decide which one to keep. If they are really duplicates, change all references to indicate the correct object and allow deletion of the other. If they are not duplicates, rename one or both objects more specifically.

Chapter 4: Maintaining and Using a Subset

Change a Subset Definition

Two Host Encyclopedia subset functions change subset definitions:

- Modify Subset Definition lets you add scoping objects to a subset definition. (Use option 1.4.2.)
- Display Subset Definition Summary is the easiest way to make all other changes. (Use option 1.4.3.)

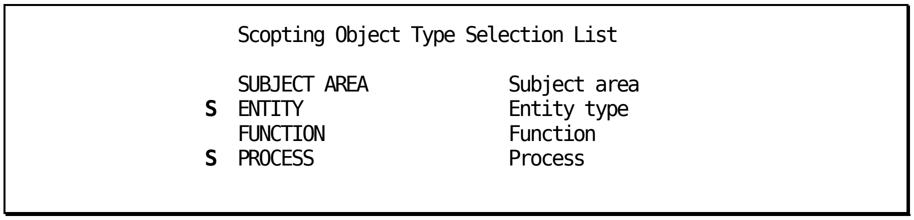
Adding Scoping Objects

To add scoping objects to a subset definition, use Modify Subset Definition.

Note: You cannot add scoping objects to a subset definition if the subset is checked out.

Follow these steps:

1. From the Main Menu, select Host Encyclopedia Functions, Subset Management, Modify Subset Definition or enter 1.4.2.
2. Complete the Modify Subset Definition screen as described.
3. Bypass the display of existing definitions by spacing over each “S” indicating previous selection.



```
     Scopting Object Type Selection List

     SUBJECT AREA              Subject area
   S ENTITY                    Entity type
     FUNCTION                  Function
   S PROCESS                   Process
```

4. Select more scoping objects and modify their definitions, if needed. For details, see the Selecting Scoping Object Types procedure in the “Add Subset Definition” chapter in this guide.

Delete Objects or Change Protection and Expansion Requests

Follow these steps:

1. From the main menu, select Host Encyclopedia Functions, Subset Management, Display Subset Definition Summary or enter 1.4.3.
2. Proceed as described in the "Creating a Subset" chapter in this guide.

Note: You cannot change a subset definition if the subset is checked out.

Check Object Consistency Before Transformation or Code Generation

Check Subset for Consistency checks the consistency of objects within a subset. The function flags problems that require correction as errors, and potential problems as warnings. You can handle most errors by changing your subset. However, we recommend that you contact Technical Support when you encounter fatal errors.

Consistency check is used to:

- Provide more diagnostics that are requested by Technical Support because of some problem encountered.
- Move forward to transformation and code generation.

Both database generation and code generation require input that is error free.

Although the warnings do not prevent you from performing the transformation and code generation, consider them an alert.

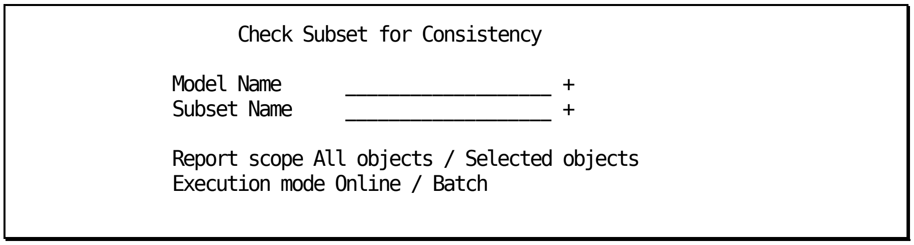
Before you can perform this task	You must correct all errors in
Transformation into Design	Analysis part of model
Code generation	Design part of model

If the subset is not checked out, the consistency check function performs a trial checkout to get the total expanded set of objects. These are then checked for consistency and the trial checkout rolled back to its initial state.

Important! Running this report is resource-intensive.

Subset Consistency Check Procedure

1. From the main menu, select Host Encyclopedia Functions, Subset Management, Check Subset for Consistency or enter 1.4.8.



```
                 Check Subset for Consistency

            Model Name        _________________ +
            Subset Name       _________________ +

            Report scope All objects / Selected objects
            Execution mode Online / Batch
```

2. Enter the Host Encyclopedia name of the model.

   To use the model name selection list instead, tab to the model name and press F4.

   Note: You can scroll or use F (Find) or L (Locate) commands on all selection lists. Enter any character except period (.) by the item you want and press Enter.

3. Enter the subset name and press Enter, or tab to the subset name and press F4 to use the subset selection list.

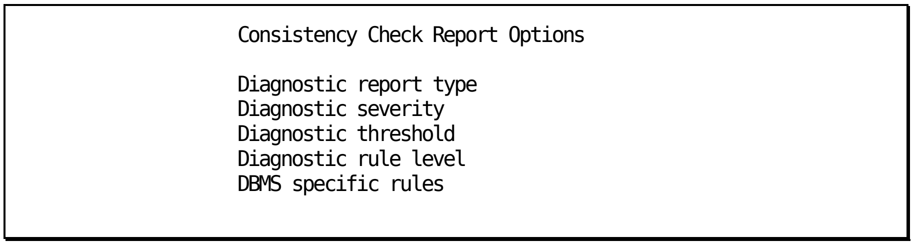
4. Select a scope for your report:

   - All objects
   - Selected objects

5. Use a slash (/) to select execution mode:

   - Online
   - Batch

   Press Enter. Other report options appear:



```
                 Consistency Check Report Options

                 Diagnostic report type
                 Diagnostic severity
                 Diagnostic threshold
                 Diagnostic rule level
                 DBMS specific rules
```

6. Specify the type of report (amount of detail):

   - N (Normal)
   - D (Detailed)
   - S (Summary)

7. Specify severity.
    - W (Warnings and errors)
    - S (Severity of warnings)
    - E (Errors only)
    - F (Fatal errors)
8. Specify diagnostic threshold, the number of messages at which to stop checking. Valid range is 1-9999. The default value is 100.

    If the consistency check produces the number of messages you specify, it stops and the Consistency Check Threshold Processing screen appears and asks you whether to continue.
9. Specify the diagnostic rule level.
    - ALL (All rules)
    - ISP (Rules for Planning stage only)
    - BAA (Rules for Analysis stage only)
    - BSD (Rules for Design stage only)
    - TD (Rules for database design only)
    - CG (Rules for code generation only)
10. Specify the DBMS-specific rule level.
    - ALL (All DBMS rules)
    - NONE (No DBMS-specific rules)
    - DB2UDB (DBMS rules specific to DB2 UDB only)
    - DB2ZOS (DBMS rules specific to DB2 for zOS only)
    - JDBC (DBMS rules specific to JDBC only)
    - MSSQL (DBMS rules specific to SQL Server only)
    - ODBC (DBMS rules specific to ODBC only)
    - ORACLE (DBMS rules specific to Oracle only)
11. Press Enter.
    - If generating online report and checking all objects, the Consistency Check Diagnostic Report appears.
    - If generating online report and checking selected objects, the Confirm Consistency Check List screen appears. It contains a list of all the objects in the subset.
12. Enter a forward slash (/) next to objects to remove them from the consistency check. Press Enter. A confirmation screen appears.

13. Press Enter to process, or enter CANCEL and press Enter to cancel.

    If generating batch report and checking all objects, an editable stream of JCL appears.

14. Edit JCL as needed.

15. Enter SUBMIT or SUB and press Enter.

    The batch job is submitted for execution.

    If generating a batch report and checking selected objects, the Confirm Consistency Check List screen appears. It contains a list of all the objects in the subset.

16. Enter a forward slash (/) next to objects to remove them from the consistency check. Press Enter. A confirmation screen appears.

17. Press Enter to process, or enter CANCEL and press Enter to cancel. An editable stream of JCL appears.

18. Edit JCL as needed.

19. Enter SUBMIT or SUB and press Enter. The batch job is submitted for execution.

Subset Consistency Check Messages

Access the complete list of consistency check messages from the Toolset Help menu and accessing the help topic on Consistency Check Error Messages.

To browse through this report, use the Up and Down function keys.

You can print the report, save it in a permanent report data set, or delete it. Press End to display the Print Report options panel and make your selection.

Report on Subset Contents or Usage

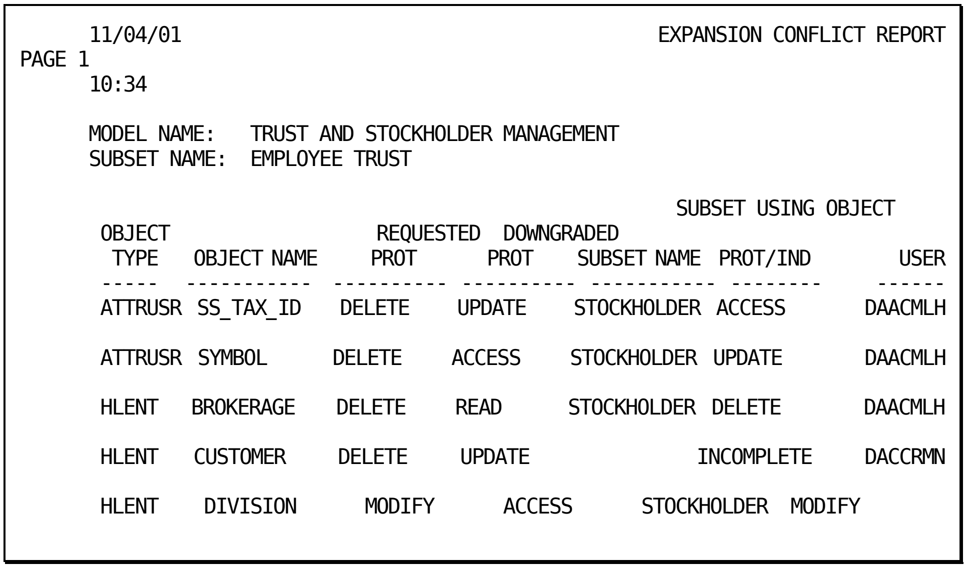
Subset reports provide online and hardcopy documentation that helps you view information in specific subsets of a model. This in turn allows you to track or anticipate the impact of impending changes. You can generate reports on any model for which one or more subsets have been defined. You generate reports on subsets already checked out although you want to rerun the reports once the subsets are checked back in by the user who have changed them.

Report Protection Downgrades to Expect at Checkout

The Expansion Conflict Report identifies potential conflicts for subset checkouts. After defining a subset, you can run the report before or after checking out the subset to identify:

- Protection downgrades in your subset
- Subsets and users that have the downgraded objects that are checked out

As a rule, run the report only if a checkout does not produce all the required objects. Running the report before checkout actually expands your subset definition: If you check out the subset after running the report, you pay for expansion twice.



```
      11/04/01                                         EXPANSION CONFLICT REPORT
PAGE 1
      10:34

      MODEL NAME:   TRUST AND STOCKHOLDER MANAGEMENT
      SUBSET NAME:  EMPLOYEE TRUST

                                                      SUBSET USING OBJECT
       OBJECT                 REQUESTED  DOWNGRADED
        TYPE    OBJECT NAME     PROT       PROT     SUBSET NAME  PROT/IND        USER
       -----   -----------  ----------  ----------  -----------  --------      ------
       ATTRUSR  SS_TAX_ID    DELETE      UPDATE     STOCKHOLDER  ACCESS       DAACMLH

       ATTRUSR  SYMBOL      DELETE      ACCESS     STOCKHOLDER  UPDATE       DAACMLH

       HLENT    BROKERAGE    DELETE      READ       STOCKHOLDER  DELETE       DAACMLH

       HLENT    CUSTOMER     DELETE      UPDATE                  INCOMPLETE   DACCRMN

       HLENT     DIVISION      MODIFY      ACCESS      STOCKHOLDER  MODIFY
```

Note: In this report:

- SUBSET NAME: EMPLOYEE TRUST is the subset being checked for conflicts.
- INCOMPLETE (fourth line under the PROT/IND column) appears, so you see the Delete Report in the current subset definition.

This report itemizes all downgraded objects, not only downgraded scoping objects. For example, it includes downgraded expansion objects:

- Attributes (ATTRUSR)
- Entity types (HLENT)

The Expansion Conflict Report shows only the first IDs that conflict with your subset. Behind these objects may be other conflicting objects. Overriding the checkout status of the objects that are shown in the report solve the problem. However, an override may not resolve conflicts behind those shown.

If a Requested Protection of Delete results in INCOMPLETE appearing in the PROT/IND column under the Subset Using Object heading on this report, it means that you did not include all usages of that item. Delete protection cannot be granted unless you have all usages of that object in the subset. To find everything you include in the subset to get delete protection, use the Object Cross-Reference Delete report.

Generating the Expansion Conflict Report

1. From the Main Menu, select Host Encyclopedia Functions, Subset Management, Reports, or enter 1.4.12.
2. Ensure that the Model Name and Subset Name fields identify the model containing the subset and the subset on which you want to report.
   - If these fields contain names, verify or replace them.
   - If empty, tab to the field and either enter the name or press F4 to display a selection list from which you select the correct name.
3. Enter 4 to select the Expansion Conflict Report press Enter.
4. Select the execution mode: Batch or Online, and press Enter.
   - If mode is online, the software generates the report, which appears in the Browse mode.
   - If mode is batch, skeleton JCL appears. Edit as needed, and enter SUB on the command line. View report in IEFRPT**.**

Report Subsets Where Each Scoping Object in Model Is Used

The Scoping Object Where Used report lists the subset scoping object types (such as entity type, function, or business system) in a given model. For each scoping object type, it lists the name of specific occurrences within the model's subsets. It also shows the name of the subset that contains the occurrences, and the objects' usage and expansion options. If objects within the subset are checked out, the report shows the ID of the users who have checked them out.

A sample report follows.

```
12/05/93                           SCOPING OBJECT WHERE USED REPORT                        PAGE 1
14:44

MODEL NAME:   ENCY REPORTS TEST MODEL 1
SCOPING OBJECT TYPE          Highest Lvl Analysis Entity Type

                                              PROTECTION
   OBJECT NAME            SUBSET NAME         CHECKOUT ID     DIR     IND     EXPANSION
   ---------------      ------------------    -----------     ---     ---     ---------
   ACTIVITY             TEST ENTITY REPORT     DACCEBB         R       R       DEFAULT

   ACTIVITY DETAIL      TEST ENTITY REPORT                     R       R       DEFAULT

   IEFUSER              TEST ENTITY REPORT                     R       R       DEFAULT

   RI_TRIGGER_ROUTINE   TEST ENTITY REPORT                     R       R       DEFAULT

   TARGET_DEFINITION    TEST ENTITY REPORT     DACCTMM         R       R       DEFAULT

   TEMPLATE             TEST ENTITY REPORT                     R       R       DEFAULT

   TEMPLATE_LINE        TEST ENTITY REPORT                     R       R       DEFAULT

   TOKEN                TEST ENTITY REPORT                     R       R       DEFAULT
```

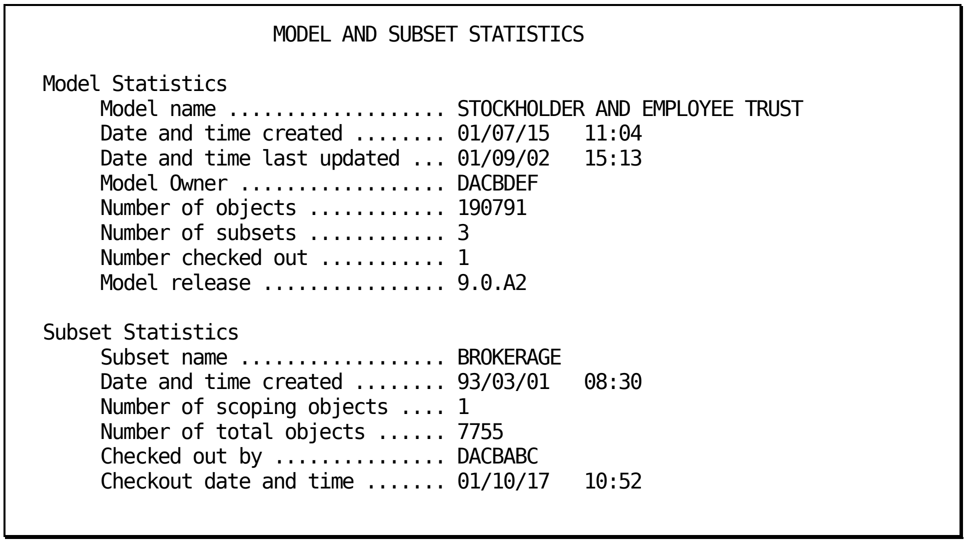
Note: In this report, the term PROTECTION (IND) refers to the protection assigned to expansion objects.

Generating the Scoping Object Where Used Report

1. From the Main Menu, select Options 1.4.12.
2. From the Subset Reports menu, select Option 6.
3. Select End to save, delete, or print.

Report Checkout Status and Object Count for Model's Subsets

The Model and Subset Statistics Report contains information about a specified subset and the model to which it belongs.



```
                       MODEL AND SUBSET STATISTICS

Model Statistics
     Model name .................. STOCKHOLDER AND EMPLOYEE TRUST
     Date and time created ........ 01/07/15   11:04
     Date and time last updated ... 01/09/02   15:13
     Model Owner .................. DACBDEF
     Number of objects ............ 190791
     Number of subsets ............ 3
     Number checked out ........... 1
     Model release ................ 9.0.A2

Subset Statistics
     Subset name .................. BROKERAGE
     Date and time created ........ 93/03/01   08:30
     Number of scoping objects .... 1
     Number of total objects ...... 7755
     Checked out by ............... DACBABC
     Checkout date and time ....... 01/10/17   10:52
```

Note: This function reports the actual number of objects in the model and in the subset. For the subset, it counts both scoping objects in the subset definition and the total number of objects in the expanded subset.

If the subset is not already checked out, the display statistics function performs a trial checkout. The display statistics function then expands all scoping objects within the subset and checks each object's requested protection for conflicts with subsets that are currently checked out. It then rolls back the trial checkout, leaving the subset in the initial state.

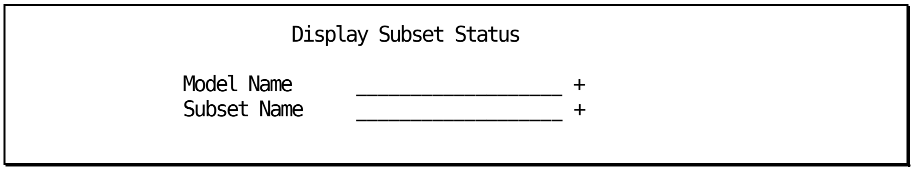
Note: The report can cause the database contention.

If the subset is large, the expansion of scoping objects require as much time as checking out the subset. Also, the larger the subset, the more resource-intensive the report.

If the subset is checked out, the report does not re-expand the subset but only counts the number of objects.

Displaying Subset Statistics

1. From the Main Menu, select Host Encyclopedia Functions, Subset Management, Display Subset Statistics or enter 1.4.11.



```
                    Display Subset Status

         Model Name      __________________ +
         Subset Name     __________________ +
```

2. Enter the Host Encyclopedia name of the model.

   To use the model name selection list instead, tab to the model name and press F4.

   Note: You can scroll or use F (Find) or L (Locate) commands on all selection lists. Enter any character except period (.) by the item you want and press Enter.

3. Enter the subset name and press Enter, or tab to the subset name and press F4 to use the subset selection list.

Report on Subset Details

1. From the Main Menu, select Host Encyclopedia Functions, Subset Management, Subset Reports or enter 1.4.12.

2. Enter the model name and subset names or press F4 to select names from selection lists.

   Note: You can scroll or use F (Find) or L (Locate) commands on all selection lists. Enter any character except period (.) by the item you want and press Enter.

3. Enter the number of the desired report and press Enter. Options include:

   - Attribute definition, which contains information about the attributes of all entities within a specified subset or subject area.
   - Elementary process information view definition, which contains the definitions of the information views (import and export data) of the subset's elementary processes.
   - Entity definition, which contains information about the entity types and entity subtypes of a specified subset, subject area, partitioning, or entity subtype.
   - Expansion conflict, which contains information about objects with the specified protection.

- Function definitions, which contain information about the functions and processes of a specified model including the hierarchy in which the functions and processes occur. The three reports are:
  - Function Hierarchy Report
  - Function Definition Report in Hierarchical Order
  - Function Definition Report in Alphabetical Order
- Scoping object where used, which lists where scoping objects are used related to other objects in the model.
- Subset contents, which lists all objects that are owned by a given subset, all of their properties, their unique identifiers, and the subset's associations (relationships between its entity types).

Copy a Subset

Copy Subset Definition creates a subset definition that lists the same subset type, scoping objects, protections, and expansions as the subset being copied. Remember that Copy Subset Definition does not cause the subset to be expanded. It only copies the initial list of scoping objects.

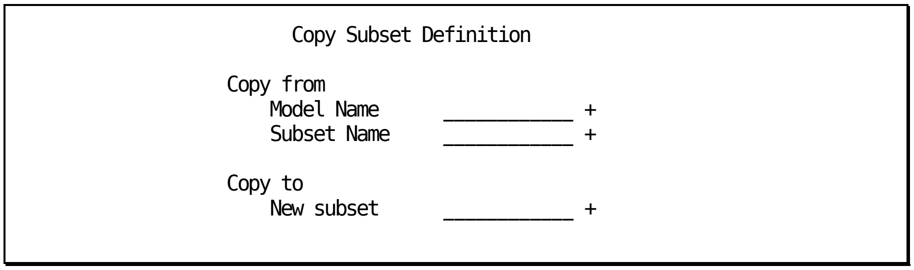
To save time, use Copy Subset Definition. By copying a subset and then modifying it, you can create variations of a subset without having to recreate the basic subset each time.

Note: Do not attempt to copy an expanded subset after checkout to a toolset for shared use by several users. Always use the Copy a Subset function on the Host Encyclopedia to perform a subset copy.

To change the copy, use Modify Subset Definition or Subset Definition Summary.

Follow these steps:

1. From the Main Menu, select Host Encyclopedia Functions, Subset Management, Copy Subset Definition or enter 1.4.5.



```
                       Copy Subset Definition

                  Copy from
                      Model Name       ____________ +
                      Subset Name      ____________ +

                  Copy to
                      New subset       ____________ +
```

2. Enter the Host Encyclopedia name of the model. To use the model name selection list instead, tab to the model name and press F4.

   Note: You can scroll or use F (Find) or L (Locate) commands on all selection lists. Enter any character except period (.) by the item you want and press Enter.

3. Enter the subset name and press Enter, or tab to the subset name and press F4 to use the subset selection list.

4. Enter a name for the new subset. Press Enter. To return to Subset Management, click Exit.

Rename a Subset

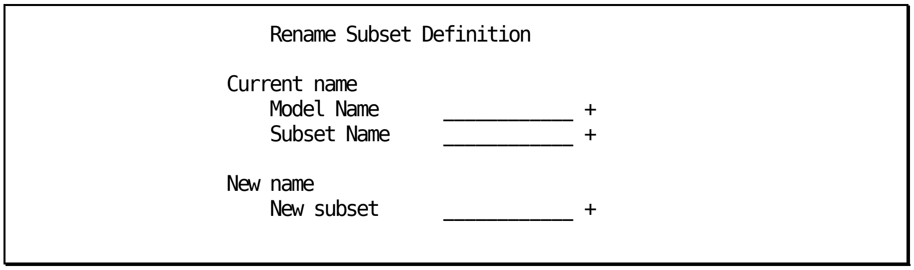
Rename Subset Definition changes the name of a subset definition. It does not affect the contents of the subset.

The rename subsets as content changes. Implement subset naming standards so that subset names indicate content. The more descriptive the name, the more easily you can determine from the subset list what is available and what is checked out.

Note: If a subset is checked out, you cannot rename it or cannot select it from a subset selection list.

Follow these steps:

1. From the Main Menu, select Host Encyclopedia Functions, Subset Management, Rename Subset Definition or enter 1.4.6.



```
                Rename Subset Definition

            Current name
                Model Name      ____________  +
                Subset Name     ____________  +

            New name
                New subset      ____________  +
```

2. Enter the Host Encyclopedia name of the model. To use the model name selection list instead, tab to the model name and press F4.

   Note: You can scroll or use F (Find) or L (Locate) commands on all selection lists. Enter any character except period (.) by the item you want and press Enter.

3. Enter the subset name and press Enter, or tab to the subset name and press F4 to use the subset selection list.

4. Enter the new subset name and press Enter.

Delete a Subset

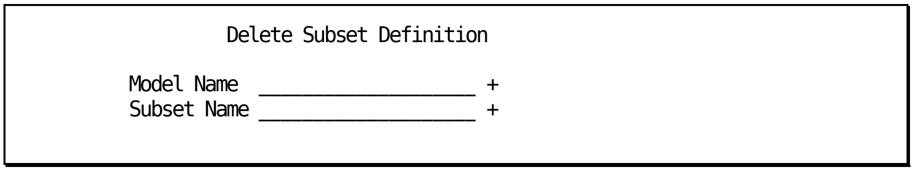
Deleting a subset removes the subset definition from the Host Encyclopedia. It does not delete any objects from the model. The delete subsets when they become obsolete.

Note: You cannot delete a subset definition if the subset is checked out. You want to consider reusing the subset instead of deleting it. This reduce the number of subset identifier allocations.

1. From the Main Menu, select Host Encyclopedia Functions, Subset Management, Delete Subset Definition or enter 1.4.7.

2. Ensure that the Model Name and Subset Name fields identify the model containing the subset and the subset to be deleted.

   - If these fields contain names, verify or replace them.
   - If empty, tab to the field and enter the name or press F4 to display a selection list from which you select the correct name.



```
                    Delete Subset Definition

          Model Name  ____________________  +
          Subset Name ____________________  +
```

3. Review the data that is displayed on the Confirm Subset Deletion screen to confirm that you identified the correct subset.

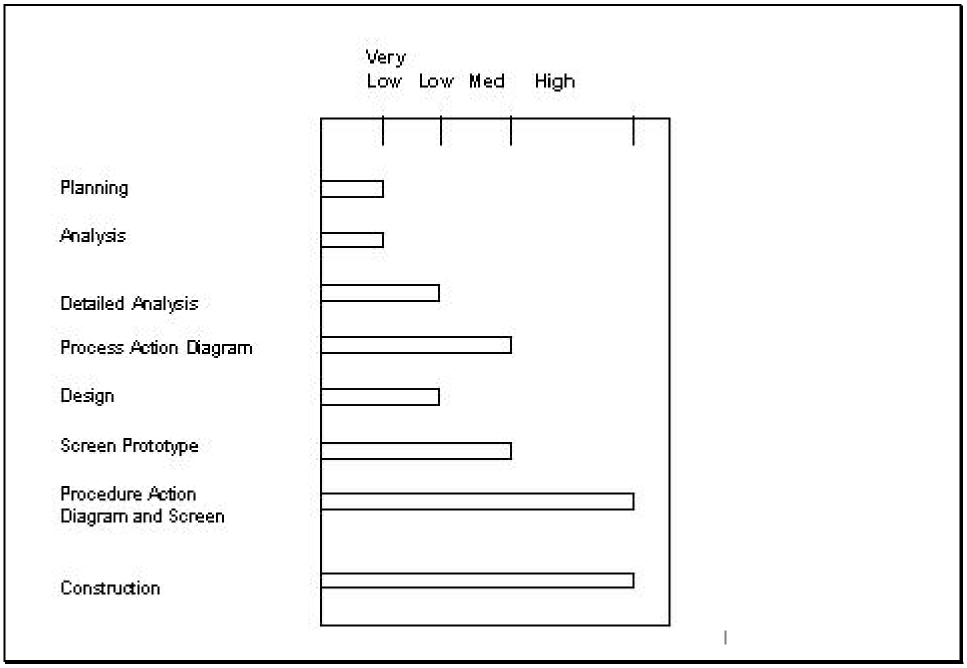
4. Press Enter to confirm the deletion or F12 to Cancel.

Chapter 5: Managing an Environment for Subsetting

Anticipate Subsetting Usage

Subsetting usage varies across the life cycle of a typical CA Gen project. This graph shows a rough approximation of the differences among the stages of the project. However, the high-level usage actually be several times the medium level of usage.

Procedure Action Diagram building and Construction are the most subsetting-intensive phases for a project because of the variety of subsets and the volume of subsetting and other Host Encyclopedia activity.



Check Out a Subset to a CSE

You can check out subsets that are created on the Host Encyclopedia to a Client Server Encyclopedia (CSE), where they can be subsetted again or checked out by a user through the Checkout Client or the Toolset. An advantage of extracting a model to a CSE is that checkouts and updates are cheaper between a CSE and the Toolset than between the Host Encyclopedia and the Toolset.

After a user checks out the extracted subset from the CSE to the workstation, that user use the Update But Do No Check In command multiple times before issuing the Update and Check In command. After the Update and Check In command is issued, you can apply these updates to the parent model on the Host Encyclopedia, which checks in the model. Then, you can delete the child model, which would be in read-only status.

To download a subset with the extract option

1. From the Main Menu, select Host Encyclopedia Functions, Subset Management, Download or enter 1.4.13.
2. Complete the Download Subset panel as follows:
   a. Identify the model containing the subset and the subset to download. To download all objects in the model, enter **ALL** as the subset name.

   b. Enter the model schema for Software version. If you do not know, enter 1.3.5 from the Main Menu to display the Model Statistics panel, where the last entry is model release.

   c. Accept the default transaction file name to use userid.IEF.TRAN as the name of the data set to be created at download or enter a different name, where your TSO userid is automatically used as the prefix.

   d. Change the default Extract option to Yes to create a child model on a CSE.

   e. Enter the Child Encyclopedia ID. If you do not know what it is, bring up the Coordination Client, and select Encyclopedia, Actions Open, select the target encyclopedia from the list and press Detail. The data in the ID field is the Child ency ID.

   f. Enter as the Child model name the name to assign to the subset or model to be downloaded.

   g. Enter the codepage.

      For a listing of supported codepages, see the *Client Server Encyclopedia User Guide*.

      Note: If you do not know what it should be, bring up the Encyclopedia Client and detail an existing model using the same language. The codepage is displayed as Language Code. This is the identifier for the language (for example, English, French, Japanese) on the destination platform (for example, Windows), here the platform where the encyclopedia resides.

   h. Accept online as the execution mode or change to batch.

If you select batch, update JCL.

1. Press Enter to begin download processing. The “IEF OK” message indicates the successful completion.

2. Verify that the server install directory for the destination CSE does not contain a LOAD.TRN file that you want to preserve. If it does, consider creating a personal directory as the destination for the LOAD.TRN file or rename the existing file to avoid overwriting it.

3. Use FTP or another file transfer utility to transfer userid.ief.tran from the Host Encyclopedia to the server install directory; name the file to be transferred LOAD.TRN.

   For example, issue the ftp command on the server, from the input subdirectory:

   ```
   ftp <input Host IP address>
   Name (input Host IP Address): <logon userid at server>
   Password: <logon password at server>
   ftp>cd '<userid on Host>.ief'
   ftp> bin
   ftp> get tran load.trn
   ftp> quit
   ```

4. Load the model that is created by the extract into the CSE. From the directory where LOAD.TRN resides, enter the following: UPLOAD -u *userid* -t LOAD.

Minimize Host Encyclopedia Contention

Host Encyclopedia contention is competition for database resources. Host Encyclopedia contention has two sources:

- CA Gen contention, which are constraints (rules) built into the CA Gen toolsets
- CA Gen contention occurs only at the model and subset level. All CA Gen constraints involve shared use of a single model. No encyclopedia-wide contention exists. If one analyst is working on Model A and another is working on Model B, CA Gen constraints create no contention. System constraints for simultaneous execution appear in the Subset Concurrency Matrix.

Here the *concurrency* means:

- A second process can access the Host Encyclopedia without being queued by a process that is already executing.
- The beginning of one process does not depend on the completion of another.
- The first process is not one that locks the entire encyclopedia during its execution.

When system contention occurs, the second process that is submitted to the Host Encyclopedia receives a message that it has been queued. That is, the second process is on hold awaiting resources that are held by the first process. The second process continues to try to determine whether the queue still applies. After ten retries, the attempt to execute the second process terminates with a resource conflict message.

Minimize Contention with Checkouts

Users have found the following techniques helpful in minimizing the contention with checkouts:

- Do the checkouts only at night and in batch in the later stages of development, when subset usage is highest.
- You want all developers to put a copy of the JCL in a JCL library to facilitate batch processing.
- Keep the checkout JCL in a common PDS. Submit the JCL at night after migrations are complete.
- Do emergency daytime checkouts only in batch.
- Use electronic mail to communicate emergency checkouts to team members. Try to schedule the activities of team members to spread out the processing load.
- Stagger the checkouts across the team so that not everyone does a checkout every night or single-thread checkouts at night.
- Check out subsets for two separate tasks. Checking out two subsets lets you keep working if checking in the first subset was unable to complete.

Note: If you use batch checkout, do a manual transfer of the file to the workstation.

Update JCL for Batch Checkout

Note: For information, see the *CA Gen Host Encyclopedia Administration Guide*.

Minimize Contention with Checkin

Users have found the following techniques helpful in minimizing the contention with the checkin:

- Do the checkins in batch during the day.

  Note: Begin batch checkins as late in the project life cycle as possible, when Procedure Action Diagramming and Construction begin in earnest. Users make fewer errors using seamless checkin than when using batch.

- Perform frequent updates without the checkin. The thumb rule is: Update after working each procedure step if possible. Be sure to update after every three to four procedure steps.
- Time your updates. If they take more than 10 minutes, you are doing too much work between updates.

  Note: If you use batch checkin, do a manual transfer of the file to the workstation.

Batch Checkin JCL

Note: For information, see the *CA Gen Host Encyclopedia Administration Guide*.

Avoid Using Subsetting for Object Deletions

Follow these steps:

- Perform the deletions of shared objects, using the Model Management utility Delete or Rename Objects on the Host Encyclopedia. Do not ordinarily delete objects using subsets.
- Become familiar with your DBA's database reorganization schedule, and perform deletions the night before reorganization.

  See the Project-Wide Delete Cycles section.

Schedule Migrations and Keep Them Small

Follow these steps:

- Do large migrations only at night and in batch. The small migrations can be done without causing contention problems.
- Schedule migrations and communicate the schedule across the project team. Developers can then ensure that other work is scheduled around the migrations.
- Designate one person to perform all migrations.

Single Thread Batch Jobs

Follow these steps:

- With standard JES, use a single job name to single-thread all migrations, copies, checkins, and checkouts that are done at night.
- Use a single job name per project or model to single thread all batch daytime activity (checkins and checkouts), except code generation.

Restrict Use of Resource Intensive Tasks

Perform the following:

Restrict use of the Expansion Conflict Report, Subset Consistency Check, and Subset Statistics because they do the same amount of work as checkout and are resource-intensive.

Use the Subset Concurrency Matrix to Schedule Tasks

The interaction of two processes can have three outcomes:

- Processes can run concurrently.
- Processes cannot run concurrently.
- Processes can run concurrently if subset names differ.

The CA Gen Concurrency Matrix that follows indicates whether the CA Gen functions can run simultaneously. The subsetting functions appear at the upper left corner.

Key
Y = Yes, the two operations can run concurrently.
U = Unique; the operations can run concurrently if the subset names differ.
<blank> = No, the operations cannot run concurrently.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1 Checkout Subset	U		U	U	U	U	U		Y			Y				Y		Y					Y	Y	
2 Upload Subset		U	U	U	U	U			Y																
3 Override Subset	U	U	U	U	U	U	U		Y			Y		Y	Y	Y	Y	Y	Y				Y	Y	Y
4 Rename Subset	U	U	U	U	U	U	U		Y			Y		Y	Y	Y	Y	Y	Y				Y	Y	Y
5 Delete Subset	U	U	U	U	U	U	U		Y			Y		Y	Y	Y	Y	Y	Y				Y	Y	Y
6 Copy Subset	U	U	U	U	U	U	U		Y			Y		Y	Y	Y	Y	Y	Y				Y	Y	Y
7 Cre Mdl From Subset (SRC)	U		U	U	U	U	U		Y			Y				Y		Y					Y	Y	
8 Cre Mdl From Subset (DST)									Y																
9 Codegen	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
10 Rename Model									Y																
11 Delete Model									Y																

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
12 Copy Model (SRC)	Y		Y	Y	Y	Y	Y		Y			Y				Y		Y					Y	Y	
13 Copy Model (DST)									Y																
14 Object Rename			Y	Y	Y	Y			Y																
15 Object Delete			Y	Y	Y	Y			Y																
16 Migration (SRC)	Y		Y	Y	Y	Y	Y		Y			Y				Y		Y					Y	Y	
17 Migration DST			Y	Y	Y	Y			Y																
18 Adoption (SRC)	Y		Y	Y	Y	Y	Y		Y			Y				Y		Y					Y	Y	
19 Adoption (DST)			Y	Y	Y	Y			Y																
20 Model Backup									Y																
21 Model Restore									Y																
22 PI Import									Y																
23 PI Export	Y		Y	Y	Y	Y	Y		Y			Y				Y		Y							Y
24 Reports	Y		Y	Y	Y	Y	Y		Y			Y				Y		Y					Y	Y	
25 Conversion			Y	Y	Y	Y	Y		Y																
26 Intelligent Regen	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
27 Packaging	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

1 Subset ALL - the entire model

Note: This matrix shows the concurrency enforced by the Host Encyclopedia. Other types of contention, such as that caused by DB2, cannot be determined from this matrix

Scenario: Subset Checkout and Code Generation

If one user wants to generate code for Model A and another wants to check out a subset of Model A simultaneously, here is how to check the matrix for concurrency.

Find CHECKOUT SUBSET at the top of the matrix. Look down the column below it until it intersects the horizontal line for CODE GEN. At the intersection, the letter “Y” appears. Therefore, both processes can proceed simultaneously.

Scenario: Subset Checkin and Code Generation

One user wants to generate code for Model A, and another wants to check in (update) a subset of Model A simultaneously. Can both processes execute simultaneously?

Find UPLOAD SUBSET at the top of the matrix. Look down the column until it intersects the horizontal line for CODE GEN. The intersection shows a period. Therefore, the second process is queued and can time out.

Scenario: Subset Checkout and Adoption

You can check out a subset of the source model at the time of an adoption, but you cannot check out a subset of the adopted model.

For some processes, like the adoption, the result varies depending on whether you are working with the source (src) or destination (dst) model. For example, if you want to adopt objects from Model A into Model B, you can check out a subset Model A simultaneously, but you cannot check out a subset of Model B.

Use Power Subsetting Strategies

Power subsetting requires roles, responsibilities, education, and reference material.

Roles and Responsibilities

Project subsetting involves different roles, which typically give developers more responsibilities. If at all possible, assign these roles to the same people throughout the life of a project.

- The project team leader assigns work based on cohesive sets of objects.
- The project team members define and check out subsets according to guidelines.
- Group the project team member's responsibilities by functional area to reduce contention and implement changes with a minimum of checkouts.
- The data administrator defines and checks out subsets for data model changes after the model is stable. This is typically done to protect the data model from inadvertent changes.
- The database administrator defines and checks out subsets for technical design tasks.
- The project subset expert assists the project team leader with task assignments, troubleshoots subsetting problems, and also create starter subsets for each unit of work to be assigned.

- Project subset experts are familiar with the application, with CA Gen, and experienced at complex subset definition. They also review subset definitions occasionally for effectiveness. This is usually a part-time role.
- Internal CA Gen Technical Support troubleshoots hardware and software problems that are related to subsetting.

Reference Material

Every team member needs the information in this documentation, especially the subsetting concepts, expansion tables, and task tables.

Apply Effective Project Management Techniques

Project management is the identification and assignment of tasks within a project. Effective subsetting enables effective project management.

The following project management factors contribute to effective subsetting.

- Appropriate scoping of projects
- Use of multiple models
- Inclusion of subsetting considerations into the project plan
- Effective work assignments
- Use of life-cycle techniques
- Managing change

Project Scoping

The Host Encyclopedia has been successfully used on large projects (over 350 entity types) and with large project teams (over 100 individuals) using a single CA Gen model. All projects of this size require exceptionally careful planning and management of subsetting.

An easier approach is a phased series of small projects, which allow smaller teams. The smaller project teams allow simpler communication and control structures and allow the team to use the experience of the first project to help development of subsequent projects.

Use of Multiple Models

If you choose to tackle a large project, you can benefit from dividing the project into two or more models and still keep shared objects at the same version level across all models.

General Guidelines

To see the general guidelines for deciding whether to use one model or multiple models, click Table.

The following table shows guidelines for deciding to use one model or multiple models.

Factors Favoring Use of	One Model	Two or More Models
Project size	1 to 100 entity types	over 100 entity types
Project team size	1 to 50 members	over 50 members
Degree of sharing	high	low
Release strategy	all at once	phased

When deciding whether to divide a project into multiple models, weigh the cost of increased version control against the benefit of more flexible development and the reduced subsetting activity against each model.

Note: For more information, see the “Checking a Subset In or Out of the Encyclopedia” chapter in this guide.

If you decide to use multiple models, it is best to delay splitting up the initial project model as long as possible to minimize the cost of the versioning. As stated earlier, subsetting usage does not reach its peak until the Procedure Action Diagram building stage of Design.

Subsetting Considerations

In planning to use subsetting:

- Include training requirements.
- Develop the standards for subsetting usage.
- Sequence assignments that are based on task dependencies.
- Assign work that is based on groupings of highly interdependent objects.
- Choose life-cycle techniques that minimize the need for subsetting.

Develop Standards for Subsetting Usage

Develop the standards with an emphasis on shared objects, project-wide delete cycles, data model change sessions, and subset naming conventions.

Shared Objects

Among the most important standards to develop for subsetting involve the creation and treatment of common objects. In particular, define the following common objects early in the Design phase:

- Action blocks that are used for validation
- Work attribute sets
- Exit states
- System defaults
- Screen templates

Test shared Design objects thoroughly for correctness and ease of implementation by the developers. To ensure the readability and usability, implement and test a small set of screens and review them with the business user.

If you wait too long to define common objects, developers create different objects for the same purpose. For example, they create multiple exit state messages for the same condition.

Carefully coordinate changes to common action blocks. Instead of checking in changes as they are made, consider keeping the action blocks on the workstation until the changes have been tested.

Consider protecting common objects by checking them out before checking out subsets for further development.

If you are using workstation Construction, work out a plan for sharing generated executables, RI triggers, and databases. Decide who generate each, how to share, what is available and when.

See the Protection Subsets section in this guide and Construction subsetting task recommendations.

Example: RI Trigger Names

To ensure consistency across multiple workstations, assign RI trigger names on the mainframe. This can be done without actually generating the triggers.

To allow developers to use the same RI trigger modules with different subsets (or with subsequent uses of the same subset):

- Checkout a subset containing the data model and technical design and generate the triggers on the workstation.

  The workstation modules are then given the same names as those on the Host Encyclopedia. These triggers can be copied from one workstation to another.

- Checkout the subsets for code generation.

  References to RI trigger modules use the names from mainframe generation and therefore is consistent with workstation-generated modules.

Repeat the process as needed when the data model changes.

Project-Wide Delete Cycles

The deletion of shared objects deserves special attention. This is one approach.

Before deleting an object:

- Rename the object with a prefix that indicates impending the deletion.
- Do not use objects that are tagged with the deletion prefix.
- Eliminate existing references to the object.
- Request the deletion. See Implement Change Control.
- Coordinate deletion cycles so that all references are removed and all subsets checked in.
- Use the Host Encyclopedia Delete Report to identify any references that have been overlooked.
- Use Host Encyclopedia Rename or Delete Objects when possible.
- Define and use protection subsets to prevent inadvertent deletion, modification of essential objects, or both.

Protection Subsets

Protection subsets are used to prevent inadvertent deletions, modifications, or both. They also enable the subset owner to perform required modifications without having to check out a subset, because the protection subset is typically checked out and has been granted the correct protection. For example, the DA/DBA would typically be the owner of the ERD/TD protection subset and use it to change the data model or data structure.

Protection subsets are defined as follows:

- The data model and associated data structure
- Global exit states

- Business system defaults
- Shared the objects such as common action blocks and work attribute sets

Project model managers own or assign protection subsets. These subsets are the first checked out from the model following any full check-in or model maintenance activity. They are organized into logical subset groupings to minimize the time that is required to override checkout status and redownload as new objects are added.

To protect or modify the data model and its associated data structure:

Subset Name	Scoped Objects	Protection	Expansion
Protect ERD/TD	Root subject area	Modify	Default

Note: If the database is scoped, it must have Delete protection to do transformation. Expansion of the root subject area by itself automatically provides the database with Delete protection and enables transformation.

To protect or modify stable global exit states:

Subset Name	Scoped Objects	Protection	Expansion
Protect EXIT STATES	Exit states	Modify	Default

To protect or modify:

- Business system defaults/Existing commands
- Existing business system exit states/Existing templates, video attributes, and edit patterns

Subset Name	Scoped Objects	Protection	Expansion
Protect BUSINESS SYSTEMS	Business Systems	Modify	Short

This subset definition does not prevent the addition of new exit states or templates, because any number of users can still have Access protection on the business system while it is checked out with Modify.

To protect/modify shared objects, such as common action blocks and work attribute sets:

Subset Name	Scoped Objects	Protection	Expansion
Protect SHARED OBJECTS	Common Action Blocks	Modify	Default
Protect SHARED OBJECTS	Work Attr Sets and $IEF	Modify	Default

Sequencing Assignments

The following examples of using task dependencies to sequence assignments:

- **During Analysis**-Use the Activity Dependency Diagram to sequence Analysis assignments. For example, if Function A enables Function B, analyze the data and activities for Function A first. Failure to analyze A first require subsequent changes to B.
- **During Business System Implementation (BSI)**-Work first on procedures that create data in tables that other procedures reference. For example, if Procedure X creates Entity Type A and Procedure Y reads Entity Type A and creates Entity Type B, then develop Procedure X first.

Effective Work Assignments

Assign work that minimizes sharing and allows greater independence - according to the stage of model development:

- During Data and Activity Analysis

  To analyze and document a parent activity and its subordinate activities and associated data, assign one person.

  Note: The individual still need to work with others to do some of the analysis.

- During Interaction Analysis

  To develop Process Action Diagrams, assign one person all elementary processes of a parent activity.

- During Design

  To develop Procedure Action Diagrams, assign one person if the diagrams:

  - Share processes and data
  - Have many flows between them
  - Before extensive action diagramming, verify that the data model is stable. Some changes to the data model can cause significant rework depending on the type of change and how widely the data object has been referenced in the model.

Alternatives

It is more appropriate to assign each individual only a few procedures if you must produce tangible results quickly such as implementing all or part of the business system to test the design.

Choosing Life-Cycle Techniques

Use the following life-cycle techniques to reduce the need for subsetting, minimize the impact of changes to the data model, and improve developer interaction:

- **Defer action diagramming**-Instead of building Process Action Diagrams before starting Design, build them after the procedures and screens are approved.
- **Prototype screens using literals**-Instead of building a screen prototype that is based on information views (which use entity types and attributes); create screens that contain only literals. This prototype can even be built in a separate model.
- **Add dialog flows after Procedure Action Diagrams are tested**-Instead of adding dialog flows at the start of Design, plan the overall dialog flow. The flows can then be added to the model after the Procedure Action Diagram and screen have been thoroughly unit-tested. Adding flows last reduces the probability that you accidentally include the procedures of other developers in your subset.
- **Use workstation construction for testing**-Instead of generating code and unit testing using host construction, unit test on the workstation. Using the workstation construction minimizes the number of updates to the Host Encyclopedia.

  Using the workstation construction also reduces the contention on the Host Encyclopedia by minimizing the number of code generation jobs.

  Note: The life-cycle techniques are not appropriate for all projects.

Implement Change Control

Change control is managing the frequency and methods of changing model objects. Change control plays a vital role when the model objects are shared and subsetting is the tool that is used to change the objects.

Effective change control is critical to the timely completion of projects. Change haphazardly or without communicating them requires time for resubsetting, recreating a database, regenerating referential integrity triggers, recreating test data, or both.

The following change control techniques prove helpful:

- Create change request.
- Analyze change request impact.

- Review change request.
- Schedule change, if approved.
- Implement change.
- Communicate change request completion.

Create a Change Request

Define separate forms for each type of change. For example:

Type of Change	Content of Change Request
Entity type	Name Description Alias DSD Name Volume Identifier Mutually Exclusive
Data Structure object* *Other Data Structure and Data Store objects can be changed at the discretion of the DBA.	Table name Column names Column properties Denormalized fields

Deleting an Object

In requesting the deletion of an object, first determine whether the model contains references to the object:

Type of Object	How to Handle Change
If no references to the object in model	Only one change request required.
If object is a mandatory attribute, relationship, or entity type that has mandatory relationships with other entity types:	Two change requests required: Make object optional and rename it to include XXX (to be deleted). The rename warns team members not to use it. Request deletion of object.

Analyze the Change Request Impact

The impact of a change request is greatest when the change affects an object share by multiple projects.

Changes Across Projects

When multiple projects share an object, the change request is submitted to the custodial team of the object. Once the change request is approved, schedule it with the other internal change requests for the custodial model and the sharing models. Haphazard model migrations can cause as much disruption as haphazard manual data model changes.

Two CA Gen reports can ensure evaluation of shared objects. The Where Exists report can identify all models that contain a version of an object. The Compare Aggregate Object report indicates whether the objects were last changed by the same user ID at the same date and time.

Changes Across Sharing Models

If the projects that share an object are testing against the same integrated database, change in all models as part of the same change session-usually starting with the custodial model. Otherwise, the unchanged models risk abending when running against the changed database.

Review Change Request

Each change request is reviewed by:

- The data administrator
- The DBA
- The project leader
- Other project teams that use the shared object

Change requests are evaluated for the benefit it bring compared to the impact it have on the project or projects. For example, changing the identifier of an important entity type result in rewriting many Process Action Diagrams, Procedure Action Diagrams, and screens, and changing the structure of many data records.

Do not hurry the review process. It is uncommon for a third or more of change requests to be refused or modified.

Schedule Change

Schedule all regular, approved changes for a particular data model change session. Each change session consists of performing a set of changes so that team members do not have to reinstall the database and referential integrity triggers on their workstation each time the data model changes.

The time that is required for a session is usually one to eight hours. It varies according to these factors:

- Amount of preparation
- Number of changes
- Amount of data to be backed up and restored

Schedule data model change sessions when they have the least impact on the project teams, usually in the evening or over a weekend. It is easiest if team members check in their subsets to ensure against conflicts, especially if deletions are planned.

High-priority change requests can be done between data model change sessions. Make such changes sparingly, however, because they can cause extra work for the team.

Choosing a Change Mechanism

Once a change has been scheduled with the appropriate projects, choose a change mechanism.

- **Migrate the changed objects**-You can do this from an aggregate set that contains the changes to the shared objects.
- **Rekey the changes**-Use this method when the change cannot be migrated automatically. For example, subject area, entity type, and relationship deletes cannot be migrated. Rekeying can be more reliable if done using the Record/Replay Audit feature.

Implement Change

The data administrator or DBA define a data maintenance subset with the following definition:

Scoping Object	Protection	Expansion
Root subject area	Modify	Default

This subset definition allows developers to build and construct action diagrams while preventing anyone but the data administrator from changing the data model. This definition also lets someone perform transformation or retransformation.

To avoid protection downgrades, checkout the data maintenance subset when no one else has subsets checked out.

Cleanup

After all changes have been made:

- Retransform the data model to resolve the changes in the Data Structure List.
- Save the retransformation reports as a record of the changes that are made to the subset.

Next the DBA adds the final touches to the Data Structure List:

- Foreign key names
- Entry point names and properties
- Tablespace names and properties
- Entry point field sequencing

Tips for DBAs: Best Practices

Use the following guidelines:

- Assign tables that are required by a single procedure to the same database during unit testing.
- If you change table names, column names, or column properties outside CA Gen, be sure to make corresponding changes in the Data Structure List and then install them using the system-generated DDL. This ensures that the system-generated programs run against the database.
- If you make the table or column name changes by changing the DSD names in the data model before to transformation, be sure to make corresponding changes in the Data Structure List. CA Gen uses DSD names when it transforms.
- If you change a table or column name after transformation, do so by changing the table name or column name in the Data Structure List. CA Gen does not use DSD names during retransformation.
- If an attribute is added to an existing transformed entity type, be sure to resequence attributes of retransformed entity types in the Data Structure List. Retransformation puts the added attribute's column at the end of the list.

- If any change is made to Referential Integrity (RI) rules on the data model, be sure to run retransformation or the RI process.
- If DBMS RI is chosen, alter the existing tables ensure that they are defined with the most current delete rules.

Consult your DBMS reference guide for other recommendations.

Scenario: Preparing for a Data Model Change Session

1. The data administrator informs the team of the date and time to check in their subset and when they can check it out again.
2. The data administrator runs the Delete report to verify that developers have removed references to objects that is deleted.
3. The data administrator changes the data model using the data maintenance subset. As each change request is implemented, it is signed and dated.
4. The data administrator runs consistency check against the data model to ensure that all changes were done correctly.
5. The DBA runs retransformation so that the data model changes are reflected in the Data Structure List.

   Note: Not all changes to the data model are reflected in the Data Structure List. For example, adding and deleting permitted values cause no change to the Data Structure List because they are not implemented directly into the database.
6. The DBA prints all reports resulting from retransformation to create a record of the changes made.
7. The DBA modify the Data Structure List and the Data Store List to meet project, corporate standards, or both.
8. The model manager works with the developers, data administrator, and DBA to define the aggregate sets or rekeying required to migrate shared objects from other models.

Scenario: Handling a Data Model Change Session

1. The data administrator updates the Host Encyclopedia by checking in the data maintenance subset.

   If the subset is not checked in, developers can get only Read protection to newly added entity types and attributes. Read protection is not sufficient to create information views or expected effects.

   Change any shared objects that Version Control does not handle.

2. The data administrator performs any deletes using the Model Management facility on the Host Encyclopedia.

   Deleting an entity type or attribute causes a cascade deletion that eliminates the data record or field.

3. The model manager performs all planned migrations that are based on predefined aggregate sets.

4. The data administrator checks out the data maintenance subset again to prevent a developer from accidentally changing the data model.

   See the data maintenance subset definition in the Implement Change section.

5. The DBA generates the DDL on the mainframe so that any mainframe databases can be updated to the latest table and column definitions and so that any test data is converted.

6. The DBA uses Intelligent Regeneration to update only the host RI triggers that need updating.

7. The data administrator creates the database and the RI triggers to ensure that workstation construction works.

8. The data administrator runs a Delete report for each object that is deleted at the next data model change session.

9. The project leader can use the list to make assignments for removing references to the objects.

10. Inform the team (and other sharing project teams) of the status and any issues arising from the data model change.

11. Project team members use Intelligent Regeneration to help them create a new version of the mainframe load modules.

    Note: A change to the data model deletion rules require the regeneration of action diagrams that call RI modules, because they also contain some of the code that enforces RI.

Chapter 6: Task Tables

General Subset Definitions

General subset definitions are definitions for Analysis, first Design subset, procedure maintenance, and workstation Construction.

Analysis Subset

To scope a typical Analysis subset, use the following subset definition:

Subset Type	Scope Occurrences of this Object Type	Protection	Expansion	Note
Design	Process	Delete	Default	1, 2
	Entity Type	Access	Short	
	Work Set	Access	Default	
	Business System	Access	Short	

Process

In the Analysis subset, scoping a process with Design subset type, Delete protection and default expansion includes these objects at these protections:

Objects Included by the Scoped Process	Protection	Note
Scoped process	Modify	1, 2
Parent, grandparent processes	Access	
Child, grandchild processes	Delete	
Dependencies for child processes	Delete	
Process information views	Delete	3
Process expected effects	Delete	
Entity type in PAD view	Access	
Entity type in expected effects	Access	
PAD (if elementary)	Delete	

Objects Included by the Scoped Process	Protection	Note
Action block USEd by PAD	Delete	4

Note:

1. The scoped process, requested at Delete protection, is downgraded to Modify, but all the child and grandchild processes are included with Delete protection.
2. To delete the scoped process, scope its parent also at Modify protection.
3. The only way to delete a view for a process is to have Delete protection on the process. If you request Delete protection for the process and receive a protection downgrade, CA Gen still lets you delete as many views as possible.
4. The first time a process action block is USEd in another action block, the process, or process action block must be scoped with modify protection so that the OWNS association can be added.

Entity Type

1. Scope on all entity types you want to add to your process in the general Analysis subset. Once the entity type is referenced in a view or expected effect, you do not need to include it in your subset definitions.
2. The short expansion option of an entity type minimizes the number of shared companion entity types that are included in the subset.

Work Set

1. Scope on all work attribute sets you want to add to your procedure in the general Analysis subset.
2. Do not scope on the system-supplied work set because it is automatically included.

Business System

1. Scope on the business system if you include exit states from that business system in your PAD (using the general Analysis subset).
2. Once an exit state from the business system is in your subset, CA Gen includes the short expansion of the business system automatically.
3. Short expansion of the business system was chosen to prevent inclusion of procedures.

First Design Subset

To scope the first Design subset, use the following subset definition.

Note: The purpose of the first Design subset is to initialize the procedure. Once the first Design subset has been checked in, use the definition for a typical Design subset.

Subset Type	Scope Occurrences of this Object Type	Protection	Expansion
Design	Skeleton Procedure	Access	Default
	Processes to implement	Access	Default
	Work Attribute Sets	Access	Default
	Common Action Blocks	Access	Default
	Entity Types	Access	Short
	Business System	Access	Short

Note: Use only short expansion for the business system. It is not necessary to scope on a template separately.

Skeleton Procedure

If you want to start your procedure by copying from an existing skeleton, you must scope on that procedure with Access. If you receive Read Only protection, CA Gen does not allow the copy.

Processes to Implement

1. If the procedure implements elementary processes, scope on the implemented processes with Access. If the protection is Read Only, CA Gen does not allow them to be implemented.
2. Full expansion of the process is not required because default expansion brings in the import and export views.

Work Attribute Set

1. Scope on all work attribute sets you want to add to your procedure.
2. Do not scope on the system-supplied work set because it is automatically included.

Common Action Blocks

Scope on all common action blocks you want to USE in your procedure.

Entity Types

1. Scope on all entity types you want to add to your procedure.
2. The short expansion option minimizes the number of shared companion entity types that are included in the subset.

Procedure Maintenance Subset

To scope for procedure maintenance, use the following subset definition:

Subset Type	Scope Occurrences of this Object Type	Protection	Expansion	Note
Unit Test	Procedure Step or Procedure	Delete Modify	Default Default	1,2
	Work Attribute Set	Access	Default	
	Common Action Blocks	Access	Default	
	Entity Types	Access	Default	

Note:

1. Scope on the procedure step if it has dialog flows.
2. Scope on the procedure if it has no dialog flows or requires dialog flow maintenance.

Important! Do not scope on the business system or a template. Scoping on them is unnecessary and can create a large subset that interferes with the work of other users.

Procedure Step

1. Scoping the procedure step keeps you from getting dialog flows and hence other procedures of the people.
2. The Delete protection that is requested is always downgraded to Modify. However, you can still:
   - Delete all views that are not matched using a dialog flow.
   - Delete the fields from a screen.
   - Delete the statements from an action diagram.
3. Do not get full expansion on the procedure step because full expansion brings in all nested action blocks with Delete protection and prevents others from using the action block in their PrAD.

Procedure

1. You can scope on the procedure when the number of its expansion objects is small (it has no dialog flows in or out). Do not request Delete protection on the procedure because Delete also brings in all nested action blocks.
2. To maintain dialog flows, you must scope on the procedure.

3. However, scoping on the procedure step, Modify/Short, Design subset type, allows you to add the flows without bringing down all associated Procedure Action Diagrams (PrADs).

Work Attribute Set

1. Scope on all work attribute sets you want to add to your procedure. Once the work attribute is referenced in an information view, scoping on the work attribute is unnecessary.
2. Do not scope on the system-supplied work set because it is automatically included.

Common Action Blocks

1. Scope on all common action blocks for which you want to add references:
   - By the PrAD
   - By one of the common action blocks USEd by the PrAD
2. If you must change a common action block, try Modify protection first. Use Delete protection only if you delete views from a common action block.

Entity Type

1. Scope on all entity types you want to add to your procedure. Once the entity type is referenced in a view, scoping the entity type again is unnecessary.
2. Avoid scoping on data tables because doing so makes the subset much bigger without adding any functionality.
3. The short expansion option minimizes the number of shared companion entity types that are included in the subset.

Workstation Construction Subset

The scope for Workstation Construction is as follows:

Subset Type	Scope Occurrences of this Object Type	Protection	Expansion	Note
Unit Test	Procedure Step or Procedure	Delete Modify	Default Default	1,2,3
	Work Attribute Set	Access	Default	
	Common Action Blocks	Access	Full	

Notes:

1. Unit Test subset type brings in all necessary data tables for code generation and RI trigger generation.
2. Scope on the procedure step if it has dialog flows.
3. Scope on the procedure if:
   - It has no dialog flows.
   - It requires dialog flow maintenance.

Important! Do not scope on the business system or a template.

Procedure Step

1. Scoping the procedure step keeps you from getting dialog flows and hence other procedures of the people.
2. The Delete protection that is requested for the procedure step is always downgraded to Modify. However, you can still:
   - Delete all views that are not matched using a dialog flow.
   - Delete the fields from a screen.
   - Delete the statements from an action diagram.
3. Do not get full expansion on the procedure step because full expansion brings in all nested action blocks with Delete protection and prevents others from using the action block in their PrAD.

Procedure

1. You can scope on the procedure when the number of its expansion objects is small (it has no dialog flows in or out). Do not request Delete protection on the procedure because Delete also brings in all nested action blocks.
2. To maintain dialog flows, you must scope on the procedure.

   However, scoping on the procedure step, Modify/Short, Design subset type, allows you to add the flows without bringing down all associated Procedure Action Diagrams (PrADs).

Work Attribute Set

1. Scope on all work attribute sets you want to add to your procedure. Once the work attribute is referenced in an information view, scoping on the work attribute is unnecessary.
2. Do not scope on the system-supplied work set because it is automatically included.

Common Action Blocks

1. Always scope on all common action blocks USEd by the PrAD. (Full expansion ensures that the common action blocks USEd by the common action blocks are included.)
2. If you must change a common action block, try Modify protection first. Use Delete protection only if you delete views from the common action block.

   Note: The Unit Test subset type bring in all necessary data tables for generating code and RI triggers.

Definition of Detail

As used in the subsetting tables, the verb detail means to add or change the properties to an object. For example, to detail an entity type, you can perform such tasks as add or change the description of the entity type or add or change attributes and identifiers.

Detailed Subset Task Definitions

Detailed subset task definitions are listed under Planning, Analysis, Design, Internal Design, Unit Test, and System Test.

Planning and Analysis Subset

Subject Area

Design Phase Tasks	P	E	Scoping Objects	Note
Create a subject area.	A	S	Parent subject area	
Create a subject area and move existing entity types into it.	M	D	Entity types to move	
Detail a subject area or any of its components and entity types.	M	S	Subject area	
Place unplaced subject area or any of its components.	M	S	Subject area	1
Delete components of a subject area.	D	D	Subject area	2, 3
Delete a subject area.	D	D	Subject area	2, 3

Entity Types

Design Phase Tasks	P	E	Scoping Objects	Note
Add an entity type to the root subject area or any other subject area.	A	S	Subject area or	4, 5
	A	S	Any entity type in it	
Add an entity type to the root subject area and join it to an entity type in another subject area.	M	S	Entity type	
Add an entity type to a subject area and join it to an entity type in another subject area.	M	S	Entity type	
	A	S	Plus subject area for new entity type	
Add an entity type to a subject area and join it to an entity type in the same subject area.	M	S	Entity type	
Move entity types to an existing subject area.	M	S	Entity type	
	A	S	Plus subject area	
Detail an entity type or any of its attributes and components.	M	S	Entity type	
Place unplaced entity type.	M	S	Entity type	
Delete components and attributes of an entity type.	D	D	Entity type	2,3
Delete an entity Type	D	D	Entity type plus related entity types	2,3

Attributes

Design Phase Tasks	P	E	Scoping Objects	Note
Add an attribute.	M	S	Entity type	
Transfer (move from one entity type to another	D	D	Source entity type	2
Copy from one entity type to another	A	D	Source entity type	
	M	D	Entity type	
Detail an attribute property.	D	D	Source entity type	2
Delete an attribute	D	D	Entity type	2

Relationships

Design Phase Tasks	P	E	Scoping Objects	Note
Add a relationship.	M	S	Both entity types	
Detail a relationship.	M	S	Both entity types	
Delete a relationship.	D	D	Either entity type	2
Transfer a relationship	D	S	Two related entity types	2
	D	S	Plus entity type to transfer relationship to	2

Entity Subtypes

Design Phase Tasks	P	E	Scoping Objects	Note
1. Promote subtype to entity type.	D	D	Parent entity type	
2. Change subtype parent.	D	D	Old parent entity	
	A	S	New parent entity	
3. Add a new subtype to an existing entity type.	M	S	Parent entity type	
4. Detail a partitioning.	M	S	Parent entity type	
5. Transfer a relationship.	D	S	Two related entity types	
	D	S	Plus entity type to transfer relationship to	
6. Delete a subtype.	D	D	Parent entity type	
7. Detail a subtype.	M	S	Parent entity type	
8. Place unplaced subtype.	M	S	Parent entity type	

Notes:

- You want to include more of the data model (by scoping on other subject areas with Access, Short).
- An attribute or relationship that is referenced in a view of an entity type cannot be deleted. The Objects Preventing Delete report identifies scoping objects define a subset that contains all references.
- An entity type that is referenced in a view or in expected effects cannot be deleted.

- You cannot add any new objects to a scoping object assigned Read protection.
- You can add new items to the root subject area from this type of subset definition, because the root subset area is present with Access protection.

Note: All entity types must be free of Consistency Check errors before Transformation can succeed.

Data Model

Design Phase Tasks	P	E	Scoping Objects	Note
Transform the entire data model (first time or subsequently).	M	D	Root subject area	1
Intelligent retransformation of the entire data model.	M	D	Root subject area	
Change data model and perform Intelligent Retransformation or implement.	M	D	Root subject area	

Entity Types

Design Phase Tasks	P	E	Scoping Objects	Note
Add an entity type and transform it.	M	S	Associated entity type	2, 3

Attributes

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add an attribute and transform it.	M	S	Entity type	
2.Delete an attribute.	D	D	Entity type	2
3.Detail theproperty of an attribute.	D	D	Entity type	2, 4
4.Copy an attribute from one entity type to another.	A M	S S	Source entity type Target entity type	2
5.Move an attribute from one entity type to another	D M	D S	Source entity type Target entity type	2

Notes:

- The subset task recommendations in Data Modeling Before Transformation allow the transformation. No special modifications are required. Expansion of the root subject area automatically provides the database with Delete protection, which enables transformation.
- Scoping an entity type with Modify protection includes its associated data table with Delete protection and the database for that data table with Delete protection also.
- If the entity type you are adding is a stand-alone, scope any data table of any entity type, and specify Modify protection.
- An attribute or relationship that is referenced in a view cannot be deleted. The Objects Preventing Delete report identifies scoping objects define a subset that contains all references.

Root Function

Design Phase Tasks	P	E	Scoping Objects	Note
Create an AHD root function	A	S	Any entity type	

Activity (Function or Process)

Design Phase Tasks	P	E	Scoping Objects	Note
Add an activity.	M	D	Intended parent	1
Detail an activity.	M	D	Target activity or	
	M	D	Any activity in the parentage of target activity	
Perform Process Synthesis on a non-elementary process.	M	D	Process	2
Transfer activity to a different parent.	D	D	Current parent	3
	M	D	Plus new parent	
Change function to process or process to function	M	D	Parent activity	4
	D	D	Child activity	

Design Phase Tasks	P	E	Scoping Objects	Note
Delete an activity and its subordinates-No subordinates are elementary processes with PADs USEd in other action diagrams.	D	D	Target activity	3
	M	D	Plus parent activity	
			or	
	D	D	Any activity in parentage of target activity	
Delete an elementary process that has been USEd in other action diagrams.	D	D	Process	5
	M	D	Plus parent process	
	M	D	Plus USEing CABs	
	M	D	Plus USEing process	
	M	D	Plus USEing procedure steps	

Notes:

- Scoping an activity automatically brings its parents into the subset with Access authority. Subordinates have the same protection as the scoped activity.
- Process Synthesis can automatically add processes, expected effects, and PADs with stereotypically action diagram statements.
- If you scope on a parent activity with Delete, the parent activity is downgraded to Modify. However, you still receive Delete on the target activity and its subordinates and are able to delete them or transfer them to a different parent.
- To change a function to a process, all lower-level activities for the function must be processes. Therefore, to change a function to a process, all grandchildren must first be processes.
- To remove a PAD, first remove all USE statements for the PAD from any PrADS and CABs.

Entity Life Cycle Diagram

The following table shows the entity life cycle diagram (ELCD).

Design Phase Tasks	P	E	Scoping Objects	Note
Add and join entity states.	M	S	Parent entity type	
Add a process to a transition.	M	D	Process	
Detail a process.	M	D	Process	

Expected Effects for Functions (Planning)

The following table shows the expected effects for functions (planning).

Design Phase Tasks	P	E	Scoping Objects	Note
Add an existing entity type and its expected effect to a function.	A M M	S D D	Entity type Plus function or Business Function/Entity Type matrix	1
Remove an entity type from a function or change its expected effects for a function.	M M	D D	Function or Business Function/Entity Type matrix	2

Notes:

- Scoping an activity automatically brings its parents into the subset with Access authority. Subordinates have the same protection as the scoped activity.
- In matrices, functions receive Modify protection, and entity types receive default (not short) expansion. Therefore, these matrices can be large.

Expected Effects for Process (Analysis)

The following table shows expected effects for process (analysis).

Design Phase Tasks	P	E	Scoping Objects	Note
Add an existing entity type and its expected effect to a process.	A M M	S D D	Entity type Plus process or ELEMENTARY Process/Entity Type matrix	1 2
Remove an entity type from a process or change its expected effects for a process.	M M	D D	Process or BUSINESS Elementary Process/Entity Type matrix	2

Notes:

- After an entity type appears in the expected effects or the view of an activity, it is automatically included in every subset that includes that activity.
- In matrices, functions receive Modify protection, and entity types receive default (not short) expansion. Therefore, these matrices can be large.

Activity Dependency (ADD)

The following table shows Activity Dependency (ADD).

Activities (Functions/Process), Events, or External Objects

Design Phase Tasks	P	E	Scoping Objects	Note
Add an activity, event, or external object.	M	D	Parent activity	1
Detail an activity, event or external object.	M	D	Target activity	
Delete an activity, event, or external object.	M D	F D	Target activity Plus parent activity	
Specify information that flows between a process and an external object.	M	D	Process	
Change parent or an activity.	D M	D D	Current parent Plus a new parent	

Dependencies

Design Phase Tasks	P	E	Scoping Objects	Note
1.Create a dependency.	M	D	Both activities	
2.Detail a dependency.	M	D	Both activities	
3.Delete a dependency.	D	D	Both activities	

Note: Scoping an activity automatically brings its parents into the subset with Access protection. Subordinates have the same protection as the scoped activity.

Organizational Hierarchy (OHD)

The following table shows organizational hierarchy (OHD).

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add an organization.	A	D	Organization unit	1
2.Detail an organization.	A	D	Organization unit	
3.Delete an organization.	D	D	Organization unit	1

Note: Scoping an organization unit brings in the entire hierarchy or organization units, parents, and subordinates.

System-Supplied Matrices

The following table shows system-supplied matrices.

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add objects to a matrix.	M	D	Matrix	2
2.Add cell values.	M	D	Matrix	
3.Change cell values.	M	D	Matrix	
4.Delete cell values.	D	D	Matrix	
5.Describe matrix (text description).	M	D	Matrix	
6.Detail objects used in rows or columns.	M	D	Matrix	
7.Delete objects from a matrix.	D	D	Matrix	3
8.Perform cluster function on Business Function/Entity Type matrix	D	D	Business Function/ Business Function Matrix	
	M	D	Entity Type/ Entity Type matrix	
	M	D	Business Function/Entity Type matrix	
9.Perform cluster function on Elementary Process/Entity Type matrix.	D	D	Elementary Process/Elementary Process matrix	
	M	D	Elementary Process/ Entity Type matrix	

Notes:

- When you scope a system-supplied matrix with Modify, you create a large subset.
- Required only when adding new objects to an existing matrix.
- Only the usage of objects in the matrix is deleted.

User-Defined Matrices

The following table shows user-defined matrices.

Matrix

Design Phase Tasks	P	E	Scoping Objects	Note
1.Create a user-defined matrix.	A A	D D	Object classes for X axis Object classes for Y axis	
2.Detail names or descriptions for objects on X or Y axis.	M M	D D	Matrix or Target class object	
3.Describe matrix (text description).	M	D	Matrix	
4.Add objects to an existing matrix.	M	D	Matrix	
5.Delete objects from a matrix.	D	D	Matrix	
6.Delete a user-defined matrix.	D	D	Matrix	

Cell Values

Design Phase Tasks	P	E	Scoping Objects	Note
1.Create a new object class.	A A	D D	Any user object or User object class	
2.Modify a user object class.	M	D	User object class	
3.Delete a user-defined class.	D M	D D	User object class Plus every matrix using this class on an X or Y axis	

UserDefined Objects

Design Phase Tasks	P	E	Scoping Objects	Note
1.Create a new object.	A	D	User object class or any object in it or Any object in it	
2.Modify a user-defined object	M	D	User-defined object	
3.Delete objects from a class.	D	D	Target user object or	
	D	D	Its object class	

Process Action Diagram (PAD)

The following tables show process action diagram (PAD).

Process Action Diagram

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add a PAD.	M	D	Process	1
2.Add, change, or delete statements in a PAD.	M	D	Process	1, 2
3.Delete a PAD.	M	D	Process	2

Action Block Synthesis

Design Phase Tasks	P	E	Scoping Objects	Note
1.Perform Action Block Synthesis on an action diagram with no existing views.	M	D		
2.Perform Action Block Synthesis on an action diagram with existing views but not USEd in any other action diagram.	D	D	Process	3
	M	D	Plus parent activity	

Design Phase Tasks	P	E	Scoping Objects	Note
3.Perform Action Block Synthesis on action diagram with existing views and USEd in another action diagram.	D	D	Process	3
	M	D	Plus common action block	
	M	D	Plus USEing CABs	
			Plus USEing Process	
	M	D	Plus USEing procedure steps	
	M	D		

View Maintenance Entity View

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add an entity view.	M	D	Process	4
	A	S	Plus entity type	
2.Change an entity view's name or properties.	M	D	Process	4
3.Change a referenced entity type of subtype.	M	D	Entity type	
4.Delete an entity view, or an attribute in the view.	D	D	Process	5, 6
	M	A	Plus all processes that use it	

Work View

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add a work view.	M	D	Process	
2.Change a local view.	M	D	Process	
3.Delete a local view.	D	D	Process	

Notes:

- The only way to add a PAD is to add a process to the Activity Hierarchy and mark it as elementary
- The only way to delete a PAD is to delete its associated process. Refer to the tasks for Activity Hierarchy.

- To create new information views and action diagrams statements, Action Block Synthesis must first delete any existing views or statements. Therefore, Delete protection is required.
- After defining expected effects, you do not need to scope the entity type.
- To delete a view, your subset must contain all references to that view (all objects that use it). The Model Cross-Reference reports help you define a subset that contains all references to the object.
- If the view is view-matched to another action diagram, you need full expansion.

Business System Definition (BSD)

The following table shows business system definition (DBS).

Design Phase Tasks	P	E	Scoping Objects	Note
1. Add a new dialect language for system defaults.	A	S	Business system	
2. Detail a business system or any of its defaults.	M	S	Business system	1,2
3. Delete a business system.	D	F	Business system	3

Notes:

- The short expansion brings in only system default objects.
- This task includes adding bilingual text to current system defaults.
- If a business system includes built procedures (with PrADs and built screens), use Host Encyclopedia Delete or Rename Object instead of a subset. Otherwise the subset is large.

Analysis and Design Subset

For view maintenance tasks, see:

- Activity Dependency
- Common Action Blocks
- Process Action Diagram
- Procedure Action Diagram

Design Subset

Business System Defaults

The following tables show business system defaults:

System-wide Commands

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add a command or synonym.	A	S	Business system	1,2
2.Change a command.	M	S	Business system	
3.Add, change, or delete bilingual command synonym.	M	S	Business system	1, 3
4.Delete a command.	D M	S D	Business system Plus all processes, procedure steps or CABs that use the command.	4

System-wide Function Keys

Design Phase Tasks	P	E	Scoping Objects	Note
1.Set command for PF Key.	M	S	Business system	1
2.Change command for PF Key.	M	S	Business system	
3.Add, change bilingual text for PF key.	M	S	Business system	
4.Assign <None> to PF Key.	M	S	Business system	

System-wide or Global Exit States

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add an exit state to the list of exit states.	A	D	Exit state	1,2
2.Add, change, or delete message (standard or bilingual) and maintain exit state properties.	M	D	Exit state	

Design Phase Tasks	P	E	Scoping Objects	Note
3.Delete an exit state.	D	D	exit state	5
	M	D	Plus all processes, procedure steps or CABs that use the exit state	

Default Video Attributes

Design Phase Tasks	P	E	Scoping Objects	Note
Change the default video display characteristics for fields, prompts, literals or error messages.	M	S	Business system	3

System-wide Delimiters

Design Phase Tasks	P	E	Scoping Objects	Note
Change the default delimiter.	M	S	Business system	

System-wide Edit Patterns

Design Phase Tasks	P	E	Scoping Objects	Note
1. Add an edit pattern.	A	S	Business system	2
2. Change an edit pattern not copied across business systems.	M	S	Business system	
3. Add, change or delete a language-specific pattern.	M	S	Business system	
4. Copy default edit pattern to another business system for use on a window	A	S	Business system that contains edit pattern	6, 7
	M	D	Plus procedure step that contains the window	
5. Change a copied edit pattern.	M	D	Business system that contains edit pattern	7
	M	D	Plus procedure steps that contain the windows using edit pattern	
6. Delete an edit pattern.	D	F	Business system	8

Design Phase Tasks	P	E	Scoping Objects	Note
7. Delete an edit pattern that is used on a window copied across business systems.	D A	F D	Business system Plus procedure step that contains the window	7

Scroll Amount Values

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add bilingual text for the scroll amount values.	M R	D S	Scroll amount values Any business system	9
2.Change or delete bilingual text for scroll amounts.	M R	D S	Scroll amount value Any business system	
3.Delete bilingual text for scroll amounts.	M R	D S	Scroll amount value Any business system	
4.Change language or DBMS (Business System Environment Parameters)	M	S	Business system	10

Custom Video Properties

The following table describes using custom video properties of a business system:

Step	Description	P	E	Objects	Notes
1.	Add a custom video property	A	S	Business system	2
2.	Change a custom video property	M	S	Business system	
3.	Select a custom video property for use with business system default video properties	M	S	Business system	
4.	Select a custom video property for use with a window, dialog box, or window control	M	D	Procedure Step	
5.	Delete a custom video property	D M	S D	Business system Plus all windows, dialog boxes that use the custom video property	11

Notes:

- The short expansion of a business system brings in the business system defaults, all bilingual text for system default, and the unexpanded technical system.
- Scoping any procedure, procedure step, screen, or template includes the business system defaults with Access protection.
- See also Scroll Amount Values in Business System Defaults.
- To delete a command, your subset must contain all references to the command. The Objects Preventing Delete report identifies scoping objects define a subset that contains all references. For more information about deletions, see Plan a Subset to Delete Views.
- To delete an exit state, your subset must contain all references to the exit state. The Objects Preventing Delete report identifies scoping objects define a subset that contains all references.
- Edit the patterns cannot be copied for screens or templates.
- The Host Encyclopedia Scoping Object Where Used report can be used to identify scoping objects for this task.
- Full expansion includes all uses of the edit pattern within the owning business system.
- Scroll the amounts do not belong to any dialect language and must be scoped directly for modification. Your subset must contain a business system to access them. See also Bilingual Support.
- For example, if the production model use COBOL and DB2, and you are testing on the workstation, you can use a subset to set or override the Construction environment parameters (COBOL and DB2). To override the environment parameters, set Construction generation defaults to C and check the box that is entitled, Override Bus Sys Target Environment with previous defaults.
- To delete a custom video property, your subset must contain all references to the custom video property.

Bilingual Support

The following tables show bilingual support:

Dialect Definition

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add a new dialect language.	R	S	Business system	1

Design Phase Tasks	P	E	Scoping Objects	Note
2.Add a new dialect language and scroll amount text in that language.	R	S	Business system	
	M	D	Plus scroll amounts	
3.Detail a dialect language.	M	D	Dialect	1
	R	S	Plus any business system	
4.Rename a dialect language	M	D	Dialect	
	R	S	Plus any business system	
5.Delete an unused dialect (one with no bilingual text).	D	D	Dialect	
	R	S	Plus any business system	
6.Delete a dialect and all bilingual text for that language.	D	D	Dialect	2
	M	F	Plus all BUSINESS systems whose defaults use that language	
	M	D	Plus global exit states having messages in that language	
			Plus entities for prompts	
	M	D	Plus scroll amount values	
	M	D	Plus system attribute set	
	M	D	Procedure steps with windows	
	M	D	that use text in that language	

Bilingual Text, Business System Defaults

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add bilingual text to business system defaults	M	S	Business system	
2.Change bilingual text for business system defaults.	M	S	Business system	
3.Delete bilingual text for business system defaults.	M	D	Business system	3

Bilingual Text, Scroll Amount Values

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add bilingual text for scroll amount value.	M	D	Business system	

Design Phase Tasks	P	E	Scoping Objects	Note
2.Change bilingual text for scroll amount value.	M	D	Scroll amount	
3.Delete bilingual text for scroll amount value.	M	D	Scroll amount	3

Bilingual Text, System-wide and Global

Design Phase Tasks	P	E	Scoping Objects	Note
Exit States 1.Add bilingual message for a global exit state.	M R	D S	Exit state Plus any business system	
2.Change bilingual message for a global exit state.	M R	D S	Exit state Plus any business system	
3.Delete bilingual message for a global exit state.	M R	D S	Exit state Plus any business system	3

Bilingual Text, Prompts and Literals

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add bilingual prompt to a screen or template.	M M	S S	Screen or Template	
2.Change bilingual prompt on a screen or template.	M M	S S	Screen or Template	
3.Delete bilingual prompt from screen or template.	M M	D D	Screen or Template	
4.Delete bilingual prompt from screen or template using system attribute.	M M	D D	Screen or Template	

Notes:

- Dialect functions are accessible only through the Design toolset. To access them, your subset must contain a business system. On the CA Gen Toolset, the Dialects function appears under the Design, Dialect Definition option.
- The Objects Preventing Delete report identify all objects that are needed for this subset.
- The Objects Preventing Delete report can identify objects using a specified dialect.
- A common action block is any action diagram that is **not** a Process Action Diagram (PAD) or a Procedure Action Diagram (PrAD). See also the Process Action Diagram and the Procedure Action Diagram section.

Common Action Block (CAB)

The following tables show common action block (CAB):

Common Action Block

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add a common action block.	A	D	Process	1
			or	
	A	D	Procedure step	
			or	
	A	D	Common action block	
			or	
	A	S	Business system	
2.Add a common action block and USE it in a Process Action Diagram.	M	D	Process	
3.Add a common action block and USE it in a Procedure Action Diagram	M	D	Procedure step	
4.Add a common action block and USE it in another existing action block.	M	D	Common action block	
5.Add, change, or delete statements in a common action block.	M	D	Common action block	

Design Phase Tasks	P	E	Scoping Objects	Note
6.Delete a common action block.	D	D	Common action block	2
	M	D	Plus USEing CABs	
	M	D	Plus USEing process	
	M	D	Plus USEing procedure steps	

Promote CAB to a Process

Design Phase Tasks	P	E	Scoping Objects	Note
Promote a common action block to a process.			Common action block to be promoted	
			Plus USEing CABs	
			Plus process to be promoted into	
	M	D		

Action Block Synthesis

Design Phase Tasks	P	E	Scoping Objects	Note
1.Perform Action Block Synthesis on an action diagram with no existing views.	M	D	Common action block	
2.Perform Action Block Synthesis on action diagram with existing views but not USEd in any other action diagram.	D	D	Common action block	3
3.Perform Action Block Synthesis on action diagram with existing views and USEd in another action diagram.	D	D	Common action block	3
	M	D	Plus USEing CABs	
	M	D	Plus USEing process	
	M	D	Plus USEing procedure steps	

View Maintenance Entity View

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add an entity view	M	D S	Action block Plus entity type	4
2.Change an entity view's name or properties.	M	D	Action block	4
3.Change a referenced entity type or subtype	M	D	Entity type	
4.Delete an entity view, or an attribute in the view.	D M	D A	Procedure step Plus all processes or procedure steps that use it	5, 6

Work View

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add a work view.	M	D	Action block	
2.Change a work view.	M	D	Action block	
3.Delete a work view.	D	D	Process	5, 6
			or	
	D	D	Procedure step	

Local View

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add a local view.	M	D	Action block	
2.Change a local view.	M	D	Action block	
3.Delete a local view.	D	D	Process	
			or	
	D	D	Procedure step	

Notes:

- Having Access on almost any object in your subset lets you add a common action block.
- To remove a CAB, first remove all USE statements for the CAB from all PADs, PrADs, and CABs.
- For Action Block Synthesis to create new information views and action diagram statements, it must first delete any existing views or statements. Therefore, Delete protection is required.
- After defining expected effects, you do not need to scope the entity type
- To delete a view, your subset must contain all references to that view (all objects that use it). The Objects Preventing Delete report helps you define a subset that contains all references to the object.
- If the view is view-matched to another action diagram, you need full expansion.

Dialog Flow (DLG)

The following tables show dialog flow (DLG):

Procedure

Design Phase Tasks	P	E	Scoping Objects	Note
1. Add a procedure.	A	S	Business system	1, 2
			or	
	A	D	Any procedure	
			or	
	A	D	Any procedure step	
2. Detail a procedure.	M	D	Procedure	
3. Delete a procedure.	M	F	Procedure to be deleted	3
	M	D	Plus any procedure or procedure step that flows to the scoped procedure	
	D	S	Plus any load modules containing its procedure steps	

Procedure Step

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add a new procedure step to an existing procedure.	A	S	Procedure	
2.Detail a procedure step, its PrAD, screen/window.	M	D	Procedure step	
3.Copy a procedure step.	A	D	Procedure step	4
4.Delete a procedure step.	D M A	D D D	Procedure step to be deleted Plus any procedure or procedure step that flows to the procedure step being deleted Plus any procedure step that the procedure step being deleted flows to if views are being sent or returned on the flow	3

Dialog Flow

Design Phase Tasks	P	E	Scoping Objects	Note
1. Add detail, or change a flow between procedures. desc flow type flows on properties data passed match/unmatch	M M	S S	Source procedure step Target procedure step	5
2.Delete a flow.	M M	S S	Source procedure step Target procedure step	
3.Add, detail, and change a flow to or from another business system	M M	S S	Source procedure step Target procedure step	

Design Phase Tasks	P	E	Scoping Objects	Note
4.Delete a flow to or from another business system.	M M	S S	Source procedure step Target procedure step	

Notes:

- The business system scoping object has a special short (S) expansion that brings in only the system-wide default objects.
- Scoping any procedure or step automatically includes the business system defaults with Access protection.
- The Objects Preventing Delete Report shows all objects that would prevent deletion of the procedure step. This report is used to help define a subset definition containing all the objects delete this scoping object.
- You must have at least Access to the procedure step to copy it.
- To change the dialog flows and what exit states initiate them, check out the procedure step from which the flows begin with at least update authority.

Procedure Action Diagram

The following tables show procedure action diagram:

Procedure Action Diagram

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add, change or delete statements in a Procedure Action Diagram.	M	D	Procedure step	1,2,3

Procedure Synthesis

Design Phase Tasks	P	E	Scoping Objects	Note
1.In business systems with no existing views or action diagram statements.	M	D	Procedure step	
2.For the first procedure step.	M M	S D	Business system Plus work attribute set	

Design Phase Tasks	P	E	Scoping Objects	Note
3.For subsequent Procedure Synthesis (after updating and checking in model).	M	D	Procedure set	

View Maintenance Entity View

Design Phase Tasks	P	M	Scoping Objects	Note
1. Add an entity view.	M	D	Procedure step	5
	A	S	Plus entity type	
2. Change an entity view's name or properties	M	D	Procedure step	5
3. Change a referenced entity type or subtype.	M	D	Entity type	
4. Delete an entity view, or an attribute in the view.	D	D	Procedure step	6, 7
	M	A	Plus all procedure steps that use it	

Work View

Design Phase Tasks	P	E	Scoping Objects	Note
1. Add a work view.	M	D	Procedure step	
2. Change a work view.	M	D	Procedure step	
3. Delete a work view.	D	D	Procedure step	5, 7

Local View

Design Phase Tasks	P	E	Scoping Objects	Note
1. Add a local view.	M	D	Procedure step	
2. Change a work view.	M	D	Procedure step	
3. Delete a work view.	D	D	Procedure step	

See also the Process Action Diagram and the Common Action Blocks sections.

Notes:

- You cannot add or delete a Procedure Action Diagram except by adding or deleting its procedure step.
- To build action diagram statements that reference objects that you have not referenced before, include those objects in your subset with at least Access protection. If the objects do not exist yet, be sure to define a subset you can use to add them. For example, you include entity types, work sets, exit states of another business system, common action blocks, and PADs.
- The expansion of a procedure brings in the action diagrams of each procedure step in the procedure with full expansion.
- The default expansion of every subset provides the system-supplied work attribute set. The default expansion also provides what you need to add other work sets (the root subject area with Access protection).
- After defining expected effects, you do not need to scope the entity type.
- To delete a view, your subset must contain all references to that view (all objects that use it). The Objects Preventing Delete Report identifies scoping objects define a subset that contains all references.
- If the view is view-matched to another action diagram, you need full expansion.

Screen Design

The following tables show screen design:

Screen

Design Phase Tasks	P	E	Scoping Objects	Note
1. Create screen fields using Screen Layout Builder.	M	S	Procedure step	1,2
2. Recreate a screen using the Screen Layout Builder.	M M	S D	Procedure step Plus any procedure or step that flows to it	2

Fields

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add a literal or special field.	M M	D S	Screen or Its procedure step	

Design Phase Tasks	P	E	Scoping Objects	Note
2.Add a field and an info view.	M	D	Screen	
			or	
	M	S	Its procedure step	
			or	
	A	S	Entity type of work set	
			or	
	A	D	work set	
3.Change or delete a field on a screen.	M	D	Screen	
			or	
	M	S	Its procedure step	

PF Keys and Commands

Design Phase Tasks	P	E	Scoping Objects	Note
Create or change local PF keys and commands assigned to them.	M	S	Procedure step	

Template Design

Design Phase Tasks	P	E	Scoping Objects	Note
1.Modify template and any screens using it	M	D	Template	
2.Modify screen items in a template.	M	S	Template	3,4,5
3. Delete a template and replace it with a new one for all screens that use it.	D M	D D	Template Plus system attribute set if they have bilingual prompt for system attributes	

Bilingual Text

Design Phase Tasks	P	E	Scoping Objects	Note
1. Add or change bilingual text for screen elements.	M M	D S	Screen or Its procedure step	

Design Phase Tasks	P	E	Scoping Objects	Note
2. Delete the contents of a screen that has bilingual prompts for system attributes.	D	D	Screen	
			or	
	D	S	Its procedure step	
3. Change or remove bilingual prompt literal for system attributes on a screen.	M	D	Procedure step	
			or	
	M	D	Screen	
4. Add or change bilingual text/ prompts for system attributes used in a template.	M	S	Template	4
5. Add or change bilingual text for a prompt.	M	D	Screen	
			or	
	M	D	Template	
6. Delete bilingual prompts for system attributes used in a template.	M	D	Template	4

Prompt

Design Phase Tasks	P	E	Scoping Objects	Note
1. Add or change bilingual text for a prompt.	M	D	Procedure step	
			or	
	M	D	Screen	
			or	
	M	D	Template	
2. Delete a usage of a prompt.	D	D	Procedure step	
3. Delete the last usage of a prompt.	D	D	Procedure step	6
4. Delete the prompt itself.	D	D	Procedure step	7
	D	S	Plus entity type	
5. Add, change, or delete prompt literal and special attribute of a screen.	D	D	Screen	
			or	
	D	D	Its procedure step	

Notes:

- You cannot add or delete a Procedure Action Diagram except by adding or deleting its procedure step.
- The layout builder does NOT support bilingual tasks.

- Use short expansion if only the template is changed.
- Use default expansion to change the template and all screens that use it.
- The short expansion of a template excludes such items as screens, procedure steps, action blocks, views, and entities. Only screen items on the template itself can be changed.
- Deleting the last usage of a prompt does **not** delete the prompt itself.
- When you delete a prompt from the last screen that uses it, you must have Delete protection on the entity type that owns the prompt.

Window Design

The following table shows window design:

Design Phase Tasks	P	E	Scoping Objects	Note
1. Create a window for a procedure step.	M	D	Procedure step	1
2. Change the contents of a window.	M	D	Procedure step	1, 2
3. Copy a default edit pattern from another business system for use on a window.	M A	D S	Procedure step Plus business system	1, 2
4. Delete a field on a window	M	D	Procedure step	2, 3
5. Delete a window.	D	D	Procedure step	2, 3

Notes:

- The contents of a window include all prompts and variables. A procedure step can have multiple windows.
- Bilingual test support for windows is not provided at present.
- A window using a default edit pattern copied from another business system cannot delete that edit pattern. Only its usage is removed.

Work set List

The following table shows work set list:

Design Phase Tasks	P	E	Scoping Objects	Note
Create a work attribute set.	A	S	Root subject area or other work attribute set	1
	A	D		

Note: The root subject area for the model is included with Access protection as part of the default set of objects.

Interface

The following table shows using Interface:

Design Phase Tasks	P	E	Scoping Objects	Note
1. Create Interface	M	D	Server Manager	1
			or	
	A	D	Server Manager	
	M	D	Procedure Steps	
			or	
	M	D	Procedure Steps with subset type set to unit or system	
2. Update Interface	M	D	Server Manager	1
			or	
	A	D	Server Manager	
	M	D	Procedure Steps	
			or	
	M	D	Procedure Steps with subset type set to unit or system	

Note: Server Manager and its procedures steps must exist in subset in order to create interfaces. Procedure Step must have modify protection.

Custom Proxy

The following table shows using Custom Proxy:

Design Phase Tasks	P	E	Scoping Objects	Note
1. Create Custom Proxy Definition and add Proxy	M	D	Server Manager	1
			or	
	A	D	Server Manager	
	M	D	Procedure Steps	

Design Phase Tasks	P	E	Scoping Objects	Note
			or	
	M	D	Procedure Steps with subset type set to unit or system	
2. Update Custom Proxy and existing Proxies	M	D	Custom Proxy	
3. Update Custom Proxy and add new Proxy	M	D	Custom Proxy	1
			and	
	M	D	Server Manager	
			or	
	A	D	Server Manager	
	M	D	Procedure Steps	
			or	
	M	D	Procedure Steps with subset type set to unit or system	
4. Generate Custom Proxy	A	D	Custom Proxy	2
			and	
	A	D	Server Manager	
			or	
	M	D	Procedure Steps	
			or	
	A	D	Custom Proxy with subset type set to unit or system	

Note: Server Manager and its procedure steps must exist in subset to select interfaces for custom proxy and generate custom proxy. Procedure Step must have modify protection.

Web Service Definition

The following table shows using Web Service Definition:

Design Phase Tasks	P	E	Scoping Objects	Note
1. Create Web Service Definition and add Web Operation	M	D	Server Manager	1
2. Update Web Service Definition and existing Web Operations	M	D	Web Service Definition	
3. Update Web Service Definition and add new Web Operation	M	D	Web Service Definition	1
			and	
	M	D	Server Manager	
			or	
	A	D	Server Manager	
	M	D	Procedure Steps	
			or	
	M	D	Procedure Steps with subset type set to unit or system	

Note: Server Manager and its procedures steps must exist in subset to select interfaces for web operation. Procedure Step must have modify protection.

Configuration Instances

The following table shows using Configuration Instances:

Design Phase Tasks	P	E	Scoping Objects	Note
1. Add Configuration Instance	A	D	All target objects: Load Modules, Procedure Steps, Action Blocks, Database, Records, and/or Storage Groups.	1, 2, 3, 4, 8, 9
2. Delete Configuration Instance	D	D	Configuration Instance	7, 8
3. Rename Configuration Instance	D	D	Configuration Instance	7, 8

Design Phase Tasks	P	E	Scoping Objects	Note
4. Modify Configuration Instance including select/unselect target objects, update generation and trace options	M	D	Configuration Instance	5, 7, 8
	A	D	Load Modules (optional)	
5. Generate Configuration Instance without making any changes	R	F	Configuration Instance	5, 6
	R	F	Load Modules (optional)	
6. Generate Configuration Instance after making changes to generation options or trace options	M	F	Configuration Instance	5, 6, 7, 8
	A	F	Load Modules (optional)	
7. To change Configuration Instance from model based to file based **Note:** Deletes the objects from the model.	D	D	Configuration Instance	7, 8
8. To change Configuration Instance from file that is based to model based **Note:** Creates the objects in the model.	A	D	All target objects	7, 8

Notes:

- Subset must include Load Modules to select objects under the Model Object Selection tab.
- Subset must include Databases to select objects under the Database tab.
- Model must not be read-only.
- If the subset type is Unit or System, must scope Database with Access.
- Required to explicitly scope load modules if Configuration Instance currently includes all load modules within the business system. Otherwise the model object selection is empty.
- Recommend the subset type set to System for generation
- The target objects must have at least Access protection. If a target object is downgraded to Read, user cannot perform this function.
- If a Configuration instance includes (or would like to include) an index, the parent Table for the index must be included in the subset with Modify or Delete protection to ensure that the Index gets the Access protection needed. Selecting a database for inclusion in a Configuration instance require that all Tables associated with that Database have Modify or Delete protection so that the associated indexes are included in the instance.
- When a configuration includes the storage group for a database, it is recommended that the database is also selected in the configuration.

Internal Design Subset

Note: To perform all Data Structure and Data Store tasks, refer to General Subset Definitions and scope on the root subject area at Modify and Default.

Transformation

The following table shows transformation:

Design Phase Tasks	P	E	Scoping Objects	Note
1. Transform entire data model (first time or subsequently).	M M D	D D D	Root subject area Tech design default Databases attached to tech design default	1, 2, 3
2. Intelligent retransformation of the entire data model.	M M D	D D D	Root subject area Tech design default Databases attached to tech design default	1, 2, 3

Design Phase Tasks	P	E	Scoping Objects	Note
3. Change data model and perform Intelligent Retransformation or implement.	M	D	Root subject area	1, 2, 3
	M	D	Tech design default	
	D	D	Databases attached to tech design default	
4. Change database defaults.	M	D	Database	

Notes:

1. All entity types must be free of Consistency Check errors before Transformation can succeed.
2. In a small model, you can scope the root subject area instead of individual entity types.
3. Scope every entity type that has any relationship to another entity type in the chain of entities to be transformed, or scope the subject areas containing them. If you fail to do this, some of the entity types come down with Access protection and transformation fails.

Data Structures

The following tables show data structure:

Data Tables

Design Phase Tasks	P	E	Scoping Objects	Note
1. Add a new data table not associated with another data table.	M	D	Unimplemented entity type	
	M	D	Any data table	
2. Add a new data table not associated with another data table.	M	D	Unimplemented entity type	1
	M	S	Its associated entity types	
3. Delete a data table not associated with another data table	D	D	Implemented entity type of target data table	1

Design Phase Tasks	P	E	Scoping Objects	Note
4. Delete a data table associated with another data table.	D D M M	D D D S	(a) target data table (b) data tables that have target data table's relationship membership name as their identifier (c) implemented entity types for data tables in (a) & (b). (d) any entity types to which the foreign key column of (a) & (b) point	

Note: 1. An entity type scoped with Modify or Delete protection includes its associated data table at the same level of protection.

Data Columns and Indexes

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add a data column.	M	D	Implemented entity type	1
2.Delete a data column.	D	D	Implemented entity type	1
3.Add an index.	M	D	Data table	
4.Delete an index.	D	D	Implemented entity type	1

Note: 1. An entity type scoped with Modify or Delete protection includes its associated data table at the same level of protection.

Constraints

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add a constraint.	M	D	Both implemented entity types	1
2.Delete a constraint.	D	D	All associated entity types	1

Note: 1. An entity type scoped with Modify or Delete protection includes its associated data table at the same level of protection.

Databases and Tablespaces

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add a database.	M	D	Any data table of an existing database	
2.Delete a database.	D	D	Database, all	
	M	D	Implemented entity types in the database	
3.Add a tablespace to an existing database.	M	D	Any data table in the database	
4.Delete a tablespace.	D	D	All data tables in the tablespace	
	M	D	Any data tables in other table spaces having TO or FROM associations with them	
	M	D	Their implemented types	

Storage Groups

Design Phase Tasks	P	E	Scoping Objects	Note
1. Define a new storage group.	M	D	Any data table or storage group	
2. Create a storage group for an existing database.	M	D	Any data table in the database	
3. Change a storage group for an existing database.	D	D	Any data table in the database	

Design Phase Tasks	P	E	Scoping Objects	Note
4. Detail a storage group.	M	D	Storage group	
5. Delete a storage group.	M	D	Storage group and all data tables in the database contained by the storage group	

Partition

Design Phase Tasks	P	E	Scoping Objects	Note
Add a partition.	M	D	Data table	

Note: An entity type scoped with Modify or Delete protection includes its associated data table at the same level of protection.

Construction Subset

Packaging for Window and Cooperative Processing

The following tables show packaging for window and cooperative processing:

Design Phase Tasks	P	E	Scoping Objects	Note
1. Add new packaging (a new load module).	M	S	All unpackaged procedure steps	1
2. Delete a packaged load module.	D	S	Load module	
3. Delete packaged procedure steps from a load module.	M	S	Load module	2

Modify Existing Packaging

Design Phase Tasks	P	E	Scoping Objects	Note
1.Add a trancode.	A	S	Load module	
2.Add procedure steps.	M	S	Unpackaged procedure steps	
	A	S	Plus load module	
3.Assign existing trancodes.	M	S	Load module	

Design Phase Tasks	P	E	Scoping Objects	Note
4.Modify clear screen and dialog flow trancodes for packaged procedure steps.	M M	S S	Unpackaged procedure steps Plus load module	

Notes:

- You can assign a clear screen and dialog flow trancodes to the newly packaged procedure steps if the trancodes are new objects on the workstation.
- You can delete packaged procedure steps from the load module with this protection if and only if the deletion of the packaged procedure step does not cause trigger deletion of the load module or any trancodes in the load module. Trigger the deletion of the load module occurs only when you are deleting 1) all packaged procedure steps in the load module or 2) the last remaining packaged procedure step. If you need more detailed information about tasks that involve Construction scoping objects, see the task tables for Procedure Step, Database, Action Block, Business System, and Subject Area.

Unit Test

The following table shows unit test:

Unit Test Tasks	P	E	Scoping Objects	Note
1. Generate database, modify and generate code, and generate RI triggers.	M	F	Procedure step	1
2. Generate database, code, and RI triggers.	R	D	Database	1
3. Generate code for single procedure step without changes.	R	F	Procedure step	
4. Change single procedure step's action block and regenerate its code.	M	F	Procedure step	1
5. Change, generate, and install USEd action block.	M R	F F	Action block Plus procedure step	
6. Generate database without changes.	A	D	Database	
7. Generate database with changes (names, defaults).	M M	D D	Root subject area or Database	
8. Generate RI triggers.	A A	D D	Database or Root subject area	

Unit Test Tasks	P	E	Scoping Objects	Note
9. Test action block statements.	R M	D A	Procedure Plus procedure steps	

Note: If a load module exists for this procedure step, it is **not** expanded. To get expansion of the module, specify System Test as the subset type. If this procedure step is part of a load module that contains other procedure steps, the other procedure steps are not included.

System Test

The following table shows system test:

System Test Tasks	P	E	Scoping Objects	Note
1. Generate code for load module without changes.	R	F	Any procedure step packaged in module	
2. Change a load module's procedure steps' action blocks and regenerate their code.	M	F	Procedure step	1
3. Change business system environment properties (for example, language, database, restart capability.	M M	F S	Procedure steps Plus their business system	1
4. Regenerate code with flows outside the load module (to maintain a valid Dialog Manager for the module).	M	F	Procedure step to be changed, regenerated.	1,2
5. Create a remote file (.rmt)	M	D	Procedure	3

Notes:

- The expansion option or protection level of the procedure step does not impact the expansion or protection of its load module. The other components of the load module receive full expansion and Read protection. The scoped procedure step receives the protection and expansion that you specify in the subset definition.
- You do not need to scope any other procedure steps and regenerate triggers since Implementation Logic Usage (IMPUSE) objects are automatically included.

Chapter 7: Expansion Tables

The expansion tables indicate the expansion objects for each type of scoping object, when scoped with Modify protection (the default). The tables are arranged in alphabetical order by type of scoping object.

Use the expansion tables if you cannot find the task that you need in the subsetting tasks appendix.

For the scoping objects, protections and expansions that are required for many specific tasks, see the "Task Tables" appendix in this guide.

The protection that is granted to the objects in a subset differ from that shown in the expansion tables, depending on the selected expansion level. The protection that is granted differ from that requested if the target object is checked out with conflicting protection.

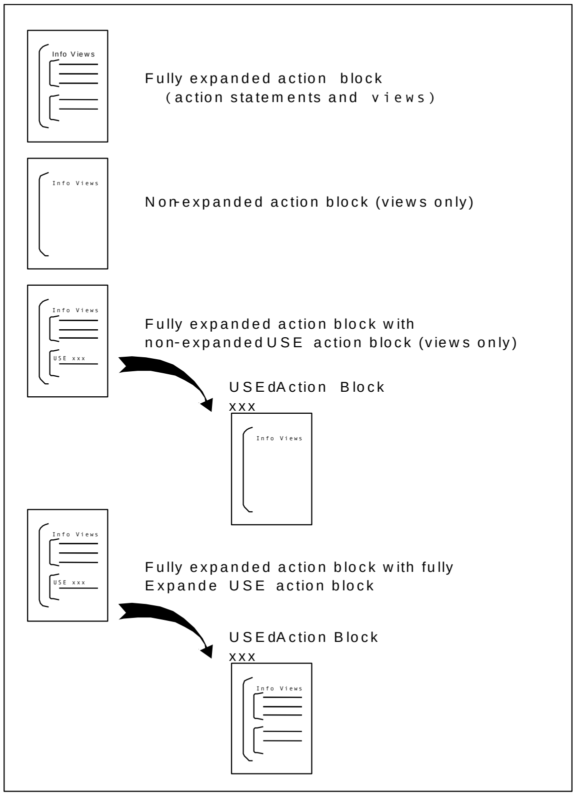
Expansion options include Full, Default, A, B, and Short. Expansion options differ in the number of expansion objects, the default protection granted the expansion objects, and if the expansion object is a scoping object, the subsequent expansion of the expansion object.

Expansion of Scoping Objects

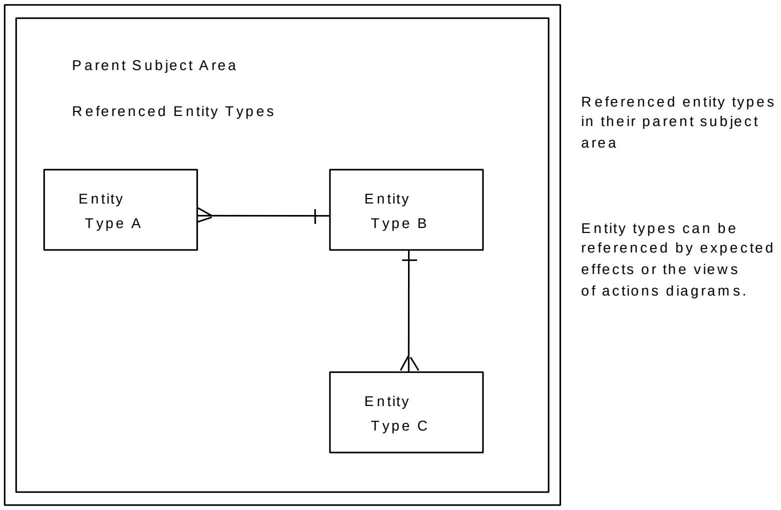
Action Block

To maintain the action statements or views in an action block, using the Action Diagram tool, use the action block scoping object.

To modify the action statements or views when the action block does not belong to a specific procedure step or process, include an action block in the definition of a subset.


Info Views
Fully expanded action block
(action statements and views)
Info Views
Non-expanded action block (views only)
Info Views
USE xxx
Fully expanded action block with
non-expanded USE action block (views only)
USEdAction Block
xxx
Info Views
Info Views
USE xxx
Fully expanded action block with fully
Expande USE action block
USEdAction Block
xxx
Info Views

The following diagram shows referenced entity types.


Parent Subject Area
Referenced Entity Types
Entity Type A
Entity Type B
Entity Type C
Referenced entity types in their parent subject area
Entity types can be referenced by expected effects or the views of actions diagrams.

The following tables show action block design, unit test, and system test:

All Views, Entity Types, Parent Subject Area, System Defaults

This action block expansion	Includes the following additional objects for all subset types	with this protection
Default, Full	All views associated with the action blocks.	Modify
Default, Full	Entity types referenced by the views.	Access
Short, Default	Owning entity type	Access
Default, Full	Parent subject area of each entity type.	Access
Default, Full	System defaults of business system owning the BAA or BSD action block.	Access
Default, Full	Typemap associated to the action block	Modify

Action Statements

This action block expansion	Includes the following additional objects	with this protection
Default, Full	All action statements in the action block.	Delete
Default, Full	ASYNC response event usages	Delete

Fully Expanded USEd Action Blocks and Action Blocks They USE

This action block expansion	Includes the following additional objects	with this protection
Default only	Views Only of USEd Action Blocks	Access
Full only	Fully expanded action blocks USEd by the scoped action block. (Fully expanded action blocks include views *and* action statements.)	Modify
Default only	Views only of all action blocks USEd by the scoped action block.	Access
Full only	All action blocks USEd by the scoped action block.	Modify

Data Interface Modules

This action block expansion	Includes the following additional objects	with this protection
Short, Default, Full	Bind package default	Delete
Short, Default, Full	RDB table usage	Delete

Batch Job

To perform code generation of one or all the batch job steps that are contained in the batch job with the Construction toolset, use the Batch Job scoping object.

A batch job is used to implement the procedure steps of a single procedure as a batch job. Steps cannot be excluded from the procedure nor included from other procedures. The expansion of the batch job is applied to each procedure step in the associated procedure rather than to the procedure itself.

Note: Jobs, like procedures, have four expansions: Short, A, Default, and Full.

Unit Test

The following tables show batch job unit test:

Procedure Steps Action Blocks, Implementation Logic, DBRM

This Batch Job expansion	Includes the following additional objects for Unit Test	with this protection
See note	**Note:** The short, A, and Full expansion options for batch job apply only to the procedure steps within the procedure being implemented.	
All	The procedure steps implemented by the procedure implemented by the batch job.	Modify
All	Fully expanded action block and USEd action blocks for each procedure step of the procedure, if appropriate.	Modify
All	Implementation logic units for these action blocks, if expanded.	Delete
All	DBRM for any implementation logic unit for these action blocks.	Delete

Job, Job Steps, Procedure and Procedure Step Execution Units, and PSB

This action block expansion	Includes the following additional objects	with this protection
All	The scoped batch job.	Modify
All	All batch job steps contained in the batch job.	Modify

This action block expansion	Includes the following additional objects	with this protection
All	The procedure implemented by the scoped batch job, containing the procedure steps of each batch job step.	Modify
All	The procedure step execution units for the batch job steps.	Delete
All	The PSB (program specification block) for the scoped batch job.	Access

System Defaults

This Batch Job expansion	Includes the following additional objects	with this protection
All	The business system defaults (PF keys, commands, exit states, and so forth).	Access

TD Action Blocks, DBRMs and Implementation Logic

This Batch Job expansion	Includes the following additional objects	with this protection
All	TD action blocks for entities or tables.	Read
All	Implementation logic units for each TD action block.	Read
All	DBRM for the implementation logic units of TD action blocks.	Read

System Test

The following table shows batch job system test:

Entity Type Shared Companion Expansion, and DBRM

This Batch Job expansion	Includes the following additional objects for System Test	with this protection
All	Expanded treatment of entity shared companion of any entity type used in the views of the action blocks.	Read
All	DBRM for any implementation logic unit for any action block.	Read

Batch Job Step

Use the batch job step scoping object to perform load module packaging maintenance tasks with the Packaging Tool.

Scope a batch job step as part of unit testing, to use the Construction Tool to generate or regenerate a single load module for a batch job.

The expansion level and protection option of the scoped batch job step are applied to the procedure step corresponding to that batch job step. In fact, this is the only part of job step expansion that changes. All objects other than procedure step are brought in for all five levels.

Note: The five expansion options for batch job step apply only to the procedure steps within the procedure being implemented. All objects other than procedure steps are brought in for all five levels.

Unit Test

The following tables show batch job step unit test:

Procedure Step

This Batch Job Step expansion	Includes the following additional objects for Unit Test	with this protection
All	The procedure step implemented by the job step with the same expansion level as the job step.	Modify

Job Step, Job and Procedure

This Batch Job Step expansion	Includes the following additional objects for Unit Test	with this protection
All	The scoped batch job step.	Modify
All	The batch job containing the batch job step.	Access
All	The procedure implemented by the batch job.	Access

Action Block, Views, Entity Types, Subject Areas

This Batch Job Step expansion	Includes the following additional objects for Unit Test	with this protection
All	The action block for the procedure step contained in the scoped batch job step.	Modify
All	All views for the procedure step action blocks.	Modify
All	All entity types used in the views.	Access
All	All subject areas of the entity types used in the views.	Access

Program Specification Block (PSB)

This Batch Job Step expansion	Includes the following additional objects for Unit Test	with this protection
All	The program specification block (PSB) for the scoped batch job step.	Access

Other Procedure Steps

This Batch Job Step expansion	Includes the following additional objects for Unit Test	with this protection
All	Other procedure steps in the procedure with short expansion (views only).	Access

System Defaults

This Batch Job Step expansion	Includes the following additional objects for Unit Test	with this protection
All	The business system defaults (PF keys, commands, exit states and so forth).	Access

Data Tables and Databases

This Batch Job Step expansion	Includes the following additional objects for Unit Test	with this protection
All	Fully expanded data table for any transformed entity type used in the view of the procedure step or other USEd action blocks.	Read
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables	Delete
All	Database and technical design for the data tables.	Read

TD Action Blocks and Their DBRMs, and Implementation Logic

This Batch Job Step expansion	Includes the following additional objects for Unit Test	with this protection
All	TD action blocks for entity or table.	Read
All	Implementation logic units for each TD action block.	Read
All	DBRM for the TD action blocks.	Read

Packaging

This Batch Job Step expansion	Includes the following additional objects for Unit Test	with this protection
All	Package list entry	Delete

System Test

The following tables show batch job step system test:

Procedure Step

This Batch Job Step expansion	Includes the following additional objects for System Test	with this protection
All	The procedure step implemented by the job step with the same expansion level as the job step.	Modify

Action Block, Implementation Logic, and DBRM

This Batch Job Step expansion	Includes the following additional objects for System Test	with this protection
All	The fully expanded action block and USEd action blocks for the procedure step contained in the scoped batch job step, if appropriate.	Modify
All	The implementation logic unit for each of these action blocks.	Delete
All	DBRM for any implementation logic unit for these action blocks.	Delete

Entity Type Shared Companion Expansion, DBRM

This Batch Job Step expansion	Includes the following additional objects for System Test	with this protection
All	Expanded treatment of entity shared companions of any entity type used in the views of the action blocks.	Read
All	DBRM for any implementation logic unit for *any* action blocks.	Read

Job Step, Job, and Procedure

This Batch Job Step expansion	Includes the following additional objects for System Test	with this protection
All	The scoped batch job step.	Modify
All	The batch job containing the batch job step.	Access
All	The procedure implemented by the batch job.	Access

Views

This Batch Job Step expansion	Includes the following additional objects for System Test	with this protection
All	All views for the procedure step action blocks.	Modify

Program Specification Block (PSB)

This Batch Job Step expansion	Includes the following additional objects for System Test	with this protection
All	The program specification block (PSB) for the scoped batch job step.	Access

Other Procedure Steps

This Batch Job Step expansion	Includes the following additional objects for System Test	with this protection
All	Other procedure steps in the procedure with short expansion (views only)	Access

System Defaults

This Batch Job Step expansion	Includes the following additional objects for System Test	with this protection
All	The business system defaults (PF keys, commands, exit states, and so forth).	Access

Data Tables and Databases

This Batch Job Step expansion	Includes the following additional objects for System Test	with this protection
All	Fully expanded data table for any transformed entity type used in the view of the procedure step or other USEd action blocks.	Read
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables	Delete
All	Database and technical design for the data tables.	Read

TD Action Blocks and Their DBRMs, and Implementation Logic

This Batch Job Step expansion	Includes the following additional objects for System Test	with this protection
All	Fully expanded TD action blocks for entities or tables.	Read
All	Implementation logic units for each TD action block.	Read
All	DBRM for the TD action blocks.	Read

Packaging

This Batch Job Step expansion	Includes the following additional objects for System Test	with this protection
All	Package list entry	Delete

Business System

To modify the system defaults, the complete Dialog Flow Diagram, or, with full expansion, any screen, window, or action diagram in the entire business system, use the business system scoping object.

When to Use

Include a business system in the definition of a subset to:

- Change the default information of the system (system commands, PF keys, screen templates, exit state, video attributes, custom video properties, and edit patterns)
- Change dialog flow or view matching in ways that affect many of the procedure steps of the business system

Important! Request Full expansion of a business system only if the following option is true: a single developer is working on the entire business and you delete a widely used system default.

Suggestion

Because it is difficult to delete business system defaults, define your defaults carefully before beginning the dialog flow design, screen or window design, and the Procedure Action Diagrams.

The following tables show business system design, unit test, and system test:

System Defaults and Technical System

This Business System expansion	Includes the following additional objects for all subset types	with this protection
Short, Default, Full	System defaults (PF keys, commands, exit states, templates, video attributes, custom video properties, and edit patterns).	Modify
Short, Default, Full	All bilingual text for system defaults.	Delete
Short, Default, Full	Unexpanded technical system.	Modify

All Procedures, Procedure Steps, Dialog Flows, and Views

This Business System expansion	Includes the following additional objects for all subset types	with this protection
Default, Full	All procedures and procedure steps in the business system. (not including the action diagrams and screens for the procedure steps).	Modify
Default, Full	Dialog flow information used by the procedures.	Delete

This Business System expansion	Includes the following additional objects for all subset types	with this protection
Default, Full	All views in the business system.	Modify

All Screens, Windows, Action Diagrams, Referenced Entity Types

This Business System expansion	Includes the following additional objects for all subset types	with this protection
Full	All screens in the business system.	Modify
Full	All fully expanded windows (with all window objects) and dialog boxes in the business system.	Delete
Full	All action blocks in the business system.	Modify
Full	All bilingual text for screen objects.	Delete
Full	All entity types referenced in all views. To modify these entity types you must specify them as scoping objects.	Access
Full	Parent subject areas for the entity types.	Access

Data Interface Modules

This Business System expansion	Includes the following additional objects for all subset types	with this protection
Short, Default, Full	Bind package default	Delete
Short, Default, Full	Package list entry	Delete

Component Implementation

The following tables show component implementation design, unit test, and system test:

Parent Subject Area

This Component Implementation expansion	Includes the following additional objects	with this protection if scoped with Modify
Short, Default, Full	Parent subject areas of the scoped component implementation	Access

TD Action Blocks (for data integrity)

This Component Implementation expansion	Includes the following additional objects	with this protection if scoped with Modify
Short, Default, Full	The TD action block for the scoped component implementation	Modify
Short, Default, Full	DBRM and implementation unit for the TD action block	Modify

Component Model

To support operations, encapsulation, classifying attributes or to display attributes and operations on an entity type, use the component model scoping object.

Include the component model in a subset definition to:

- Modify or delete art objects
- Change references to entity types, relationships, subtypes, or partitionings

The following table shows component model design, unit test, and system test:

Art Objects

This Component Model expansion	Includes the following additional objects	with this protection if scoped with Modify
Default, Full	Art objects	Delete

Entity Types

This Component Model expansion	Includes the following additional objects	with this protection if scoped with Modify
Default	Referenced entity types	Access
Full	Referenced entity types	Modify
Default	Referenced relationships	Access
Full	Referenced relationships	Modify
Default	Referenced subtypes	Access
Full	Referenced subtypes	Modify
Default	Referenced partitionings	Access
Full	Referenced partitionings	Modify
Default	Parent entity types of referenced relationships	Access
Full	Parent entity types of referenced relationships	Modify
Default	Parent entity types of referenced subtypes	Access
Full	Parent entity types of referenced subtypes	Modify
Default	Parent entity types of referenced partitionings	Access
Full	Parent entity types of referenced partitionings	Modify

Subject Area

This Component Model expansion	Includes the following additional objects	with this protection if scoped with Modify
Default	Scoping Subject Area	Access
Full	Scoping Subject Area	Modify

Component Specification

The following tables show component specification design, unit test, and system test:

Parent Subject Area

This Component Specification expansion	Includes the following additional objects	with this protection if scoped with Modify
Short, Default, and Full	Parent subject areas of the scoped component specification.	Access

TD Action Blocks (for data integrity)

This Component Specification expansion	Includes the following additional objects	with this protection if scoped with Modify
Short, Default, and Full	The TD action block for the scoped component specification	Modify
Short, Default, and Full	DBRM and implementation unit for the TD action block	Modify

Interface Types

This Component Specification expansion	Includes the following additional objects	with this protection if scoped with Modify
Default, Full	Interface types offered by component specification	Modify
Default, Full	Interface type model scoped by each interface type	Modify

Configuration Instance

The following table shows Configuration Instance design, unit test, and system test:

This Configuration Instance expansion	Include the following additional object types for all subset types	with this protection
Default Full	Referenced Business System	Access

This Configuration Instance expansion	Include the following additional object types for all subset types	with this protection
Default	Referenced Load Module	Access
Full	Referenced Load Module	Modify
Default	Referenced Procedure Step	Access
Full	Referenced Procedure Step	Modify
Default	Referenced Action Block	Access
Full	Referenced Action Block	Modify
Default	Parent Procedure Step for Referenced Window and Dialog Box	Access
Full	Parent Procedure Step for Referenced Window and Dialog Box	Modify
Default	Referenced Storage Group	Access
Full	Referenced Storage Group	Modify
Default	Referenced Database	Access
Full	Referenced Database	Modify
Default	Parent Database for referenced Tablespace and Index	Access
Full	Parent Database for referenced Tablespace and Index	Modify
Default	Referenced Record	Access
Full	Referenced Record	Modify
Default	Scoping Subject Area	Access
Full	Scoping Subject Area	Modfiy

Current Info System

The following table shows current info system design, unit test, and system test:

This Current Info System expansion	Include the following additional object types for all subset types	with this protection
Default	Produced Current Effects	Modify

Custom Proxies

The following table shows custom proxy definition design, unit test, and system test:

This Custom Proxies expansion	Include the following additional object types for all subset types	with this protection
Short Default Full	Owning Business System	Access
Short Default Full	Custom Proxies included in the Custom Proxy Definition	Modify
Short	Procedure Steps included in the Custom Proxy Definition	Modify
Default	Procedure Steps included in the Custom Proxy Definition	Modify
Full	Procedure Steps included in the Custom Proxy Definition	Modify

Data Table

To let multiple users update tables in the same database, use the data table scoping object.

Design and Unit Test

The following tables show data table design and unit test:

Indexes, Columns, Link Tables, and Constraints

This Data Table expansion	Includes the following additional objects for Design and Unit Test	with this protection
Default	All indexes defined for the table.	Delete
Short, Default	Extended indexes for included indexes	Delete
Default	The indexspaces containing the indexes.	Delete
Short, Default	Extended indexspaces for included indexspaces	Delete
Default	All columns contained in the table.	Modify
Short, Default	Extended columns	Delete
Default	All from and to tables having constraints to the table.	Read
Default	All constraints to and from the table. To modify the link table, you must also scope the data table on the other end with Modify protection.	Modify
Short, Default	Extended constraints for included constraints	Delete

Database, Data Store, Data Sets, and Storage Groups

This Data Table expansion	Includes the following additional objects for Design and Unit Test	with this protection
Default	Database that contains the table.	Modify
Default	Data store that contains the table.	Delete
Short, Default	Data sets for all included tablespaces and indexspaces	Delete
Default	Any storage group that contains the data sets.	Access
Default	Any default storage group for the database.	Access
See note	All storage groups in the technical design object. **Note:** Included only if you receive Delete protection on the scoped table.	Delete
Short, Default	Tablespace for the base data table	Modify

This Data Table expansion	Includes the following additional objects for Design and Unit Test	with this protection
Short, Default	Tablespace referenced by extended tables	Access
Short, Default	Tablespace referenced by extended indexspace Oracle	Access
Short, Default	Extended tablespaces for included tablespaces	Delete
Short, Default	Extended tables	Delete

Implemented Entity Type and Parent Subject Areas

This Data Table expansion	Includes the following additional objects for Design and Unit Test	with this protection
Default	Entity type implemented by the data table.	Access
Default	Parent subject areas for the entity types.	Access

TD Action Blocks

This Data Table expansion	Includes the following additional objects for Design and Unit Test	with this protection
Default	Unexpanded TD action blocks associated with a table using a constraint.	Modify
Default	Unexpanded TD action block associated with an entity implemented by the data table.	Access

System Test

The following tables show data table system test:

Indexes, Columns, Link Tables and Constraints

This Data Table expansion	Includes the following additional objects for System Test	with this protection
Default	All indexes defined in the table.	Modify
Short, Default	Extended indexes for included indexes	Delete
Default	The indexspaces containing the index.	Delete

This Data Table expansion	Includes the following additional objects for System Test	with this protection
Short, Default	Extended indexspaces for included indexspaces	Delete
Default	All columns contained in the table.	Modify
Short, Default	Extended columns	Delete
Default	From and to tables having constraints to the table.	Read
Default	All constraints to and from the table. To modify the link table, you must also scope the table on the other end with Modify protection.	Modify
Short, Default	Extended constraints for included constraints	Delete

Database, Data Store, Data Sets, Storage Groups

This Data Table expansion	Includes the following additional objects for System Test	with this protection
Default	Database that contains the table.	Modify
Default	Data store that contains the table.	Modify
Short, Default	Data sets for all included tablespaces and indexspaces.	Delete
Default	Any storage group that contains the data sets.	Access
Default	Any default storage group for the database.	Access
See note	All storage groups in the technical design object. **Note:** Included only if you receive Delete protection on the scoped table.	Delete
Short, Default	Tablespace for the base data table	Modify
Short, Default	Tablespace referenced by extended tables	Access
Short, Default	Tablespace referenced by extended indexspace Oracle	Access
Short, Default	Extended tablespaces for included tablespaces	Delete
Short, Default	Extended tables	Delete

Implemented Entity Type and Parent Subject Areas

This Data Table expansion	Includes the following additional objects for System Test	with this protection
Default	Entity type implemented by the table.	Access

This Data Table expansion	Includes the following additional objects for System Test	with this protection
Default	Parent subject areas for the entity types.	Access

TD Action Blocks

This Data Table expansion	Includes the following additional objects for System Test	with this protection
Default	Expanded TD action blocks associated with a table using constraint.	Access
Default	Expanded TD action block associated with an entity implemented by the data table.	Access
Default	Expanded USEd TD action blocks.	Access

Database

Use the database scoping object when you want to maintain a database with the Data Structure List and Data Storage List.

Design and Unit Test

The following tables show database design and unit test:

Databases, Data Stores, Data Sets and Storage Groups

This Database expansion	Includes the following additional objects for Design and Unit Test	with this protection
Short, Default	Database files	Delete
Short, Default	Extended databases	Delete
Default	All data stores that belong to the database.	Delete
Default	Tablespaces	Modify
Short, Default	Tablespaces referenced by extended DB2 and UNIX databases	Access
Default	Tablespaces referenced by extended tables	Delete

This Database expansion	Includes the following additional objects for Design and Unit Test	with this protection
Default	Tablespace referenced by extended indexspace Oracle	Delete
Short, Default	Extended tablespaces for included tablespaces	Delete
Default	Any default storage group for the database.	Access

Any storage groups that contain the data sets.

This Database expansion	Includes the following additional objects for Design and Unit Test	with this protection
See note	All storage groups in the technical design object. **Note:** Included only if you receive Delete protection on the scoped database.	Delete
Default	Data sets for all tablespaces and indexspaces	Delete

Data Tables and Columns

This Database expansion	Includes the following additional objects for Design and Unit Test	with this protection
Default	All tables stored in the database.	Modify
Default	Extended tables	Delete
Default	All columns contained in the tables.	Modify
Default	Extended columns	Delete
Default	From and to tables from other databases for constraints.	Access

Indexes, Indexspaces, and Constraints

This Database expansion	Includes the following additional objects for Design and Unit Test	with this protection
Default	All indexes for the tables.	Delete

This Database expansion	Includes the following additional objects for Design and Unit Test	with this protection
Default	Extended indexes for included indexes	Delete
Default	The indexspaces for the indexes.	Delete
Default	Extended indexspaces for included indexspaces	Delete
Default	All constraints to and from the tables.	Modify
Default	Extended constraints for included constraints	Delete

Implemented Entity Types and Parent Subject Areas

This Database expansion	Includes the following additional objects for Design and Unit Test	with this protection
Default	Entity types implemented by the tables.	Access
Default	Parent subject areas for the entity types.	Access

Technical Design Object

This Database expansion	Includes the following additional objects for Design and Unit Test	with this protection
Default	Technical design object that uses the database.	Modify

System Test

The following tables show database system test:

Databases, Data Stores, Data Sets, Storage Groups

This Database expansion	Includes the following additional objects for System Test	with this protection
Short, Default	Database files	Delete
Short, Default	Extended databases	Delete
Default	All data stores that belong to the database.	Modify
Default	Tablespaces	Modify

This Database expansion	Includes the following additional objects for System Test	with this protection
Short	Tablespace referenced by extended DB2 and UNIX databases	Delete
Default	Tablespace referenced by extended tables	Delete
Short, Default	Extended tablespaces for included tablespaces	Delete
Default	Any default storage group for the database.	Access
Default	Any storage groups that contains the data sets.	Access
See note	All storage groups in the technical design object. **Note:** Included only if you receive Delete protection on the scoped table.	Delete
Default	Data sets for all tablespaces and indexspaces	Delete

Data Tables

This Database expansion	Includes the following additional objects for System Test	with this protection
Default	All data tables stored in the database.	Modify
Default	Extended tables	Delete
Default	From and to tables from other databases for constraints.	Access

Indexes, Columns, and Constraints

This Database expansion	Includes the following additional objects for System Test	with this protection
Default	All indexes for the tables.	Modify
Default	Extended indexes for included indexes	Delete
Default	All columns contained in the tables.	Modify
Default	Extended columns	Delete

This Database expansion	Includes the following additional objects for System Test	with this protection
Default	All constraints to and from the tables.	Modify
Default	Extended constraints for included constraints	Delete
Default	Entity types implemented by the tables.	Access
Default	Parent subject areas for the entity types.	Access

Technical Design Object

This Database expansion	Includes the following additional objects for System Test	with this protection
Default	Technical design object that uses the database.	Modify

TD Action Blocks

This Database expansion	Includes the following additional objects for System Test	with this protection
Default	Expanded TD action block associated with a table using constraints	Modify
Default	Expanded TD action block associated with an entity implemented by the data tables.	Access

Dialect

Use the dialect scoping object for creating bilingual text for models with the Design toolset. Dialect is not scoped for unit or system testing.

Include a scoping dialect in the definition of a subset to:

- Add a dialect (language) type to a model to create bilingual text for business system defaults, exit states, and prompts
- Delete an unused dialect from a model

Once you have defined a new dialect, it is not necessary to scope it to add bilingual text in that language. All dialects in a model are included in your subset definition as part of the default set of objects, if they are not scoped directly.

Note: You must scope a business system (or any object whose expansion includes a business system) to access the dialect functions. Dialects are added using the Dialect Definition option on the Design menu.

The following table shows dialect design:

This Dialect expansion	Includes the following additional objects in Design	with this protection
Default	The scoped dialect object.	Modify

Entity Type

To maintain the part of the model that is related to the subset's entity types with the Data Modeling tool, use the entity type scoping object.

When to Use

Include an entity type in a subset definition to:

- Add new components, such as attributes, to the entity type
- Change the properties or description of the entity type or any of its components,
- Create a view that references the entity type (if no other view in the subset already references it).

Rules

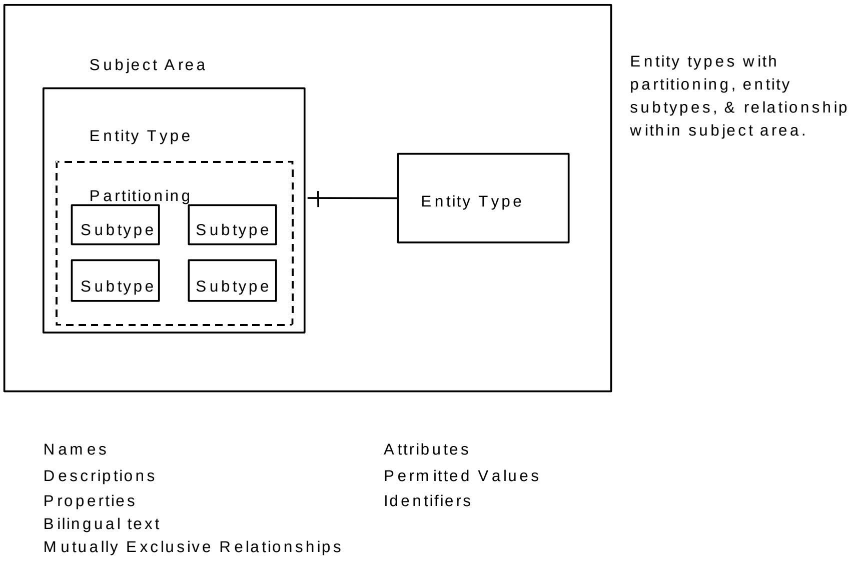
When you secure Delete usage for an entity type, you receive Delete usage for all its relationships.

If new entity types are added to the subset at the workstation, relationships are added between the new entity types and the entity types that are scoped with Modify protection in the subset definition.

To include data tables, databases, and TD action blocks, specify:

- Any expansion of entity type
- Unit Test or System Test subset type

The following diagram shows entity types and subtypes.



Design

The following tables show entity type design:

Entity Type, Entity Subtype, Details, Relationships

This Entity expansion	Includes the following additional objects for Design	with this protection
Default, Full	Entity type or work attribute set included contents are used by	Access

This Entity expansion	Includes the following additional objects for Design	with this protection
Short Only	A chain of shared companion entity types that have identifying relationships to the scoped entity type, its subtypes, or its shared companion entity types. The shared companion entity types include their attributes and permitted values and additional layers of shared companions based on identifiers of the first set of shared companion entity types found for the scoped entity type. Short expansion brings in all identifying relationships down the chain of entity types and for subtypes. It includes identifying shared companion relationships, not anything else.	Access
Default, Full	All shared companion entity types. The shared companion entity types include their attributes and permitted values. Default expansion includes every relationship whether it is an identifying relationship.	Access
Short, Default, Full	Parent Entity of attributes from which foreign key attributes are derived	Access
Short, Default, Full	Subtypes	Modify
Short, Default, Full	Partitionings	Modify
Short, Default, Full	All attributes and foreign key attributes (including those of subtypes).	Modify
Short, Default, Full	Aliases and Identifiers.	Delete
Short, Default, Full	All permitted values, prompt for values, and prompts.	Modify
Short, Default	All bilingual text for prompts.	Delete
Default, Full	Contents	Delete
Short, Default, Full	All relationships between scoped entity types.	Modify
Short, Default, Full	Relationships for included attribute relationship usages	Access
Short, Default, Full	Attribute relationship usages	Delete

This Entity expansion	Includes the following additional objects for Design	with this protection
Default only	Any mutually exclusive relationship set structures of the scoped entity type.	Modify
Full	Mutually exclusives.	Delete

Parent Subject Areas

This Entity expansion	Includes the following additional objects for Design	with this protection
Short, Default, and Full	Parent subject areas of the scoped entity type.	Access
Short, Default	Parent subject areas of shared companion entity types.	Access

Action Blocks for Derived Attributes

This Entity expansion	Includes the following additional objects for Design	with this protection
Short, Default, Full	All action blocks for derived or default attributes.	Modify
Short, Default	All action blocks USEd by action blocks for derived attributes.	Modify
Default	Owned action blocks	Access
Full	Owned action blocks	Modify

Views of Action Blocks

This Entity expansion	Includes the following additional objects for Design	with this protection
Short, Default	All views of all action blocks in the subset.	Modify
Short, Default	All components of entity types in the views.	Access

Entity State Transitions

This Entity expansion	Includes the following additional objects for Design	with this protection
Short, Default, Full	Entity state transition objects associated to the entity type and its subtypes.	Delete

Data Table, Database, and Technical Design

This Entity expansion	Includes the following additional objects for Design	with this protection
Short, Default, Full See note	The data table implementing the scoped entity type if it has been transformed. **Note:** Included only if the entity type is scoped with Modify or Delete protection.	Read
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables	Delete
See note	The database and technical design for the table **Note:** Included only if the entity type is scoped with Modify or Delete protection.	Delete

TD Action Blocks (for data integrity)

This Entity expansion	Includes the following additional objects for Design	with this protection
Short, Default, Full	The TD action block for the scoped entity type.	Modify
Short, Default	The TD action block for any shared companion entity type.	Access
Short, Default, Full	DBRM and implementation unit for TD action blocks.	Modify

Transactions

This Entity expansion	Includes the following additional objects for Design	with this protection
Default	Owned transactions	Access

This Entity expansion	Includes the following additional objects for Design	with this protection
Full	Owned transactions	Modify

Unit Test

The following tables show entity type unit test:

Entity Type, Entity Subtype, Details, Relationships

This Entity expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	The scoped entity type and all its partitionings including subtypes.	Modify
Short, Default	Partitionings	Modify
Default	Entity type or work attribute set included contents are used by	Access
Short, Default	All attributes (including those of subtypes).	Modify
Short, Default	All names, descriptions, properties, permitted values, and identifiers.	Modify
Short, Default	All bilingual text for prompts.	Delete
Default	Contents	Delete
Short, Default	All relationships between scoped entity types.	Modify

Parent Subject Areas

This Entity expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	Parent subject areas of the scoped entity type.	Access
Short, Default	Parent subject areas of shared companion entity types.	Access

Action Blocks for Derived Attributes

This Entity expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	All action blocks for derived attributes.	Modify
Short, Default	All action blocks USEd by action blocks for derived attributes.	Modify
Default	Owned action blocks	Access

Views of Action Blocks

This Entity expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	All views of all action blocks in the subset.	Modify
Short, Default	All components of entity types in the views.	Access

Data Table, and Databases

This Entity expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	Fully expanded data table for the transformed entity type.	Read
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables	Delete
Short, Default	The database and technical design for the data table.	Read

TD Action Blocks, DBRMs and Implementation Logic

This Entity expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	Unexpanded TD action block for the scoped entity type and table.	Modify

This Entity expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	DBRM and implementation logic units for the TD action blocks.	Modify

Entity Type used in the views of these Action Blocks

This Entity expansion	Includes the following additional objects for Unit Test	with this protection
Short Only	A chain of shared companion entity types that have identifying relationships to the scoped entity type, its subtypes, or its shared companion entity types. The shared companion entity types include their attributes and permitted values and additional layers of shared companion based on identifiers of the first set of shared companion entity types found for the scoped entity type.	Read
Default Only	All shared companion entity types. The shared companion entity types include their attributes and permitted values. The first shared companion layer is every entity type with a relationship to the scoped entity type. The second layer of shared companions is based on identifiers of the first level of shared companion entity types found.	Access

Transactions

This Entity expansion	Includes the following additional objects for Unit Test	with this protection
Default	Owned transactions	Access
Default Only	Any mutually exclusive relationship set of the scoped entity type.	Modify

System Test

The following tables show entity type system test:

Entity Type, Entity Subtype, Details, Relationships

This Entity expansion	Includes the following additional objects for System Test	with this protection
Short, Default	The scoped entity type and all its partitionings including entity subtypes.	Modify
Short, Default	Partitionings	Modify
Default	Entity type or work attribute set included contents are used by	Access
Short, Default	All attributes (including those of subtypes).	Modify
Short, Default	All names, descriptions, properties, permitted values, prompts, and identifiers.	Modify
Short, Default	All bilingual text for prompts.	Delete
Default	Contents	Delete
Short, Default	All relationships between scoped entity types.	Modify

Parent Subject Area

This Entity expansion	Includes the following additional objects for System Test	with this protection
Short, Default	Parent subject areas of the scoped entity type.	Access
Short, Default	Parent subject areas of shared companion entity types.	Access

Action Blocks for Derived Attributes

This Entity expansion	Includes the following additional objects for System Test	with this protection
Short, Default	All action blocks for derived attributes.	Modify
Short, Default	All action blocks USEd by action blocks for derived attributes.	Modify
Default	Owned action blocks	Access

Views of Action Blocks

This Entity expansion	Includes the following additional objects for System Test	with this protection
Short, Default	All views of all action blocks in the subset.	Modify
Short, Default	All components of entity types in the views.	Access

Data Table and Database

This Entity expansion	Includes the following additional objects for System Test	with this protection
Short, Default	Fully expanded data table for the transformed entity type.	Read
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables	Delete
Short, Default	The database and technical design for the table.	Read

TD Action Blocks, DBRMs, and Implementation Logic

This Entity expansion	Includes the following additional objects for System Test	with this protection
Short, Default	Fully expanded TD action blocks for entity types, data tables.	Modify
Short, Default	DBRM and implementation logic units for each TD action block.	Modify

Entity Type used in the Views of these Action Blocks

This Entity expansion	Includes the following additional objects for System Test	with this protection
Default	All entity types with any kind of relationship to the used entity type or one of its neighbors, that is, its full chain of relationships, not just identifying relationships.	Read
Short Only	A chain of shared companion entity types that have identifying relationships to the scoped entity type, its subtypes, or its shared companion entity types. The shared companion entity types include their attributes and permitted values and additional layers of shared companions based on identifiers of the first set of shared companion entity types found for the scoped entity type.	Access
Default Only	All shared companion entity types. The shared companion entity types include their attributes and permitted values. The first shared companion layer is every entity type with a relationship to the scoped entity type. The second layer of shared companions is based on identifiers of the first level of shared companion entity types found.	Access

Transactions

This Entity expansion	Includes the following additional objects for System Test	with this protection
Default	Owned transactions	Access
Default Only	Any mutually exclusive relationship set of the scoped entity type.	Modify

Exit State

Use the exit state scoping object to maintain:

- Global exit states (those defined during Analysis and belonging to the entire model).
- Business system-specific exit states (those defined during the Design stage and belonging to a specific business system).

Include an exit state in a subset to:

- Add or change its message
- Change its type (rollback compared to normal)

If global exit states exist in a model, they are included in the subset as part of the default set, if not previously included through the expansion of some other object.

The following tables show exit state design, unit test, and system test:

Exit State, Message, and Bilingual Tex

This Exit State expansion	Includes the following additional objects	with this protection
Default	The scoped exit state.	Modify
Default	Message and bilingual text.	Delete

System Defaults and Bilingual Text

This Exit State expansion	Includes the following additional objects	with this protection
Default See note	The business system it belongs to, at short expansion (system defaults only). **Note:** For BSD exit states only.	Access
Default See note	Any bilingual text for the system defaults. **Note:** For BSD exit states only.	Read

Matrix Usages and Cell Values

This Exit State expansion	Includes the following additional objects	with this protection
Default See note	Any matrix usages. **Note:** Included only if Delete protection is requested for the scoped exit state.	Access
Default See note	Any matrix cell values. **Note:** Included only if Delete protection is requested for the scoped exit state.	Access

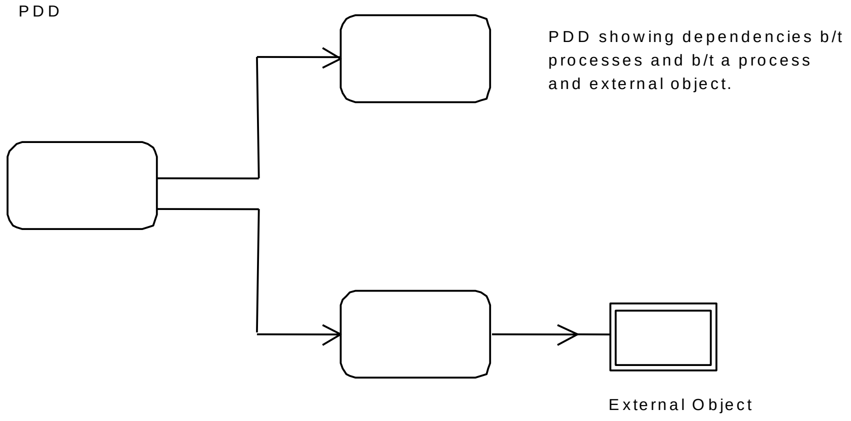
External Object

To maintain external objects in the Process Dependency Diagram (PDD), use the external object scoping object.

Include an external object in a subset to:

- Associate a new external object with a function or process.
- Change the external description of the object.
- Delete an external object (if it has never been used).

The following diagram shows external object.


PDD
PDD showing dependencies b/t processes and b/t a process and external object.
External Object

The following tables show external object design, unit test, and system test:

External Object

This External Object expansion	Includes the following additional objects for all subset types	with this protection
Default	The scoped external object.	Modify

Matrix Usages and Cell Values

This External Object expansion	Includes the following additional objects for all subset types	with this protection
Default See note	Any matrix usages.	Access
Default See note	Any Matrix cell values.	Access
	Note: Included only if external object is scoped with Delete protection.	

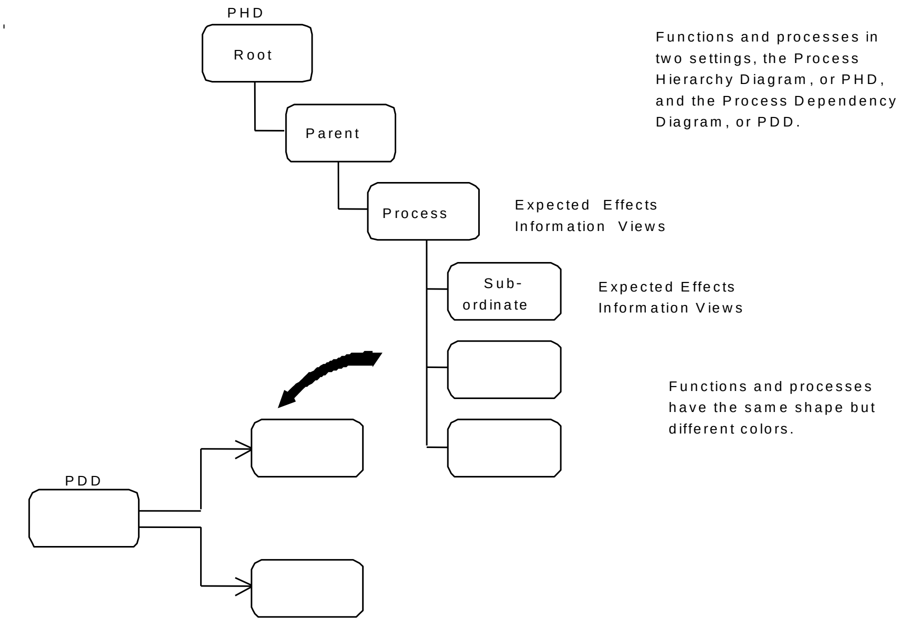
Function and Process

To maintain the part of the model that is related to the functions and processes of the subset with the Activity Hierarchy Diagram (AHD) and Activity Dependency Diagram (ADD), use the function scoping object.

Include functions and processes in a subset definition to:

- Change the AHD or ADD.
- Create or change the Process Action Diagram (PAD).
- Modify the views of a process.

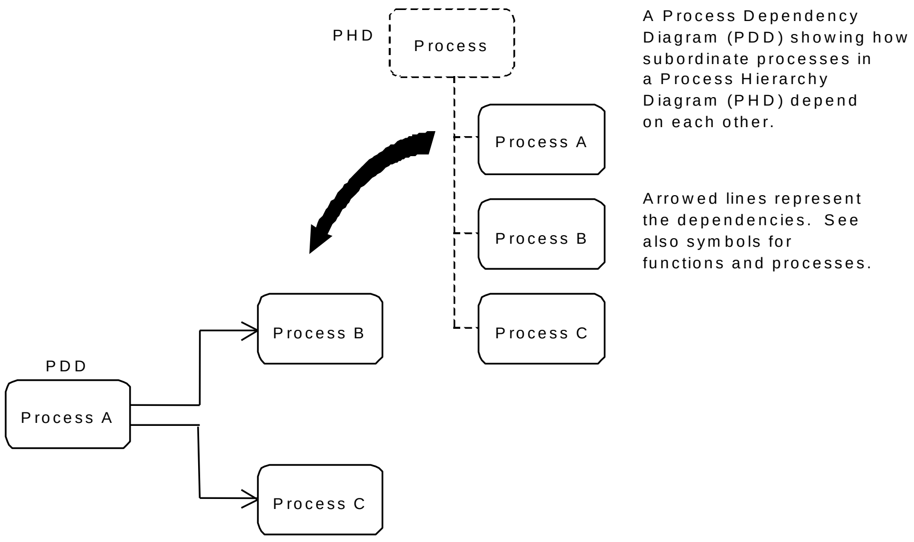
The following diagram shows functions and processes.



Functions and processes in two settings, the Process Hierarchy Diagram, or PHD, and the Process Dependency Diagram, or PDD.

Functions and processes have the same shape but different colors.

The following diagram shows dependencies.


PHD
Process
Process A
Process B
Process C
PDD
Process A
Process B
Process C
A Process Dependency Diagram (PDD) showing how subordinate processes in a Process Hierarchy Diagram (PHD) depend on each other.
Arrowed lines represent the dependencies. See also symbols for functions and processes.

The following tables show function or process design, unit test, and system test:

Function or Process, Subordinates, and Parents

This Function expansion	Includes the following additional objects	with this protection
Short, Default, Full	The scoped function or process.	Modify
Short, Default, Full	All functions and processes subordinate to the scoped object.	Modify
Short, Default, Full	Parents of the scoped object	Access

Views, Referenced Entity Types, and their Subject Areas

This Function expansion	Includes the following additional objects	with this protection
Short, Default, Full	All views for the scoped object and its subordinates.	Modify
Short, Default, Full	Entity types referenced in views or expected effects.	Access

This Function expansion	Includes the following additional objects	with this protection
Short, Default, Full	Parent subject areas of the entity types.	Access

External Objects and Events

This Function expansion	Includes the following additional objects	with this protection
Short, Default, Full	All external objects and events of the scoped object.	Modify

Dependencies

This Function expansion	Includes the following additional objects	with this protection
Short, Default, Full	All dependencies.	Modify

Entity State Transitions and Usages

This Function expansion	Includes the following additional objects	with this protection
Short, Default, Full	Entity type referenced by the transition.	Modify
Short, Default, Full	Entity state transition usages associated to the process.	Delete
Short, Default, Full	Entity state transitions associated to the entity state usage.	Read

Fully Expanded Action Blocks and Views of USEd Action Blocks

This Function expansion	Includes the following additional objects	with this protection
Default, Full	Fully expanded process action blocks of the scoped activity and its subordinate activities.	Modify
Default, Full	Views only of all USEd action blocks referenced by these process action blocks.	Access

This Function expansion	Includes the following additional objects	with this protection
Default, Full	Full expansion of all USEd action blocks referenced by these process action blocks.	Access

Fully Expanded USEd Action Blocks

This Function expansion	Includes the following additional objects	with this protection
Full	Full expansion of all USEd action blocks referenced by these process action blocks.	Modify

Interface Type

Design

The following tables show interface type design:

Entity Type, Entity Subtype, Details, Relationships

This Interface Type expansion	Includes the following additional objects for Design	with this protection
Default, Full	Entity type or work attribute set included contents are used by	Access
Short Only	A chain of shared companion entity types that have identifying relationships to the scoped interface type, its subtypes, or its shared companion entity types. The shared companion entity types include their attributes and permitted values and additional layers of shared companions based on identifiers of the first set of shared companion entity types found for the scoped interface type. Short expansion brings in all identifying relationships down the chain of entity types and for subtypes. It includes identifying shared companion relationships, not anything else.	Access

This Interface Type expansion	Includes the following additional objects for Design	with this protection
Default, Full	All shared companion entity types. The shared companion entity types include their attributes and permitted values. Default expansion includes every relationship whether or not it is an identifying relationship.	Access
Short, Default, Full	parent Entity of attributes from which foreign key attributes are derived	Access
Short, Default, Full	Subtypes	Modify
Short, Default, Full	Partitionings	Modify
Short, Default, Full	All attributes and foreign key attributes (including those of subtypes).	Modify
Short, Default, Full	Aliases and Identifiers.	Delete
Short, Default, Full	All permitted values, prompt for values, and prompts.	Modify
Short, Default	All bilingual text for prompts.	Delete
Default, Full	Contents	Delete
Short, Default, Full	All relationships between scoped entity types.	Modify
Short, Default, Full	Relationships for included attribute relationship usages	Access
Short, Default, Full	Attribute relationship usages	Delete
Default only	Any mutually exclusive relationship set structures of the scoped interface type.	Modify
Full	Mutually exclusives.	Delete

Parent Subject Areas

This Interface Type expansion	Includes the following additional objects for Design	with this protection
Short, Default, and Full	Parent subject areas of the scoped interface type.	Access
Short, Default	Parent subject areas of shared companion entity types.	Access

Action Blocks for Derived Attributes

This Interface Type expansion	Includes the following additional objects for Design	with this protection
Short, Default, Full	All action blocks for derived or default attributes.	Modify
Short, Default	All action blocks USEd by action blocks for derived attributes.	Modify
Default	Owned action blocks	Access
Full	Owned action blocks	Modify

Views of Action Blocks

This Interface Type expansion	Includes the following additional objects for Design	with this protection
Short, Default	All views of all action blocks in the subset.	Modify
Short, Default	All components of entity types in the views.	Access

Entity State Transitions

This Interface Type expansion	Includes the following additional objects for Design	with this protection
Short, Default, Full	Entity state transition objects associated to the entity type and its subtypes.	Delete

Data Table, Database, and Technical Design

This Interface Type expansion	Includes the following additional objects for Design	with this protection
Short, Default, Full See note	The data table implementing the scoped interface type if it has been transformed.	Read
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables	Delete
See note	The database and technical design for the table	Delete

This Interface Type expansion	Includes the following additional objects for Design	with this protection
	Note: Included only if the interface type is scoped with Modify or Delete protection.	

TD Action Blocks (for data integrity)

This Interface Type expansion	Includes the following additional objects for Design	with this protection
Short, Default, Full	The TD action block for the scoped interface type.	Modify
Short, Default	The TD action block for any shared companion entity type.	Access
Short, Default, Full	Implementation unit for TD action blocks.	Modify

Transactions

This Interface Type expansion	Includes the following additional objects for Design	with this protection
Default	Owned transactions	Access
Full	Owned transactions	Modify

Interface Type Model

This Interface Type expansion	Includes the following additional objects for Design	with this protection
Short, Default, Full	Interface type model scoped by interface type	Modify

Unit Test

The following tables show interface type unit test:

Entity Type, Entity Subtype, Details, Relationships

This Interface Type expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	The scoped interface type and all its partitionings including subtypes.	Modify
Short, Default	Partitionings	Modify
Default	Entity type or work attribute set included contents are used by	Access
Short, Default	All attributes (including those of subtypes).	Modify
Short, Default	All names, descriptions, properties, permitted values, and identifiers.	Modify
Short, Default	All bilingual text for prompts.	Delete
Default	Contents	Delete
Short, Default	All relationships between scoped entity types.	Modify

Parent Subject Areas

This Interface Type expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	Parent subject areas of the scoped interface type.	Access
Short, Default	Parent subject areas of shared companion entity types.	Access

Action Blocks for Derived Attributes

This Interface Type expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	All action blocks for derived attributes.	Modify
Short, Default	All action blocks USEd by action blocks for derived attributes.	Modify
Default	Owned action blocks	Access

Views of Action Blocks

This Interface Type expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	All views of all action blocks in the subset.	Modify
Short, Default	All components of entity types in the views.	Access

Data Table, and Databases

This Interface Type expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	Fully expanded data table for the transformed interface type.	Read
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables	Delete
Short, Default	The database and technical design for the data table.	Read

TD Action Blocks, DBRMs and Implementation Logic

This Interface Type expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	Unexpanded TD action block for the scoped interface type and table.	Modify
Short, Default	DBRM and implementation logic units for the TD action blocks.	Modify

Entity Type Used in the Views of these Action Blocks

This Interface Type expansion	Includes the following additional objects for Unit Test	with this protection
Short Only	A chain of shared companion entity types that have identifying relationships to the scoped interface type, its subtypes, or its shared companion entity types. The shared companion entity types include their attributes and permitted values and additional layers of shared companion based on identifiers of the first set of shared companion entity types found for the scoped interface type.	Read
Default Only	All shared companion entity types. The shared companion entity types include their attributes and permitted values. The first shared companion layer is every entity type with a relationship to the scoped interface type. The second layer of shared companions is based on identifiers of the first level of shared companion entity types found.	Access

Transactions

This Interface Type expansion	Includes the following additional objects for Unit Test	with this protection
Default	Owned transactions	Access
Default Only	Any mutually exclusive relationship set of the scoped interface type.	Modify

Interface Type Model

This Interface Type expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default, Full	Interface type model scoped by interface type	Modify

System Test

The following tables show interface type system test:

Entity Type, Entity Subtype, Details, Relationships

This Interface expansion	Includes the following additional objects for System Test	with this protection
Default	The scoped interface type and all its partitionings including entity subtypes.	Modify
Short, Default	Partitionings	Modify
Default	Entity type or work attribute set included contents are used by	Access
Default	All attributes (including those of subtypes).	Modify
Default	All names, descriptions, properties, permitted values, prompts, and identifiers.	Modify
Default	All bilingual text for prompts.	Delete
Default	Contents	Delete
Default	All relationships between scoped entity types.	Modify

Parent Subject Area

This Interface expansion	Includes the following additional objects for System Test	with this protection
Default	Parent subject areas of the scoped interface type.	Access
Default	Parent subject areas of shared companion entity types.	Access

Action Blocks for Derived Attributes

This Interface expansion	Includes the following additional objects for System Test	with this protection
Default	All action blocks for derived attributes.	Modify

This Interface expansion	Includes the following additional objects for System Test	with this protection
Default	All action blocks USEd by action blocks for derived attributes.	Modify
Default	Owned action blocks	Access

Views of Action Blocks

This Interface expansion	Includes the following additional objects for System Test	with this protection
Default	All views of all action blocks in the subset.	Modify
Default	All components of entity types in the views.	Access

Data Table and Database

This Interface expansion	Includes the following additional objects for System Test	with this protection
Default	Fully expanded data table for the transformed interface type.	Read
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables	Delete
Default	The database and technical design for the table.	Read

TD Action Blocks, DBRMs, and Implementation Logic

This Interface expansion	Includes the following additional objects for System Test	with this protection
Default	Fully expanded TD action blocks for entity types, data tables.	Modify
Default	DBRM and implementation logic units for each TD action block.	Modify

Entity Type used in the Views of these Action Blocks

This Interface expansion	Includes the following additional objects for System Test	with this protection
Default	All entity types with any kind of relationship to the used entity type or one of its neighbors, that is, its full chain of relationships, not just identifying relationships.	Read
Short Only	A chain of shared companion entity types that have identifying relationships to the scoped interface type, its subtypes, or its shared companion entity types. The shared companion entity types include their attributes and permitted values and additional layers of shared companions based on identifiers of the first set of shared companion entity types found for the scoped interface type.	Access
Default Only	All shared companion entity types. The shared companion entity types include their attributes and permitted values. The first shared companion layer is every entity type with a relationship to the scoped interface type. The second layer of shared companions is based on identifiers of the first level of shared companion entity types found.	Access

Transactions

This Interface expansion	Includes the following additional objects for System Test	with this protection
Default	Owned transactions	Access
Default Only	Any mutually exclusive relationship set of the scoped interface type.	Modify

Interface Type Model

This Interface expansion	Includes the following additional objects for System Test	with this protection
Short, Default, Full	Interface type model scoped by interface type	Modify

Interface Type Model

The following tables show interface type model design, unit test, and system test:

Art Objects

This Interface Type Model expansion	Includes the following additional objects	with this protection if scoped with Modify
Default, Full	Art objects	Delete

Entity Types

This Interface Type Model expansion	Includes the following additional objects	with this protection if scoped with Modify
Default	Referenced entity types	Access
Full	Referenced entity types	Modify
Default	Referenced relationships	Access
Full	Referenced relationships	Modify
Default	Referenced subtypes	Access
Full	Referenced subtypes	Modify
Default	Referenced partitionings	Access
Full	Referenced partitionings	Modify
Default	Parent entity types of referenced relationships	Access
Full	Parent entity types of referenced relationships	Modify
Default	Parent entity types of referenced subtypes	Access

This Interface Type Model expansion	Includes the following additional objects	with this protection if scoped with Modify
Full	Parent entity types of referenced subtypes	Modify
Default	Parent entity types of referenced partitionings	Access
Full	Parent entity types of referenced partitionings	Modify

Interface Type

This Interface Type Model expansion	Includes the following additional objects	with this protection if scoped with Modify
Default	Scoping Interface Type	Access
Full	Scoping Interface Type	Modify

Matrix

To create or maintain matrices with the Matrices tool, use the matrix scoping object. A matrix tables relationship between key types of objects in a model.

Two types of matrices are provided as scoping objects:

- System-supplied
- User-defined

The system-supplied matrices are predefined pairings of types of objects. These matrices are related to Planning, Analysis, and Design tasks.

The user-defined matrices consist of three possible combinations:

- The user-defined objects for both axes
- A user-defined object for one axis and a system-supplied object for the other axis
- The system-supplied objects for both X and Y axes

When to Use System-Supplied

Include a system-supplied matrix in a subset to:

- Add the objects to the matrix

- Add, or change, or delete cell values
- Describe the matrix in a text description
- Detail objects that are used in rows and columns
- Remove the objects from a matrix
- Perform cluster function on:
  - Business Function/Business Function
  - Entity Type/Entity Type

When to Use User-Defined

Include a user-defined matrix in a subset to:

- Do all the tasks described earlier *except* the cluster function to an existing matrix
- Create a matrix

Rules for System-Supplied

- You must scope a matrix to include it in your subset.
- You cannot delete a system-supplied matrix. You can only delete the usage of specific objects in the matrix.
- A matrix using business function as an axis displays only the lowest level (leaf) function. However, a subset that is created for this matrix contains every function and process in the model.
- A matrix using elementary process as an axis displays only the elementary processes. A subset that is created for this matrix contains elementary processes plus the processes and functions that are in their direct path to the root.
- When used on an axis of a matrix, function and processes receive default-**not** Short-expansion.

Special Cases

Two system-supplied matrices have special requirements for expansion and protection. These matrices are:

- Business Function/Entity Type (Planning-related)
- Elementary Process/Entity Type (Analysis-related)

These two matrices use expected effects of functions or processes that have been detailed using an AHD or ADD. They also allow you to define or change expected effects using the matrix. For this reason, functions, processes, and entity types that are used in these two matrices receive different treatment:

- Functions and processes receive Modify protection, not Access.

- Entity types receive default expansion, not short expansion.

Because of their different treatment, these matrices can be large, and the rules and limitations for Modify protection apply to the functions and processes in them.

Additional Requirements

The two matrices that are mentioned in Special Cases have some special requirements when using certain features of the Matrix Processor Tool:

- If an axis contains more than 256 objects, you cannot cluster, sort, and so forth.
- To use the cluster function with the Business Function/Entity Type matrix, you must scope two more matrices with Delete protection:
  - Business Function/Business Function
  - Entity Type/Entity Type
- To use the cluster function with the Elementary Process/Entity Type matrix, also scope the Elementary Process/ Elementary Process matrix with Delete protection.

Expansions for User-Defined

- The basic expansion of a user-defined matrix is the same as that of a system-supplied matrix.
- The expansion of a user-defined object class that is used in a matrix includes every occurrence of that object type in the model with Access protection.

Rules for User-Defined

- You must scope a user-defined matrix to include it in your subset.
- You can delete a user-defined matrix. If the matrix uses a system-defined object class for one or both of its axes, you do not delete the object class, only their usage in the matrix.
- If the matrix consists solely of user-defined objects, everything can be deleted: matrix, objects, cell values, and so forth.
- A subset for a matrix using business function as an axis contains every function and process in the model.
- A subset for a matrix using elementary process as an axis displays only the elementary processes. A subset that is created for this matrix contains elementary processes plus the processes and functions that are in their direct path to the root.
- When used on an axis of a matrix, function and processes receive default expansion, not Short expansion.

The following diagram shows a matrix.

	Process 1	Process 2	Process 3	Process 4	Process 5	Process 6	Process 7	Process 8	Process 9	Process 10	Process 11	Process 12	Process 13
Entity Type 1		X											
Entity Type 2	X												
Entity Type 3					X								
Entity Type 4													
Entity Type 5													
Entity Type 6						X							
Entity Type 7													

The following tables show matrix design, unit test, and system test:

Matrix Usages, Cell Values

This Matrix expansion	Includes the following additional objects	with this protection
Default	The scoped matrix object.	Modify
Default	Matrix cell values.	Delete

Object Classes and Object Class Occurrences

This Matrix expansion	Includes the following additional objects	with this protection
Default	The object classes used on the X and Y axis. See Class expansions for individual charts of system-supplied object classes.	Access
Default	All occurrences of each object class.	Access

Navigation Diagram

To maintain the part of the model that is related to the windows and dialog boxes of the subset, use the navigation diagram scoping object. Use the navigation diagram as the central hub for creating and maintaining interactions between application components.

Include the navigation diagram in a subset definition to:

- Change the interaction between a window and dialog boxes.
- Access procedure steps that are used by the navigation diagram.

The following table shows navigation diagram design, unit test, and system test:

This Navigation Diagram expansion	Includes the following additional objects	with this protection if scoped with Modify
Default, Full	Window usages and procedure step window usages	Delete
Default	Procedure steps owning windows/dialog boxes used by the navigation diagram	Access
Full	Procedure steps owning windows/dialog boxes used by the navigation diagram	Modify
Default	Procedure steps directly used by navigation diagram	Access
Full	Procedure steps directly used by navigation diagram	Modify

Online Load Module

To check out all procedure steps that are contained in the scoped load module, Use the online load module scoping object.

Scope an online load module to perform code generation tasks.

The expansion level and protection option of the scoped online load module step are applied to each procedure step within that load module. All packaging is uploaded to the Host Encyclopedia.

Note: The online load module has the same five levels of expansion as procedure steps: Short, A, B, Default, and Full. The expansion level and protection option of the scoped online load module apply to each procedure step within that load module. Items that are marked if appropriate (such as screens) are included only if included for a procedure step at the expansion you select.

Unit Test

The following tables show online load module unit test:

Program

This Online Load Module expansion	Include the following additional objects	with this protection
All (Short, A, B, Default, Full)	The scoped online program.	Modify

Packaged Procedure Steps

This Online Load Module expansion	Include the following additional objects	with this protection
All	All procedure steps packaged in the scoped online program.	Modify
All	The procedure step execution units for each step in the program.	Delete
All	Package list entry	Delete

Trancodes

This Online Load Module expansion	Include the following additional objects	with this protection
All	The trancodes for the scoped online program.	Access
All	The business systems containing these procedures.	Access

System Defaults

This Online Load Module expansion	Include the following additional objects	with this protection
All	System default information (PF keys, commands, exit states, templates, video attributes, and edit patterns).	Modify

Screen and Windows

This Online Load Module expansion	Include the following additional objects	with this protection
All	All screen and screen objects (templates, prompts, variables) for the procedure steps of the scoped online program, if appropriate.	Modify
All	All windows and window objects (prompts, variables, literals) for the procedure steps of the scoped online program, if appropriate.	Delete

Action Blocks

This Online Load Module expansion	Include the following additional objects	with this protection
All	The fully expanded action blocks and USEd action blocks of all procedure steps for the scoped online program, if appropriate.	

Referenced and Shared Companion Entity Types

This Online Load Module expansion	Include the following additional objects	with this protection
All	All entity types referenced in any action block views.	
	Database and Data Tables	

This Online Load Module expansion	Include the following additional objects	with this protection
All	Fully expanded data table for any transformed entity type used in the view of the procedure step or other USEd action blocks.	Read
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables	Delete
All	Database and technical design for the data tables.	Read

TD Action Blocks, DBRMs, and Implementation Logic

This Online Load Module expansion	Include the following additional objects	with this protection
All	Unexpanded TD action blocks for every entity and table in the subset.	Read
All	Implementation logic units and DBRMs for every TD action block.	Read

System Test

The following tables show online load module system test:

Program

This Online Load Module expansion	Includes the following additional objects for System Test	with this protection
All	The scoped online program.	Modify

Packaged Procedure Steps

This Online Load Module expansion	Includes the following additional objects for System Test	with this protection
All	All procedure steps packaged in the scoped online program.	Modify

This Online Load Module expansion	Includes the following additional objects for System Test	with this protection
All	The procedure step execution units for each step in the program.	Delete
All	Package list entry	Delete

Trancodes

This Online Load Module expansion	Includes the following additional objects for System Test	with this protection
All	Clear screen and Dialog Flow trancodes for all the procedure steps.	Modify
All	Trancodes for the scoped online program.	Access

Procedures and Business Systems

This Online Load Module expansion	Includes the following additional objects for System Test	with this protection
All	The procedures for the procedure steps of the program.	Access
All	The business systems containing these procedures.	Access

System Defaults

This Online Load Module expansion	Includes the following additional objects for System Test	with this protection
All	System default information (PF keys, commands, exit states, templates, video attributes, and edit patterns.	Access

Screen and Windows

This Online Load Module expansion	Includes the following additional objects for System Test	with this protection
All	All screens and screen objects (templates, prompts, variables) for the procedure steps of the scoped online program, if appropriate.	Modify
All	All windows and window objects (prompts, variables, literals) for the procedure steps of the scoped online program, if appropriate.	Delete

Action Blocks, Implementation Logic, DBRMs

This Online Load Module expansion	Includes the following additional objects for System Test	with this protection
All	The fully expanded action blocks and USEd action blocks of the procedure steps of the online load module, if appropriate.	Modify
All	The implementation logic objects for these action blocks.	Delete

Referenced and Shared Companion Entity Types

This Online Load Module expansion	Includes the following additional objects for System Test	with this protection
All	Expanded treatment of entity shared companions for any entity type used in views of action blocks.	Read

Data Tables and Databases

This Online Load Module expansion	Includes the following additional objects for System Test	with this protection
All	Fully expanded data table for any transformed entity type used in the view of the procedure step or other USEd action blocks.	Read

This Online Load Module expansion	Includes the following additional objects for System Test	with this protection
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables	Delete
All	Database and technical design for the data tables	Read

TD Action Blocks, DBRMs, and Implementation Logic

This Online Load Module expansion	Includes the following additional objects for System Test	with this protection
All	Fully expanded TD action blocks for each entity type.	
All	Implementation logic units and DBRMs for every TD action block.	Read

Dialog Manager

This Online Load Module expansion	Includes the following additional objects for System Test	with this protection
All	A Dialog Manager between load modules that includes the following: ■ Flows into or out of each expanded load module in the subset. ■ Procedure steps for these flows with short expansion. ■ Entities and attributes used in the views of these steps. ■ Data tables for the entities, fully expanded. ■ Extended tables. ■ Tablespace referenced by extended tables. ■ Databases for the tables fully expanded. ■ Unexpanded TD action block for the entities and tables. ■ Implementation logic unit and DBRM for the TD action blocks. ■ Trancodes for the flowed to load modules. ■ Unexpanded load modules being flowed to. ■ Implementation units for the load module only.	Read

Operations Library

The following table shows operations library design, unit test, and system test:

This Operations Library expansion	Includes the following additional objects for all subset types	with this protection
All	Technical System	Access

This Operations Library expansion	Includes the following additional objects for all subset types	with this protection
All	Implementation units contained by Operations Library	Modify
All	Action block associated to each Implementation unit	Modify

Organizational Unit

To maintain organization units in the Organization Hierarchy Diagram or matrices, use the organizational unit scoping object.

To add, modify, or delete organization units within a hierarchy, include an organization unit in a subset.

If a root organization exists in a model, it is included in the subset as part of the default set, if not previously included through the expansion of some other object.

The following tables show organizational unit design, unit test, and system test:

Organization Units

This Org Unit expansion	Includes the following additional objects	with this protection
Default	The scoped organization unit.	Modify
Default	All organization units directly subordinate to the scoped unit.	Modify
Default	All parent organization units of the scoped unit.	Modify

Matrix Usages and Cell Values

This Procedure Step expansion	Includes the following additional objects	with this protection
Default, if scoped with Delete	Any matrix usages.	Access
Default, if scoped with Delete	Any matrix cell values.	Access

Procedure

To maintain the following items that are related to the scoped procedure with the Procedure Action Diagram, Screen Design, and Dialog Flow Diagram, and Window Design, use the procedure scoping object:

- Action diagrams
- BSD (Design) action blocks
- Screens
- Dialog flows

When to Use

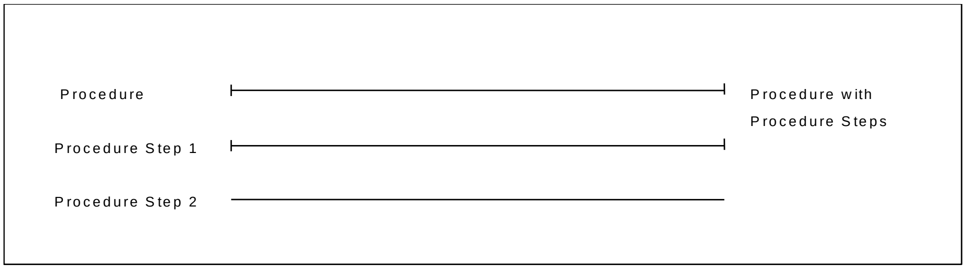
Include a procedure in the subset definition to modify:

- The Procedure Action Diagram and screens for all the procedure steps in the procedure.
- The Dialog Flow Diagram for the scoped procedure. Scope two procedure steps with short expansion to modify the Dialog Flow Diagram between the two procedure steps.

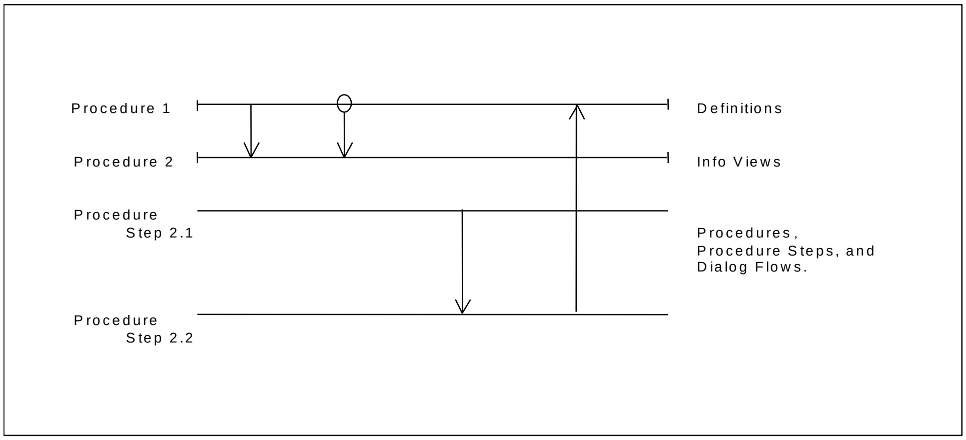
When Not to Use

In many cases during Design, a subset definition contains only one scoping object, the procedure step to be designed. Whenever possible, scope one or more procedure steps rather than scoping a procedure. Scoping procedure steps helps avoid the problems that large procedure-related subsets can entail.

The following diagram shows procedure and procedure steps.



The following diagram shows dialog flows with procedure and procedure steps.



Design

The following tables show procedure design:

Procedure Steps, Action Blocks, Screens, Windows and Dialog Flow

This Procedure expansion	Includes the following additional objects for Design	with this protection
Short, A, Default, Full	The procedure steps of the scoped procedure.	Modify
Short, A, Default, Full	The full expansion of the action blocks and screens that belong to the procedure steps of the scoped procedure.	Modify
Short, A, Default, Full	Fully expanded windows and dialog boxes associated with the procedure steps of the scoped procedure.	Delete
Short, A, Default, Full	Dialog flow information between the procedure steps of the scoped procedure.	Delete
Short, A, Default, Full	Bilingual text for screen objects.	Delete

Views

This Procedure Step expansion	Includes the following additional objects	with this protection
Short, A, Default, Full	All views associated with the procedure and its procedure steps	Modify

System Defaults

This Procedure Step expansion	Includes the following additional objects	with this protection
Short, A, Default, Full	System defaults (PF keys, commands, exit states, templates, and video attributes).	Access
Short, A, Default, Full	All bilingual text for the system defaults.	Read

Dialog Flow

This Procedure Step expansion	Includes the following additional objects	with this protection
Default Only	All procedures and procedure steps that flow to or from the scoping procedure (its shared companions).	Access
Default Only	Dialog flow information used by the procedure steps that flow from the steps of the scoped procedure.	Delete
Default Only	Dialog flow information used by the procedure steps that flow to the steps of the scoped procedure.	Read
Default Only	Entity types referenced by the information views in Procedure Action Diagrams.	Access
Default Only	You can add or modify views that reference the entity types, but you cannot add or change entity type components without scoping entity types.	

Elementary Processes

This Procedure Step expansion	Includes the following ad	with this protection
A or Full	Elementary Processes implemented by the scoped procedure.	Modify

Shared Companion Elementary Processes, Action Blocks, and Screens

This Procedure Step expansion	Includes the following ad	with this protection
Full	Elementary processes implemented by shared companion procedures of the scoped procedure.	Access
Full	Action block and screen definition objects of procedure steps that flow to or from the scoped procedure.	Access
Full	Window definition objects of any procedure steps that flow to or from the scoped procedure.	Read
Full	Bilingual text for shared companion screen objects.	Read

Unit Test

The following tables show procedure unit test:

System Defaults

This Procedure expansion	Includes the following additional objects for Unit Test	with this protection
All	System Defaults (PF keys, commands, exit states, templates, and video attributes).	Access

Action Blocks

This Procedure expansion	Includes the following additional objects for Unit Test	with this protection
All	The fully expanded action blocks and USEd action blocks of all procedure steps for the scoped procedure.	Modify

Screens and Windows

This Procedure expansion	Includes the following additional objects for Unit Test	with this protection
All	All screens and screen objects (templates, prompts, variables).	Modify
All	All windows, window objects (prompts, variables, literals), and HTML objects.	Delete

Referenced and Shared Companion Entity Types

This Procedure expansion	Includes the following additional objects for Unit Test	with this protection
All	All entity types referenced in any action block views.	Access

Database, Data Tables, and TD Action Blocks

This Procedure expansion	Includes the following additional objects for Unit Test	with this protection
All	Fully expanded data tables for any transformed entity types used the views of the scoped procedure.	Read
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables	Delete
All	Database and technical design of the data tables.	Read
All	TD action blocks for the entities and data tables.	Read

This Procedure expansion	Includes the following additional objects for Unit Test	with this protection
All	Implementation logic units for the TD action blocks.	Read
All	DBRMs for the TD action blocks.	Read

Load Module, Execution Units, and Packaging Units

This Procedure expansion	Includes the following additional objects for Unit Test	with this protection
See note 1	Unexpanded load module or batch job for the procedure steps of the scoped procedure.	Access
See note 2	Unexpanded load module or batch job for the procedure steps of the scoped procedure.	Read
See note 1	Procedure step execution units and packaging units for all procedure steps of the scoped procedure.	Delete
See note 2	Procedure step execution units and packaging units for all procedure steps of the scoped procedure.	Read
	Note: 1. Included for All, if you scope the procedure with Modify or Delete protection. 2. Included for All, if you scope the procedure with Access or Read protection.	

Trancodes

This Procedure expansion	Includes the following additional objects for Unit Test	with this protection
See note	Clear screen and Dialog Flow trancodes for the procedure steps of the scoped procedure. **Note:** Included for All, if you scope the procedure with Modify or Delete protection.	Access

This Procedure expansion	Includes the following additional objects for Unit Test	with this protection
See note	Clear screen and Dialog Flow trancodes for the procedure steps of the scoped procedure. **Note:** Included for All, if you scope the procedure with Access or Read protection.	Read

System Test

The following tables show procedure system test:

System Defaults

This Procedure expansion	Includes the following additional objects for System Test	with this protection
All	System defaults (PF keys, commands, exit states, templates, and video attributes).	Access

Procedure Steps and Action Blocks

This Procedure expansion	Includes the following additional objects for System Test	with this protection
All	Fully expanded procedure steps and USEd action blocks.	Modify

Screen and Windows

This Procedure expansion	Includes the following additional objects for System Test	with this protection
All	The screen and screen objects (templates, prompts, variables) for all procedure steps of the scoped procedure.	Modify
All	All windows, window objects (prompts, variables, literals), and HTML objects for all procedure steps of the scoped procedure.	Delete

Referenced and Shared Companion Entity Types

This Procedure expansion	Includes the following additional objects for System Test	with this protection
All	All entity types referenced in any action block views with default expansion rather than short.	Access
All	A broader definition of shared companions for all entities directly referenced in views.	Read

Database and Data Tables

This Procedure expansion	Includes the following additional objects for System Test	with this protection
All	Fully expanded data tables for any transformed entity types used in the views of the scoped procedure.	Read
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables	Delete
All	Database and technical design for the data tables.	Read

Load Module, Execution Units, and Packaging Units

This Procedure expansion	Includes the following additional objects for System Test	with this protection
All	The fully expanded load module or batch job for the procedure steps of the scoped procedure.	Read
All	Full expansion of other procedure steps in each load module or job.	Read
All	Procedure step execution units and packaging units for every procedure step packaged in each load module or job being expanded.	Read

Trancodes

This Procedure expansion	Includes the following additional objects for System Test	with this protection
All	Clear screen and Dialog Flow trancodes for all steps in the subset.	Read

Implementation Logic and DBRMS

This Procedure expansion	Includes the following additional objects for System Test	with this protection
All	Implementation logic objects for every action block in the subset.	Read
All	DBRMs for every implementation logic object in the subset.	Read

TD Action Blocks

This Procedure expansion	Includes the following additional objects for System Test	with this protection
All	Full expansion of TD action blocks for every entity and table.	Read
All	Implementation logic units and DBRMs for every TD action block.	Read

Dialog Manager

This Procedure expansion	Includes the following additional objects for System Test	with this protection
All	A Dialog Manager between load modules that includes the following: ■ Flows into or out of each expanded load module in the subset. ■ Procedure steps for these flows with Short expansion. ■ Entities and attributes used in the views of these steps. ■ TD action blocks for each of these entities ■ Implementation units and DBRMs for the TD action blocks. ■ Trancodes for the flowed to load modules. ■ Unexpanded load modules being flowed to. ■ Implementation units for the load module only.	Read

Procedure Step

To maintain the screens, windows, and action blocks that are defined for the procedure steps of the subset with the Screen Design and Procedure Action Diagram, use the procedure step scoping object.

When to Use

Scope individual procedure steps to:

- Modify the step's Procedure Action Diagrams and screen or windows
- Maintain the action blocks referenced by the step

Choosing the Procedure Step Expansion

The expansion of the procedure step affects only the expansion level of the action block. The following table shows guidelines for expanding a procedure step.

To	Use this expansion	At this protection
Delete lines in an action block or fields on a screen but not delete the screen or action block itself.	Short	Modify
Match views with common (USEd) action blocks but do not need the actual action statements of the USEd action blocks.	Default	Access
Perform maintenance on screens.	Short	Modify
Expand USEd action blocks fully as well as the step's action block.	A	Access
Fully expand only the procedure step's action block.	B	Access

Design

The following tables show procedure step design:

Procedure Step and Parent

This Procedure Step expansion	Includes the following additional objects for Design	with this protection
All (Short, B, A, Default, Full)	The scoped procedure step.	Modify
All	The parent procedure of the scoped procedure step.	Access

Referenced Entity Types

This Procedure Step expansion	Includes the following additional objects	with this protection
Short	All entity types referenced in views of the procedure block or USEd action books. Unless explicitly scoped, the entity types referenced in view for procedure steps receive short expansion. Short expansion provides all attributes, partitionings, subtypes, identifiers, and identifying relationships of the scoped entity types and of their shared companion entity types.	Access

System Defaults

This Procedure Step expansion	Includes the following additional objects	with this protection
All	System defaults (PF keys, commands, exit states, templates, and video attributes).	Access
All	All bilingual text for system defaults.	Read

Load Module, Execution Units, and Packaging Units

This Procedure Step expansion	Includes the following additional objects	with this protection
See note	Unexpanded load module for the scoped procedure step. **Note:** Included for All, if you scope the procedure step with Modify or Delete protection.	Access
See note	Procedure step execution unit and packaging unit for the scoped procedure step. **Note:** Included for All, if you scope the procedure step with Modify or Delete protection.	Delete

Trancodes and Transactions

This Procedure Step expansion	Includes the following additional objects	with this protection
See note	Clear screen and Dialog Flow trancodes for the scoped procedure step. **Note:** Included for All, if you scope the procedure step with Modify or Delete protection.	Access
Default, Full	Supported transaction operations.	Modify

Fully Expanded Action Block, Windows and Dialog Boxes

This Procedure Step expansion	Includes the following additional objects	with this protection
B, A, Default, Full	Fully expanded action block of the procedure step (action statements and views).	Modify
B, A, Default, Full	Fully expanded windows (with all window objects) and dialog boxes.	Delete

Unexpanded Action Block, Windows and Dialog Boxes

This Procedure Step expansion	Includes the following additional objects	with this protection
Short Only	Unexpanded action block of the procedure step (views only).	Access
Short Only	Unexpanded (empty) windows and dialog boxes.	Access

Fully expanded USEd Action Blocks

This Procedure Step expansion	Includes the following additional objects	with this protection
A or Full	Fully expanded action blocks (action statements and views) USEd by the action block of the scoped procedure step.	Modify

Web Operation

This Procedure Step expansion	Include the following additional object types for all subset types	with this protection
All	Web Operations defined for scoped Procedure Step	Modify

Unit Test

The following tables show procedure step unit test:

System Defaults

This Procedure Step expansion	Includes the following additional objects for Unit Test	with this protection
All (Short, B, A, Default, Full)	System defaults (PF keys, commands, exit states, templates, and video attributes).	Access

Screen and Windows

This Procedure Step expansion	Includes the following additional objects for Unit Test	with this protection
All	The screen and screen objects (templates, prompts, variables).	Modify
All	All windows, window objects (prompts, variables, literals), and HTML objects.	Delete

Action Block

This Procedure Step expansion	Includes the following additional objects for Unit Test	with this protection
See note	The appropriate expansion of action block. **Note:** The expansion of the action block depends on the expansion chosen for the scoped procedure step.	Modify

Referenced and Shared Companion Entity Types

This Procedure Step expansion	Includes the following additional objects for Unit Test	with this protection
All	All entity types referenced in any action block views.	Access

Database and Data Tables

This Procedure Step expansion	Includes the following additional objects for Unit Test	with this protection
All	Database and technical design for the data tables.	Read
All	Fully expanded data tables for any transformed entity types used in the view of the scoped procedure step.	Read
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables	Delete

Load Module, Execution Units, and Packaging Units

This Procedure Step expansion	Includes the following additional objects for Unit Test	with this protection
See note 1	Unexpanded load module for the scoped procedure step.	Access
See note 2	Unexpanded load module for the scoped procedure step.	Read
See note 1	Procedure step execution unit and packaging unit for the scoped procedure step.	Delete
See note 2	Procedure step execution unit and packaging unit for the scoped procedure step.	Read
	Note: ■ 1. Included for All, if you scope the procedure step with Modify or Delete protection. ■ 2. Included for All, if you scope the procedure step with Access or Read protection.	

Trancodes and Transactions

This Procedure Step expansion	Includes the following additional objects for Unit Test	with this protection
Default, Full	Supported transaction operations.	Modify
See note	Clear screen and Dialog Flow trancodes for the scoped procedure step. **Note:** Included for All, if you scope the procedure step with Modify or Delete protection.	Access
See note	Clear screen and Dialog Flow Trancodes for the scoped procedure step. **Note:** Included for All, if you scope the procedure step with Access or Read protection.	Read

Web Operation

This Procedure Step expansion	Include the following additional object types for all subset types	with this protection
All	Web Operations defined for scoped Procedure Step	Modify

System Test

The following tables show procedure step system test:

System Defaults

This Procedure Step expansion	Includes the following additional objects for System Test	with this protection
All	System defaults (PF keys, commands, exit states, templates, video attributes, and edit patterns).	Access

Screen and Windows

This Procedure Step expansion	Includes the following additional objects for System Test	with this protection
All	The screen and screen objects (templates, prompts, variables).	Modify
All	All windows, window objects (prompts, variables, literals), and HTML objects.	Delete
All	All dialog boxes.	Delete

Action Block

This Procedure Step expansion	Includes the following additional objects for System Test	with this protection
See note	The appropriate expansion of action block. **Note:** The expansion of the action block depends on the expansion chosen for the scoped procedure step.	Modify

Referenced and Shared Companion Entity Types

This Procedure Step expansion	Includes the following additional objects for System Test	with this protection
All	All entity types referenced in any action block views with default expansion rather than short.	Access
All	A broader definition of shared companion for entity types in views.	Read

Database and Data Tables

This Procedure Step expansion	Includes the following additional objects for System Test	with this protection
All	Database and technical design for the data tables.	Read

This Procedure Step expansion	Includes the following additional objects for System Test	with this protection
All	Fully expanded data tables for any transformed entity types used in the views of the scoped procedure step.	Read
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables	Delete

Load Module, Execution Units, and Packaging Units

This Procedure Step expansion	Includes the following additional objects for System Test	with this protection
All	The fully expanded load module for the scoped procedure step.	Read
All	Full expansion of other procedure steps in the load module.	Read
All	Procedure step execution units and packaging units for every procedure step packaged in the load module being expanded.	Read

Trancodes and Transactions

This Procedure Step expansion	Includes the following additional objects for System Test	with this protection
All	Clear screen and Dialog Flow trancodes for all steps in the load module being expanded.	Read
Default, Full	Supported transaction operations.	Modify

Implementation Logic and DBRMS

This Procedure Step expansion	Includes the following additional objects for System Test	with this protection
All	Implementation logic objects for every action block in the subset.	Read

This Procedure Step expansion	Includes the following additional objects for System Test	with this protection
All	DBRMs for every implementation logic object in the subset.	Read

Dialog Manager

This Procedure Step expansion	Includes the following additional objects for System Test	with this protection
All	A Dialog Manager between load modules that includes the following: ■ Flows into or out of each expanded load module in the subset. ■ Procedure steps for these flows with Short expansion. ■ Entities and attributes used in the views of these steps. ■ Trancodes for the flowed to load modules. ■ Unexpanded load modules being flowed to.	Read

Web Operation

This Procedure Step expansion	Include the following additional object types for all subset types	with this protection
All	Web Operations defined for scoped Procedure Step	Modify

Process

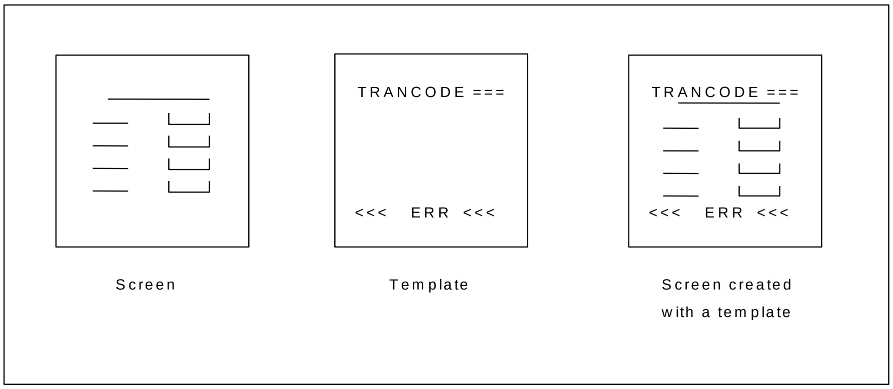
See the Function and Process section.

Screen

To maintain a specific screen with the Screen Design tool, use the screen scoping object.

To change the existing screen, include a scroll amount value in the subset definition.

The following diagram shows screen and template:



The following tables show screen design, unit test, and system test:

Screen, Screen Template, and Bilingual Text

This Screen expansion	Includes the following additional objects for all subset types	with this protection
Default	The scoped screen.	Modify
Default	The template for the scoped screen.	Modify
Default	A bilingual text for screen items and template items (literals, prompts, edit patterns).	Delete

Procedure Step, Action Block, Windows and Procedure

This Screen expansion	Includes the following additional objects for all subset types	with this protection
Default	Procedure step for the scoped screen.	Modify
Default	The non-expanded action block for the procedure step (views only).	Access
Default	The procedure for the procedure step.	Access
Default	All windows for the procedure step, if they exist.	Delete

Referenced Entity Types and Parent Subject Areas

This Screen expansion	Includes the following additional objects for all subset types	with this protection
Default	Entity types referenced in action block views.	Access
Default	Parent subject area of referenced entity types.	Access

System Defaults

This Screen expansion	Includes the following additional objects for all subset types	with this protection
Default	System defaults (PF keys, commands, exit states, templates, video attributes, prompts, and edit patterns).	Access
Default	All bilingual text for system defaults.	Read

Scroll Amount Value

To maintain scroll amount values with the Design toolset, use the scroll amount value scoping object.

Include a scroll amount value in the subset definition to:

- Create, modify, or delete bilingual text for scroll amount values
- Modify the default scroll amount value

Every model contains five possible scroll amount values:

- Page
- Half
- Cursor
- Max
- Locate

The default scroll amount value is cursor. If you do not scope scroll amount values directly, they are added to your subset as part of the default set of objects.

The following table shows scroll amount value design, unit test, and system test:

Scroll Amount Value and Bilingual Text

This Scroll Amount Value expansion	Includes the following additional objects for all subset types	with this protection
Default	The scoped scroll amount value.	Modify
Default	Any bilingual text for the scroll amount values.	Delete

Server Manager

To check out all procedure steps that are contained in the scoped load module, use the server manager scoping object.

To perform code generation tasks, scope a server manager.

The expansion level and protection option of the scoped server manager step are applied to each procedure step within that load module.

Note: The server manager has the same five levels of expansion as procedure steps: Short, A, B, Default, and Full. The expansion level and protection option of the scoped server manager apply to each procedure step within that load module. Items that are marked if appropriate (such as screens) are included only if included for a procedure step at the expansion you select.

Unit Test

The following tables show server manager unit test:

Program

This Server Manager expansion	Include the following additional objects for Unit Test	with this protection
All (Short, A, B, Default, Full)	The scoped server manager.	Modify

Packaged Procedure Steps

This Server Manager expansion	Include the following additional objects for Unit Test	with this protection
All	All procedure steps packaged in the scoped server manager.	Modify
All	The procedure step execution units for each step in the program.	Delete
All	Package list entry	Delete

Trancodes

This Server Manager expansion	Include the following additional objects for Unit Test	with this protection
All	The trancodes for the scoped server manager.	Access
All	The business systems containing these procedures.	Access

System Defaults

This Server Manager expansion	Include the following additional objects for Unit Test	with this protection
All	System default information (PF keys, commands, exit states, templates, video attributes, and edit patterns).	Modify

Screen and Windows

This Server Manager expansion	Include the following additional objects for Unit Test	with this protection
All	All screen and screen objects (templates, prompts, variables) for the procedure steps of the scoped server manager, if appropriate.	Modify
All	All windows and window objects (prompts, variables, literals) for the procedure steps of the scoped server manager, if appropriate.	Delete

Action Blocks

This Server Manager expansion	Include the following additional objects for Unit Test	with this protection
All	The fully expanded action blocks and USEd action blocks of all procedure steps for the scoped server manager, if appropriate.	

Referenced and Shared Companion Entity Types

This Server Manager expansion	Include the following additional objects for Unit Test	with this protection
All	All entity types referenced in any action block views.	

Database and Data Tables

This Server Manager expansion	Include the following additional objects for Unit Test	with this protection
All	Fully expanded data table for any transformed entity type used in the view of the procedure step or other USEd action blocks.	Read
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables	Delete
All	Database and technical design for the data tables.	Read

TD Action Blocks, DBRMs, and Implementation Logic

This Server Manager expansion	Include the following additional objects for Unit Test	with this protection
All	Unexpanded TD action blocks for every entity and table in the subset.	Read
All	Implementation logic units and DBRMs for every TD action block.	Read

System Test

The following tables show server manager system test:

Program

This Server Manager expansion	Includes the following additional objects for System Test	with this protection
All	The scoped server manager.	Modify

Packaged Procedure Steps

This Server Manager expansion	Includes the following additional objects for System Test	with this protection
All	All procedure steps packaged in the scoped server manager.	Modify
All	The procedure step execution units for each step in the program.	Delete
All	Package list entry	Delete

Trancodes

This Server Manager expansion	Includes the following additional objects for System Test	with this protection
All	Clear screen and Dialog Flow trancodes for all the procedure steps.	Modify
All	Trancodes for the scoped server manager.	Access

Procedures and Business Systems

This Server Manager expansion	Includes the following additional objects for System Test	with this protection
All	The procedures for the procedure steps of the program.	Access
All	The business systems containing these procedures.	Access

System Defaults

This Server Manager expansion	Includes the following additional objects for System Test	with this protection
All	System default information (PF keys, commands, exit states, templates, video attributes, and edit patterns.	Access

Screen and Windows

This Server Manager expansion	Includes the following additional objects for System Test	with this protection
All	All screens and screen objects (templates, prompts, variables) for the procedure steps of the scoped server manager, if appropriate.	Modify
All	All windows and window objects (prompts, variables, literals) for the procedure steps of the scoped server manager, if appropriate.	Delete

Action Blocks, Implementation Logic, DBRMs

This Server Manager expansion	Includes the following additional objects for System Test	with this protection
All	The fully expanded action blocks and USEd action blocks of the procedure steps of the server manager, if appropriate.	Modify
All	The implementation logic objects for these action blocks.	Delete

Referenced and Shared Companion Entity Types

This Server Manager expansion	Includes the following additional objects for System Test	with this protection
All	Expanded treatment of entity shared companions for any entity type used in views of action blocks.	Read

Data Tables and Databases

This Server Manager expansion	Includes the following additional objects for System Test	with this protection
All	Fully expanded data table for any transformed entity type used in the view of the procedure step or other USEd action blocks.	Read

This Server Manager expansion	Includes the following additional objects for System Test	with this protection
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables	Delete
All	Database and technical design for the data tables	Read

TD Action Blocks, DBRMs, and Implementation Logic

This Server Manager expansion	Includes the following additional objects for System Test	with this protection
All	Fully expanded TD action blocks for each entity type.	
All	Implementation logic units and DBRMs for every TD action block.	Read

Dialog Manager

This Server Manager expansion	Includes the following additional objects for System Test	with this protection
All	A Dialog Manager between load modules that includes the following: ■ Flows into or out of each expanded load module in the subset ■ Procedure steps for these flows with short expansion ■ Entities and attributes used in the views of these steps ■ Data tables for the entities, fully expanded ■ Extended tables ■ Tablespace referenced by extended tables ■ Databases for the tables, fully expanded ■ Unexpanded TD action block for the entities and tables ■ Implementation logic unit and DBRM for the TD action blocks ■ Trancodes for the flowed to load modules ■ Unexpanded load modules being flowed to Implementation units for the load module only	Read

Specification Type

Design

The following tables show specification type design:

Entity Type, Entity Subtype, Details, Relationships

This Specification Type expansion	Includes the following additional objects for Design	with this protection
Default, Full	Entity type or work attribute set included contents are used by	Access
Short Only	A chain of shared companion entity types that have identifying relationships to the scoped specification type, its subtypes, or its shared companion entity types. The shared companion entity types include their attributes and permitted values and additional layers of shared companions based on identifiers of the first set of shared companion entity types found for the scoped specification type. Short expansion brings in all identifying relationships down the chain of entity types and for subtypes. It includes identifying shared companion relationships, not anything else.	Access
Default, Full	All shared companion entity types. The shared companion entity types include their attributes and permitted values. Default expansion includes every relationship whether or not it is an identifying relationship.	Access
Short, Default, Full	Parent Entity of attributes from which foreign key attributes are derived	Access
Short, Default, Full	Subtypes	Modify
Short, Default, Full	Partitionings	Modify
Short, Default, Full	All attributes and foreign key attributes (including those of subtypes).	Modify

This Specification Type expansion	Includes the following additional objects for Design	with this protection
Short, Default, Full	Aliases and Identifiers.	Delete
Short, Default, Full	All permitted values, prompt for values, and prompts.	Modify
Short, Default	All bilingual text for prompts.	Delete
Default, Full	Contents	Delete
Short, Default, Full	All relationships between scoped entity types.	Modify
Short, Default, Full	Relationships for included attribute relationship usages	Access
Short, Default, Full	Attribute relationship usages	Delete
Default only	Any mutually exclusive relationship set structures of the scoped specification type.	Modify
Full	Mutually exclusives.	Delete

Parent Subject Areas

This Specification Type expansion	Includes the following additional objects for Design	with this protection
Short, Default, and Full	Parent subject areas of the scoped specification type.	Access
Short, Default	Parent subject areas of shared companion entity types.	Access

Action Blocks for Derived Attributes

This Specification Type expansion	Includes the following additional objects for Design	with this protection
Short, Default, Full	All action blocks for derived or default attributes.	Modify
Short, Default	All action blocks USEd by action blocks for derived attributes.	Modify
Default	Owned action blocks	Access

This Specification Type expansion	Includes the following additional objects for Design	with this protection
Full	Owned action blocks	Modify

Views of Action Blocks

This Specification Type expansion	Includes the following additional objects for Design	with this protection
Short, Default	All views of all action blocks in the subset.	Modify
Short, Default	All components of entity types in the views.	Access

Entity State Transitions

This Specification Type expansion	Includes the following additional objects for Design	with this protection
Short, Default, Full	Entity state transition objects associated to the entity type and its subtypes.	Delete

Data Table, Database, and Technical Design

This Specification Type expansion	Includes the following additional objects for Design	with this protection
Short, Default, Full See note	The data table implementing the scoped specification type if it has been transformed. **Note:** Included only if the specification type is scoped with Modify or Delete protection.	Read
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables	Delete
See note	The database and technical design for the table. **Note:** Included only if the specification type is scoped with Modify or Delete protection.	Delete

TD Action Blocks (for data integrity)

This Specification Type expansion	Includes the following additional objects for Design	with this protection
Short, Default, Full	The TD action block for the scoped specification type.	Modify
Short, Default	The TD action block for any shared companion entity type.	Access
Short, Default, Full	DBRM and implementation unit for TD action blocks.	Modify

Transactions

This Specification Type expansion	Includes the following additional objects for Design	with this protection
Default	Owned transactions	Access
Full	Owned transactions	Modify

Unit Test

The following tables show specification type unit test:

Entity Type, Entity Subtype, Details, Relationships

This Specification Type expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	The scoped specification type and all its partitionings including subtypes.	Modify
Short, Default	Partitionings	Modify
Default	Entity type or work attribute set included contents are used by	Access
Short, Default	All attributes (including those of subtypes).	Modify
Short, Default	All names, descriptions, properties, permitted values, and identifiers.	Modify
Short, Default	All bilingual text for prompts.	Delete
Default	Contents	Delete

This Specification Type expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	All relationships between scoped entity types.	Modify

Parent Subject Areas

This Specification Type expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	Parent subject areas of the scoped specification type.	Access
Short, Default	Parent subject areas of shared companion entity types.	Access

Action Blocks for Derived Attributes

This Specification Type expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	All action blocks for derived attributes.	Modify
Short, Default	All action blocks USEd by action blocks for derived attributes.	Modify
Default	Owned action blocks	Access

Views of Action Blocks

This Specification Type expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	All views of all action blocks in the subset.	Modify
Short, Default	All components of entity types in the views.	Access

Data Table, and Databases

This Specification Type expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	Fully expanded data table for the transformed specification type.	Read
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables	Delete
Short, Default	The database and technical design for the data table.	Read

TD Action Blocks, DBRMs and Implementation Logic

This Specification Type expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	Unexpanded TD action block for the scoped specification type and table.	Modify
Short, Default	DBRM and implementation logic units for the TD action blocks.	Modify

Entity Type Used in the Views of those Action Blocks

This Specification Type expansion	Includes the following additional objects for Unit Test	with this protection
Short Only	A chain of shared companion entity types that have identifying relationships to the scoped specification type, its subtypes, or its shared companion entity types. The shared companion entity types include their attributes and permitted values and additional layers of shared companion based on identifiers of the first set of shared companion entity types found for the scoped specification type.	Read
Default Only	All shared companion entity types. The shared companion entity types include their attributes and permitted values. The first shared companion layer is every entity type with a relationship to the scoped specification type. The second layer of shared companions is based on identifiers of the first level of shared companion entity types found.	Access

Transactions

This Specification Type expansion	Includes the following additional objects for Unit Test	with this protection
Default	Owned transactions	Access
Default Only	Any mutually exclusive relationship set of the scoped specification type.	Modify

System Test

The following tables show specification type system test:

Entity Type, Entity Subtype, Details, Relationships

This Specification Type expansion	Includes the following additional objects for System Test	with this protection
Default	The scoped specification type and all its partitionings including entity subtypes.	Modify
Short, Default	Partitionings	Modify
Default	Entity type or work attribute set included contents are used by	Access
Default	All attributes (including those of subtypes).	Modify
Default	All names, descriptions, properties, permitted values, prompts, and identifiers.	Modify
Default	All bilingual text for prompts.	Delete
Default	Contents	Delete
Default	All relationships between scoped entity types.	Modify

Parent Subject Area

This Specification Type expansion	Includes the following additional objects for System Test	with this protection
Default	Parent subject areas of the scoped specification type.	Access
Default	Parent subject areas of shared companion entity types.	Access

Action Blocks for Derived Attributes

This Specification Type expansion	Includes the following additional objects for System Test	with this protection
Default	All action blocks for derived attributes.	Modify
Default	All action blocks USEd by action blocks for derived attributes.	Modify
Default	Owned action blocks	Access

Views of Action Blocks

This Specification Type expansion	Includes the following additional objects for System Test	with this protection
Default	All views of all action blocks in the subset.	Modify
Default	All components of entity types in the views.	Access

Data Table and Database

This Specification Type expansion	Includes the following additional objects for System Test	with this protection
Default	Fully expanded data table for the transformed specification type.	Read
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables.	Delete
Default	The database and technical design for the table.	Read

TD Action Blocks, DBRMs, and Implementation Logic

This Specification Type expansion	Includes the following additional objects for System Test	with this protection
Default	Fully expanded TD action blocks for entity types, data tables	Modify
Default	DBRM and implementation logic units for each TD action block.	Modify

Entity Type used in the Views of these Action Blocks

This Specification Type expansion	Includes the following additional objects for System Test	with this protection
Default	All entity types with any kind of relationship to the used entity type or one of its neighbors, that is, its full chain of relationships, not just identifying relationships.	Read
Short Only	A chain of shared companion entity types that have identifying relationships to the scoped specification type, its subtypes, or its shared companion entity types. The shared companion entity types include their attributes and permitted values and additional layers of shared companions based on identifiers of the first set of shared companion entity types found for the scoped specification type.	Access

This Specification Type expansion	Includes the following additional objects for System Test	with this protection
Default Only	All shared companion entity types. The shared companion entity types include their attributes and permitted values. The first shared companion layer is every entity type with a relationship to the scoped specification type. The second layer of shared companions is based on identifiers of the first level of shared companion entity types found.	Access

Transactions

This Specification Type expansion	Includes the following additional objects for System Test	with this protection
Default	Owned transactions	Access
Default Only	Any mutually exclusive relationship set of the scoped specification type.	Modify

Storage Group

To create and maintain storage groups with the Technical Design tool, use the storage group scoping object.

Include a storage group in a subset definition to:

- Add new or change existing DASD volume IDs used by the storage group
- Add a storage group and assign it DASD volume IDs
- Delete an unused storage group

When a database or data table is scoped with Delete protection, the subset also includes all storage groups that are associated with the technical design object, the databases, and the data stores, which are included with Delete protection also.

The following table shows storage group design:

Storage Group, DASD Volumes, and Technical Design

This Storage Group expansion	Includes the following additional objects for Design	with this protection
Default	The scoped storage group.	Modify
Default	All DASD volumes used by the storage group.	Delete
Default	The technical design to which the storage group belongs.	Access

Subject Area

To maintain all entity types within that subject area with the Data Modeling tool, use the subject area scoping object.

Scope a subject area

- When you want to bring in all entity types for the scoped subject area and any subordinate subject areas, the full expansion of all their entity types, and the parent subject areas without expansion.
- If you plan to use the subset for transformation or retransformation into the Data Structure List.

Note: The default protection that is shown for the root subject area assumes that it is supplied by the default set of objects and not scoped individually.

Design

The following tables show subject area design:

Subject Areas and Their Entity Types

This Subject Area expansion	Includes the following additional objects for Design	with this protection
Short, Default	The root subject area of the model.	Access
Short, Default	The scoped subject area.	Modify
Short, Default	All parent subject areas of the scoped subject area without the parents' entity types.	Access
Short, Default	Any subordinate subject areas and all their entity types at default expansion.	Modify

This Subject Area expansion	Includes the following additional objects for Design	with this protection
Short, Default	Subject areas that contain the shared companion entity types.	Access
Short, Default	Parent subject areas of the shared companion subject areas.	Access
Short, Default	All entity types (default expansion) within the scoped subject area.	Modify

Shared Companion Entity Types

This Subject Area expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	A chain of shared companion entity types that have identifying relationships to an entity type in the scoped subject area or to any of its shared companion entity types. The shared companion entity types include their attributes, identifiers, partitionings, and subtypes.	Access

Data Tables and Databases

This Subject Area expansion	Includes the following additional objects for Unit Test	with this protection
See note	Fully expanded data table for any transformed entity type. **Note:** Included if the entity type within the scoped subject area receive Modify protection.	Access
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables.	Delete

This Subject Area expansion	Includes the following additional objects for Unit Test	with this protection
See note	Database and technical design for the data tables. **Note:** Included if the entity types within the scoped subject area receive Modify protection.	Access

TD Action Blocks, DBRMs, and Implementation Logic

This Subject Area expansion	Includes the following additional objects for Unit Test	with this protection
See note	Unexpanded TD action blocks, DBRMs and implementation logic units. **Note:** Included if the entity types within the scoped subject area receive Modify protection.	Access
Default Only	All shared companion entity types (default expansion) of the entity types within the scoped subject area or any of its subordinate subject areas.	Access

Unit Test

The following tables show subject area unit test:

Subject Areas and Their Entity Types

This Subject Area expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	The root subject area of the model.	Access
Short, Default	The scoped subject area.	Modify
Short, Default	All parent subject areas of the scoped subject area without the parents' entity types.	Access
Short, Default	Any subordinate subject areas and all their entity types at default expansion.	Modify
Short, Default	Subject areas that contain the shared companion entity types.	Access

This Subject Area expansion	Includes the following additional objects for Unit Test	with this protection
Short, Default	Parent subject areas of the shared companion subject areas.	Access
Short, Default	All entity types (default expansion) within the scoped subject area.	Modify

Shared Companion Entity Types

This Subject Area expansion	Includes the following additional objects for System Test	with this protection
Short Only	A chain of shared companion entity types that have identifying relationships to an entity type in the scoped subject area or to any of its shared companion entity types. The shared companion entity types include their attributes, identifiers, partitionings, and subtypes.	Access
Default Only	All shared companion entity types (default expansion) of the entity types within the scoped subject area or any of its subordinate subject areas.	Access

Data Tables and Databases

This Subject Area expansion	Includes the following additional objects for System Test	with this protection
Short, Default	Fully expanded data table for any transformed entity type.	Read
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables.	Delete
Short, Default	Database and technical design for the data tables.	Read

TD Action Blocks, DBRMs and Implementation Logic

This Subject Area expansion	Includes the following additional objects for System Test	with this protection
Short, Default	Unexpanded TD action blocks, DBRMs and implementation logic units.	Access

System Test

The following tables show subject area system test:

Subject Areas

This Subject Area expansion	Includes the following additional objects for System Test	with this protection
Short, Default	The root subject area of the model.	Access
Short, Default	The scoped subject area.	Modify
Short, Default	Any subordinate subject areas and all their entity types at default expansion.	Modify
Short, Default	All parent subject areas of the scoped subject area without the parents' entity types.	Access
Short, Default	Subject areas that contain shared companion entity types.	Access
Short, Default	Parent subject areas of shared companion subject areas.	Access

Entity Types

This Subject Area expansion	Includes the following additional objects for System Test	with this protection
Short, Default	All entity types (default expansion) within the scoped subject area.	Modify

Data Tables and Databases

This Subject Area expansion	Includes the following additional objects for System Test	with this protection
See note	Fully expanded data table for any transformed entity type that is part of the scoped subject area or its subordinates. **Note:** Included if the entity types in the scoped subject area receive Modify protection.	Read
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables	Delete
See note	Database and technical design for the data tables. **Note:** Included if the entity types in the scoped subject area receive Modify protection.	Read

Shared Companion Entity Types

This Subject Area expansion	Includes the following additional objects for System Test	with this protection
Short Only	A chain of shared companion entity types that have identifying relationships to an entity type in the scoped subject area or to any of its shared companion entity types. The shared companion entity types include their attributes, identifiers, partitionings, and subtypes.	Access
Default Only	All shared companion entity types (default expansion) within the scoped subject area or any of its subordinate subject areas.	Access

TD Action Blocks, DBRMs, and Implementation Logic

This Subject Area expansion	Includes the following additional objects for System Test	with this protection
Short, Default	Fully expanded TD action blocks for entity types, tables, and action blocks of derived attributes.	Read
Short, Default	Implementation logic for each TD action block.	Read
Short, Default	DBRM for each TD action block.	Read
Short, Default	Expanded treatment of shared companion entity types of any entity type used in the views of these action blocks.	Read

System Defined Object Class

The following table shows system defined object class design, unit test, and system test:

This System Defined Object Class expansion	Includes the following additional objects for all subset types	with this protection
Default	Aggregate objects	Access
Default	System defined objects	Modify

Technical Design Default

To maintain DBMS-specific physical data structure definitions, use the technical design default scoping object.

Include a technical design default object in the subset with Modify or Delete to be able to modify DBMS properties and default settings.

The following table shows technical design default design, unit test, and system test:

This Tech Design Default expansion	Includes the following additional objects for all subset types	w/ this protection if scoped w/ Modify or Delete
Default	Extended technical design defaults	Delete
Default	Bind package default	Delete
Default	Package list entry	Delete

Template

To maintain all screen items in the template with the Screen Design tool, use the template scoping object.

To modify or delete screen items in the template, include a template at short expansion. To delete the existing template and replace it with a new one for all referencing screens, use default expansion.

The following tables show template design, unit test, and system test:

Screen Template and Bilingual Text

This Template expansion	Includes the following additional objects for all subset types	with this protection
Short, Default	The scoped screen template.	Modify
Short, Default	All screen items in the screen template.	Delete
Short, Default	All bilingual text for screen literals, edit patterns, and prompts.	Delete

Business System and System Defaults

This Template expansion	Includes the following additional objects for all subset types	with this protection
Short, Default	Business system that contains the scoped screen template and its system defaults (PF keys, commands, exit states, templates, video attributes, prompts, and edit patterns).	Access
Short, Default	All bilingual text for system defaults.	Read

Screens, Procedure Steps, Action Blocks, Windows, and Procedures

This Template expansion	Includes the following additional objects for all subset types	with this protection
Default	All screens that use the scoped screen template.	Modify
Default	Procedure steps for the included screens.	Access
Default	The non-expanded action block for the procedure steps included (views only).	Access
Default	Any windows for the procedure steps.	Read
Default	Procedures for the procedure steps that include the screens.	Access

Referenced Entity Types and Parent Subject Areas

This Template expansion	Includes the following additional objects for all subset types	with this protection
Default	Entity types referenced in action block views.	Access
Default	Parent subject areas of referenced entity types.	Access

Transaction Operation

The transaction operation scoping object is a single-purpose commit unit of processing typically belonging to a Business System object type, but which can exist independently.

For related information about the owning entity and supporting procedure step, see the respective expansions for these objects.

Trans Operation

The following table shows trans operation design, unit test, and system test:

This Trans Operation expansion	Includes the following additional objects for all subset types	with this protection
Default	Constraints and external parameters	Delete
Default	Extended constraints for included constraints	Delete
Default	Delegated to transaction operation	Access
Default	Referenced commands	Access
Default	Owning entity or work attribute set	Access
Default	Supporting procedure step	Access

Typemap

The following table shows typemap design, unit test, and system test:

This Typemap expansion	Includes the following additional objects for all subset types	with this protection
Default	Owned correspondences	Delete
Default	Target attributes	Modify
Default	Forward mapped action blocks	Modify
Default	Backward mapped action blocks	Modify

User Defined Object

The following table shows user defined object design, unit test, and system test:

This User Defined Object expansion	Includes the following additional objects for all subset types	with this protection
Default	Parent user defined object class	Access

User Defined Object Class

The following table shows user defined object class design, unit test, and system test.

This User Defined Object Class expansion	Includes the following additional objects for all subset types	with this protection
Default	Aggregate objects	Access
Default	User defined objects	Modify

Web Service Definition

The following table shows web service definition design, unit test, and system test:

This Web Service Definition expansion	Include the following additional object types for all subset types	with this protection
Short Default Full	Owning Business System	Access
Short Default Full	Web Services included in the Web Service Definition	Modify
Short	Procedure Steps included in the Web Service	Modify
Default	Procedure Steps included in the Web Service	Modify

This Web Service Definition expansion	Include the following additional object types for all subset types	with this protection
Full	Procedure Steps included in the Web Service	Modify

Window Load Module

To check out all procedure steps that are contained in a load module for a windowed application, use the window load module scoping object.

Scope a windowed load module to perform

- Window packaging
- Window code generation tasks

A procedure step can have many windows. It can also have both windows and a single screen. However, the combination of windows and screen restricts the usage of windows.

The expansion level and protection option of the scoped window load module are applied to each procedure step within that program. All packaging is uploaded to the Host Encyclopedia.

Note: The window load module has the same five levels of expansion as procedure steps: Short, A, B, Default, and Full. The expansion level and protection option of the scoped window load module apply to each procedure step within that program, Items that are marked if appropriate (such as screens) are included only if included for a procedure step at the expansion you select.

Unit Test

The following tables show Window Load Module - Unit Test:

Program

This Window Load Module expansion	Includes the following additional objects for Unit Test	with this protection
All (Short, A, B, Default, Full)	The scoped window load module.	Modify

Packaged Procedure Steps

This Window Load Module expansion	Includes the following additional objects for Unit Test	with this protection
All	All procedure steps packaged in the scoped window load module.	Modify
All	The procedure step execution units for each step in the program.	Delete
All	Package list entry	Delete

Trancodes

This Window Load Module expansion	Includes the following additional objects for Unit Test	with this protection
All	The trancodes for the scoped window load module.	Modify

Procedures and Business Systems

This Window Load Module expansion	Includes the following additional objects for Unit Test	with this protection
All	The procedures for the procedure steps of the program.	Access
All	The business systems containing these procedures.	Access

System Defaults

This Window Load Module expansion	Includes the following additional objects for Unit Test	with this protection
All	System defaults (PF keys, commands, exit states, templates, video attributes, and edit patterns).	Access

Screen and Windows

This Window Load Module expansion	Includes the following additional objects for Unit Test	with this protection
All	All screen and screen objects (templates, prompts, variables) for the procedure steps of the scoped window load module, if appropriate.	Modify
All	All windows and window objects (prompts, variables, literals) for the procedure steps of the scoped window load module, if appropriate.	Delete

Action Blocks

This Window Load Module expansion	Includes the following additional objects for Unit Test	with this protection
All	The fully expanded action blocks and USEd action blocks of all procedure steps for the scoped window load module, if appropriate.	Modify

Referenced and Shared Companion Entity Types

This Window Load Module expansion	Includes the following additional objects for Unit Test	with this protection
All	All entity types referenced in any action block views.	Access

Database and Data Tables

This Window Load Module expansion	Includes the following additional objects for Unit Test	with this protection
All	Fully expanded data tables for any transformed entity types.	Read
Default	Extended tables	Delete

This Window Load Module expansion	Includes the following additional objects for Unit Test	with this protection
Default	Tablespace referenced by extended tables.	Delete
All	Database and technical design for the data tables.	Read

TD Action Blocks, DBRMs, and Implementation Logic

This Window Load Module expansion	Includes the following additional objects for Unit Test	with this protection
All	Unexpanded TD action blocks for every entity and table in the subset.	Read
All	Implementation logic units and DBRMs for every TD action block.	Read

System Test

The following tables show window load module - system test.

Program and Its Trancodes

This Window Load Module expansion	Includes the following additional objects for System Test	with this protection
All	The scoped window load module.	Modify
All	The trancodes for the scoped window load module.	Access

Packaged Procedure Steps

This Window Load Module expansion	Includes the following additional objects for System Test	with this protection
All	All procedure steps packaged in the scoped window load module.	Modify
All	The procedure step execution units for each step in the program.	Delete
All	Package list entry	Delete

Clear Screen and Dialog Flow Trancodes

This Window Load Module expansion	Includes the following additional objects for System Test	with this protection
All	Clear screen and dialog Flow trancodes for all the procedure steps.	Modify

Procedures and Business Systems

This Window Load Module expansion	Includes the following additional objects for System Test	with this protection
All	The procedures for the procedure steps of the program.	Access
All	The business systems containing these procedures.	Access

System Defaults

This Window Load Module expansion	Includes the following additional objects for System Test	with this protection
All	System default information (PF keys, commands, exit states, templates, video attributes, and edit patterns.)	Access

Screen and Windows

This Window Load Module expansion	Includes the following additional objects for System Test	with this protection
All	All screens and screen objects (templates, prompts, variables) for the procedure steps of the scoped window load module, if appropriate.	Modify
All	All windows and window objects (prompts, variables, literals) for the procedure steps of the scoped window load module, if appropriate.	Delete

Action Blocks, Implementation Logic, DBRMs

This Window Load Module expansion	Includes the following additional objects for System Test	with this protection
Short, A, B, Default	The stubs for all USEd action blocks of the procedure steps of the window load module.	Modify
Full	The fully expanded action blocks and USEd action blocks of the procedure steps of the window load module, if appropriate.	Modify
All	The implementation logic objects for these action blocks.	Delete

This Window Load Module expansion	Includes the following additional objects for System Test	with this protection
All	DBRMs for these implementation logic objects.	Delete

Referenced and Shared Companion Entity Types

This Window Load Module expansion	Includes the following additional objects for System Test	with this protection
All	Expanded treatment of entity shared companions for all entity types used in views of action blocks.	Read

Data Tables and Databases

This Window Load Module expansion	Includes the following additional objects for System Test	with this protection
All	Fully expanded data table for any transformed entity type used in the view of the procedure step or other USEd action blocks.	Read
Default	Extended tables	Delete
Default	Tablespace referenced by extended tables.	Delete
All	Database and technical design for the data tables.	Read

TD Action Blocks, DBRMs, and Implementation Logic

This Window Load Module expansion	Includes the following additional objects for System Test	with this protection
All	Fully expanded TD action blocks for each entity type.	Read

This Window Load Module expansion	Includes the following additional objects for System Test	with this protection
All	Implementation logic units and DBRMs for every TD action block.	Read

Dialog Manager

This Window Load Module expansion	Includes the following additional objects for System Test	with this protection
All	A Dialog Manager between load modules that includes the following: ■ Flows into or out of each expanded load module in the subset ■ Procedure steps for these flows with short expansion ■ Entities and attributes used in the views of these steps ■ Data tables for the entities, fully expanded ■ Extended tables ■ Tablespace referenced by extended tables ■ Databases for the tables, fully expanded ■ Unexpanded TD action block for the entities and tables ■ Implementation logic unit and DBRM for the TD action blocks ■ Trancodes for the flowed to load modules ■ Unexpanded load modules being flowed to ■ Implementation units for the load module only	Read

Work/System Attribute Set

To maintain any of the three types of work/system attribute sets, use the work/system attribute set scoping object:

- The system-defined attribute set, which contains all CA Gen system attributes
- IEF_SUPPLIED work attribute set
- User-defined work attribute sets

CA Gen attribute sets are included as part of the default set of objects that are supplied with every subset if they are not explicitly scoped.

Include a work attribute set in a subset to:

- Change or delete a user-defined entity type or its attributes in the work view of an information view
- Change a system-supplied work attribute
- Change or delete the prompts (and any associated bilingual text) associated with a system attribute

Scoping the $IEF system attribute set lets you change or delete the prompts (and any associated bilingual text) associated with a system attribute.

To access dialect-related functions, also scope a business system or any object whose expansion includes a business system.

Work/System Attribute Set

The following tables show work/system attr set - design, unit test, and system test:

Entity Types and Attributes

This Work/ System Attr Set expansion	Includes the following additional objects for all subset types	with this protection
Default	Entity types used in the work views.	Modify
Default	Properties of the work attribute set.	Modify
Short, Default	Attributes.	Modify
Short, Default	Permitted values.	Modify

Prompts and Bilingual Test

This Work/ System Attr Set expansion	Includes the following additional objects for all subset types	with this protection
Default	Prompts for the attributes.	Modify
Short, Default	Prompt for values.	Modify
Short, Default	Prompts.	Modify
Default	All bilingual text for the prompts.	Delete
Short, Default	Dialect text for prompt for values and prompts.	Delete

Transactions

This Work/ System Attr Set expansion	Includes the following additional objects for all subset types	with this protection
Default	Owned transaction operations.	Access

z/OS Library

The following table shows z/OS Library design, unit test, and system test:

This z/OS Library expansion	Include the following additional object types for all subset types	with this protection
Short Default Full	Owning Business System	Access
Short Default Full	Action Blocks included in the z/OS Library	Modify

Index