CA Gen

Analysis Tools Reference Guide

Release 8.5

CA Technologies Product References

This document references the following CA Technologies products:

- CA Gen
- AllFusion® Gen
- COOL: 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: Data and Activity Analysis

This chapter provides information about data and activity analysis tasks and tools.

Definition

Analysis is the definition and refinement of:

- Data aspect of the model, which describes things of interest to the business and the relationships between them.
- Activity aspect of the model, which records things the business does (or should do.)
- Interaction aspect of the model, which details how the activities affect the data.

Purpose

You create and refine data, activity, and interaction aspects (or components) of your business model so that you can understand them separate from your business'

- Current organization
- Application systems
- Information technology

This independence lets you focus on completeness and consistency, critical characteristics later when CA Gen (formerly known as COOL:Gen, AllFusion Gen) helps you transform your conceptual models into physical ones. Your physical models will only be as complete and accurate as your conceptual ones.

Deliverables

After completing Analysis, you get:

- An Entity Relationship Diagram (ERD) that describes the data requirements of the entire business
- Activity Hierarchy and Activity Dependency Diagrams that describe the activity requirements of the entire business
- Supporting documentation for the data and activity diagrams

Where To Go Next

After completing Data and Activity Analysis, you will need to understand how to create process action diagrams before the next phase of development, Design. For complete information about what is involved in these tasks, see the *Action Diagram User Guide* and the *Design Guide*.

Analysis Tasks

You will perform numerous tasks during the Analysis phase of your development effort. You will perform some of these tasks using the CA Gen Analysis toolset. You will perform other tasks during information gathering sessions and other research activities. This guide presents instruction and techniques for those tasks that you will complete through the Analysis toolset. It is assumed that you already know why you must perform a particular activity and just require help on using the tools CA Gen provides to achieve your goal.

Common Tasks

The common tasks are as follows:

- Find the data and activities.
- Define the data and activities.
- Detail the data and its relationships. See the chapter Data Model Objects (see page 17) in this guide.
- Detail the activities and their hierarchies and dependencies.
- Refine the data and activities to remove redundancies.
- Verify that the data and activities are consistent and complete using Consistency Check.

Analysis Tools

CA Gen provides a rich set of tools to help you analyze data and activities as you build your application. These tools help you to perform the following tasks:

- Organize and document all relevant data and activities
- Identify and eliminate data and activity ambiguity, inconsistency, and duplication
- Ensure consistent use of data and activity terminology

For more information about these tasks, see the context-sensitive and overview level help of all of the tools.

Data Analysis Tools

The following tools are used primarily for analyzing data, but they can also help in activity analysis:

- Data Model Tool
- Data Model Browser Tool
- Data Model List Tool
- Entity Lifecycle Diagramming Tool

For more information about Data Modeling tools, see the chapter Introducing Data Analysis (see page 13).

Activity Analysis Tools

The following tools are used primarily for analyzing activities, but they can also help in data analysis.

- Activity Hierarchy Diagramming Tool
- Activity Dependency Diagramming Tool

Interaction Analysis Tool

Use the Matrices tool for analyzing data and activities and for analyzing interaction between data and activities.

Chapter 2: Introducing Data Analysis

This chapter discusses about the various data analysis tools and the process modeling considerations.

Data Analysis Tools

CA Gen provides a rich set of tools to help you analyze data (and activities) as you build your application. These tools help you to perform these tasks:

- Organize and document all relevant data
- Identify and eliminate data ambiguity, inconsistency, and duplication
- Ensure consistent use of data terminology

For more information about these tasks, see the context-sensitive and overview level help of the tools.

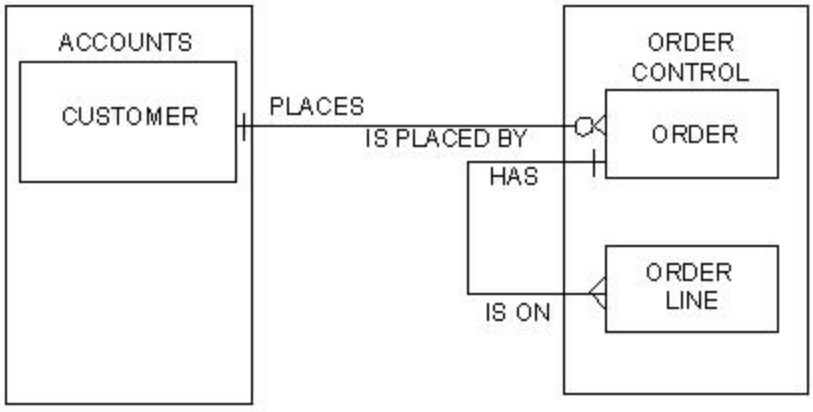
Data Model Tool

The Data Model Tool lets you add, change, and delete data in a graphic style. Use this tool to produce the central diagram in data analysis. The ERD lets you perform the following tasks:

- Graphically show the types of data (entity types)
- Show connections (relationships) between entity types
- Record data characteristics (attributes)
- Record data uniqueness (identifiers)
- Logically group entity types into subject areas

This graphic tool is useful in initial data analysis, and when you need to review the data with users who may not be familiar with it. For additional information about the graphic tool, see the chapter Data Model Objects (see page 17).

A Sample Entity Relationship Diagram is shown next.



Data Model Browser Tool

The Data Model Browser Tool lets you add, change, and delete data in a text style. This tool displays entity types, attributes, relationships, descriptions, and associated action blocks in an object browser style. When you select an entity type, the tool gives you a *wide band* look at the type, its characteristics, and the action blocks that use it. For more information about entity types, see the chapter titled Data Model Objects (see page 17). This tool is useful to analysts and designers as they build applications.

Data Model List Tool

The Data Model List Tool lets you add, change, and delete data in a list style. This tool shows you Subject Areas and Entity Types and Attributes. When you select one, the tool gives you a focused, *narrow band* view of the selected object at a detailed level. For more information about Subject Areas and Entity Types and Attributes , see the chapter titled Data Model Objects (see page 17). This tool is useful to data and database administrators who need to review data content in detail.

You can also use this tool concurrently with the Activity Hierarchy Diagramming (AHD) Tool which helps you analyze the activity aspect of your business model. Using the two tools together lets you discover and classify data and activities in parallel.

Component Model

The Component Model is a flexible tool for constructing and viewing the data model. It offers an alternate view of the Data Model Diagram, Data Model Browser and Data Model List tools. Changes made to the data model in one of these tools are visible in all other tools.

The Component Model allows you to:

- Create any number of diagrams, including overlapping diagrams, to show pieces of the data model based on your needs
- Add, modify and delete entity types, subtypes, operations, relationships, and attributes

Using the Component Model, you select entity types to place on the diagram. It is not necessary to place every entity type. Also, the same entity type can be placed on more than one diagram.

A Component Model diagram is restricted to a particular subject area. When the root subject area, representing the entire model, is selected, any entity type in the model can be placed in the Component Model. When a lower level subject area is selected, only the entity types in the selected subject area can be placed on the diagram.

There is no limit to the number of Component Model Diagrams that can be created for any given subject area.

Usage

The Component Model is useful for both component-based development (CBD) and non-CBD projects.

For CBD projects, the diagram enables you to classify subject areas as specifications or implementations, and to draw separate diagrams of component specifications and implementations.

For non-CBD projects, the diagram offers multiple perspectives of the data model by enabling you to create diagrams and select the entity types to be placed in each diagram according to your needs.

Entity Life Cycle Diagramming Tool

The Entity Life Cycle Diagramming Tool lets you see how and when data changes in a dialog flow style. This tool lets you analyze the life cycle through which an Entity Type passes. You can see the Life Cycle Partitioning of an Entity Type and detail the Elementary Processes which cause a life cycle change.

You can use this tool to perform the following tasks:

- Identify Entity States and their associated Transition Processes
- Determine if states and processes support each other
- Add the discovered and identified states and processes to their diagrams

Process Modeling Considerations

This section discusses the process modeling considerations.

Process Synthesis

Process synthesis is the automatic generation of processes and detailed process logic based on a consistent part of the data model. Process Synthesis tracks entity types related to the subject entity type, creating the required views and action statements from each. The data objects you identify during analysis will be used to produce the views and action statements. For more information about the views in actions statements, see the *Action Diagram User Guide.*

Construction

Data objects you create are also used as input to define the executable components of a system. Action statements you create become SQL statements that refer to the Data Definition Language objects.

Note: For more information about DDL, see the *Workstation Construction User Guide*.

Chapter 3: Data Model Objects

This chapter discusses about the subject area and gives you an overview of entity type, relationship, attribute, partitioning, and entity subtype object.

Subject Area

A subject area is an object that aggregates or collects Entity Types of interest by major resource, product, or activity. Sales, personnel, accounts, and products are typical subject areas in a business.

Subject Area Name

Name each subject area with a plural noun.

Entity Type Overview

An Entity Type is an object that groups entities (which are instances of entity types), together by definitions and characteristics. You group Entity Types within appropriate Subject Areas. Together these two objects should capture and group any data of interest to a business. For example, Customer could be the Entity Type of a group of entities (such as people or companies) who buy products.

Each CA Gen stage references Entity Types. In Analysis and Design, Entity Types are referenced in entity views. (View definitions in CA Gen are like parameters or variable declarations in programming languages.) In the activity analysis and interaction analysis of phases of Analysis, entity views let you specify the data used by the business area's functions and processes. In Design, procedures use entity views, and each Entity Type is implemented as a record type in the Data Structure List, and subsequently as a table in a relational DBMS.

Entity Type Properties Dialog

Describes the estimated number of entities and the anticipated increase (or decrease) in that number over time.

Expected Number of Occurrences

Specifies the number of entities the business expects to store. Record the number in terms of the minimum, maximum, and average. Derive these numbers from past records or from business forecasts. You cannot enter a maximum less than the minimum, or an average greater than the maximum.

Expected Growth Rate

Specifies the expected increase (or decrease) in the number of entities (entity type occurrence) over time. Express the value as a percentage per year, month, week, or day.

Note: During Design, the specified numbers for number of occurrences and growth rate provide a basis for physical design. They are used to calculate database data set sizes, and to help the database designer determine automation cost benefits.

Data Structure

Specifies the suggested name for the record that will implement the Entity Type. The Data Structure name can be different from the Entity Type name. If a different name is not specified, the Data Structure name defaults to the Entity Type name after entering the Entity Type name.

Relationship Overview

A Relationship is an object that shows a relevant association among Entity Types. Each Relationship always has two memberships, one for each participating Entity Type. An Entity Type can also have a Relationship with itself. Give every Relationship Membership a name and properties.

Relationship Description

Create a block of text description of the source and destination properties.

CA Gen distinguishes the two memberships as the source membership and the destination membership. The arbitrary designation depends solely which Entity Type you select first (source) and second (destination) when joining them.

For example if you select EMPLOYEE first and DEPARTMENT second when joining Entity Types, the source membership (EMPLOYEE) is linked to the destination membership (DEPARTMENT).

Note: In Design, a Relationship is implemented as a linkage in the Data Structure List. For more information, see the *Toolset Help*.

Membership Quantity of Occurrences

The quantity of occurrences of pairings are as follows:

- Frequency of pairings

  Specifies the estimated percentage of times entities of the destination Entity Type are paired with entities of the source Entity Type.

- Number of pairings

  Specifies the minimum/maximum or average number (estimated or actual) of entities of the destination Entity Type that can participate in the pairing.

Note: Frequency of pairing and number of pairings become important during database design because they help determine allotted space amounts.

Membership Deletion Rule

The Deletion Rule specifies source and destination properties and helps you prevent unintended cascade deletions. This rule determines how deletion of one entity in a pairing affects the other entity - whether the deletion of the source entity will:

- Delete each paired entity of the destination Entity Type
- Disassociate each paired entity of the destination Entity Type
- Disallow deletion if there are one or more paired entities of the destination and Entity Type

Source and Destination Memberships

Source and destination are distinctive memberships of a Relationship. You enter the source membership at the top of the Relationship Properties Panel, the destination membership at the bottom.

Attribute Overview

An Attribute is a fact that describes an Entity Type. Its values are associated with individual entities of a specific Entity Type. For example, Name and Date of Birth are Attributes of EMPLOYEE. You may create a set of Attributes for each Entity Type.

Attributes do not appear graphically on the ERD, but are available behind the scenes. You can create, edit, and manipulate them while building the diagram.

Throughout Design, Attributes are used in data views, and they are implemented as fields when you build screens, windows, or web pages. They are also implemented as fields in the Data Structure List. For more information, see the online help for the Data Structure List and the Data Store List.

Attribute Domain

The Domain specifies a collection of values from which each value of an Attribute must be taken. Each Attribute falls into one of the following domains:

- Text attribute value is a character string
- Date attribute value represents a date (including century)
- Time attribute value represents an hour, minute, and second
- Number attribute value is numeric. If you use an arithmetic calculation to specify the value of an attribute, the attribute's domain must be a number.
- Timestamp attribute value represents a combination of the date and the time of day to the millisecond.
- DBCS text type indicates a double-byte character set.
- Mixed text type indicates a combination of a double-byte character set and a single-byte character set.
- GUI Object represents a pointer to an OLE object (this option is available only with the workset attribute).
- BLOB attribute value represents a variable sized Binary Large Object used to store large data objects such as digital images, video, or sound files.

Note: For numeric attributes, the actual precision of arithmetic operations and number of significant digits depends on the version of the platform, compiler, and DBMS. In general, you should be able to get up to at least 15 significant digits of accuracy.

Attribute Length

- The Length specifies the maximum number of characters or digits allowed for each value of an Attribute, according to the domain.
- Text Attributes have a limit of 4094 for varying-length attribute, 255 for fixed-length attributes.

- Date Attributes have eight characters, including four digits for year (YYYYMMDD). For example, January 1, 1992 becomes 19920101.
- Time Attributes have six characters (HHMMSS). You cannot change the number of characters for date or time.
- Numeric Attributes (actual maximum and minimum precision of arithmetic operations, number of significant digits) depend on the version of the platform, compiler, and DBMS.
- Timestamp Attributes have 26 characters, including four digits for year (YYYY-MM-DD-HH.MM.SS.NNNNNN, where N is a microsecond.)
- BLOB Attributes have a limit of 2 gigabytes.

The Attribute length specified in the ERD is the amount of space reserved for the field in the database. This may be different from the field as it is displayed on a screen. For more information about the Attribute Length and dialect definition, see the online help.

Note: Different DBMS products provide different levels of precision for fractions of a second. Refer to your specific DBMS documentation for more information.

Attribute Varying Length

Lets you specify that the length of the text Attribute can vary when stored in the database. The database only stores the actual length of the text string.

Note: Different DBMS products provide different levels of precision for fractions of a second. Refer to your specific DBMS documentation for more information.

Attribute Case Sensitivity

Case Sensitivity indicates if the case of alphabetic characters in the Attribute value is significant. A check in this field means case does matter on all screen and window uses of the Attribute. Also note that this means the case will be maintained in the database. When this field is not checked, all characters that have upper and lower case equivalents will be converted to upper-case.

Attribute TD Name

The TD Name identifies the record in the database associated with the Attribute. The TD Name can be different from the Attribute name. If a different name is not specified, the TD Name defaults to the Attribute name.

Attribute Derivation Algorithm

A Derived Attribute does not have permitted values or defaults. Its value is calculated whenever the attribute is read from the database.

Specify the name of the derivation algorithm that calculates the Derived Attribute's values. When you name it, CA Gen establishes an action block and names it the same as your specified algorithm. Specify the formula for a derivation algorithm in its action block. Each time a READ or READ EACH action statement uses a view that contains a Derived Attribute, the Attribute's derivation algorithm is invoked automatically.

Note: Derivation algorithm names must be unique. If you try to use a name already used, an error message appears, and you should change the name. Each derivation algorithm derives only one Attribute's value.

Attribute Aliases

Attribute aliases are names that the attribute is also known as. This is documentation that is used to help analysts know that this attribute may also be called by other names. The aliases are not used at design time: only the attribute name is used.

Attribute Optionality

Optionality is a characteristic of an attribute indicating whether the attribute is mandatory or optional. A mandatory attribute must have a value for every entity of the type or subtype it describes whenever an entity of the particular type or subtype is referred to or used.

Partitioning Overview

A Partitioning is an Entity Type subdivision or decomposition that lets you create Entity Subtypes within that Entity Type. You must define a classifying Attribute, which is the attribute whose value determines which Entity Subtype an entity occurrence belongs to. For example, the Entity Type EMPLOYEE might have MANAGER and STAFF Entity Subtypes, based on the classifying Attribute Job Type. Employees could also be RETIRED and ACTIVE Entity Subtypes based on Job Status.

Partitioning Discovery

You discover Partitioning by discovering Entity Subtypes.

Partitioning Name

Use the classifying Attribute name.

Partitioning Description

Create a block of text description that specifies the Entity Type and its division into Entity Subtypes.

Partitioning Properties

You can associate a Partitioning with a classifying Attribute. A classifying attribute is one whose value determines which Entity Subtype an entity belongs to.

If a partitioning is fully enumerated, every entity belonging to the entity type it divides must belong to one of the subtypes in the partitioning. Every value of the classifying Attribute identifies an Entity Subtype, and every parent Entity Type occurrence is an occurrence of an Entity Subtype.

For example, Job Type could be fully enumerated because company employees are either managers or staff. Job Status could be not fully enumerated because Job Status could also be LEAVE OF ABSENCE.

Life Cycle Partitioning specifies the states an entity can pass through from creation to archiving or deletion. For example, in time NEW EMPLOYEE becomes RETIRED EMPLOYEE or FORMER EMPLOYEE.

An Entity Type can have only one life cycle Partitioning, which must be fully enumerated. Life Cycle Partitioning is used in Life Cycle Analysis.

Partitioning Classifying Values

Create an Entity Subtype first, then specify its properties.

An Entity Type or Entity Subtype can have multiple Partitionings if it has more than one classifying Attribute. For example, EMPLOYEE can be subdivided into EXEMPT EMPLOYEE and NONEXEMPT EMPLOYEE in one Partitioning, and into PERMANENT EMPLOYEE and TEMPORARY EMPLOYEE in another.

After specifying its properties, define an Entity Subtype's classifying value or the state of a Partitioning.

For example, if the Entity Type is EMPLOYEE and the classifying Attribute is Nationality, Foreign and Domestic can be classifying values of the Partitioning.

An Entity Subtype subdivides an Entity Type based on a narrower definition and additional Attributes and Relationships. It must have at least one Attribute and/or one Relationship not shared by others. Entity Type CUSTOMER may fall into two mutually exclusive groups: FOREIGN and DOMESTIC. All customers share a set of characteristics, but foreign and domestic customers have Attributes and/or Relationships beyond those commonly shared.

Life Cycle States

In life cycle partitionings, there is a series of Entity States, which may have been defined as subtypes. An entity can exist in four different types of states:

- Creation state

  Specifies the initial state of an entity after the business becomes interested in it

- Termination state

  Specifies the final state in the life of an entity

- Null state

  Specifies the state of an entity when no information is available, as in an entity before the creation state

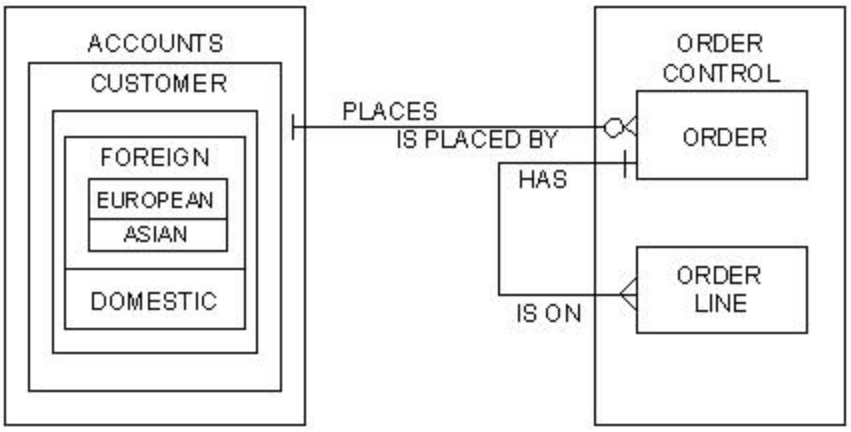
- Intermediate state

  Specifies any state that is not a creation, termination, or null state

Entity Subtype Overview

You can also subdivide an Entity Subtype. For instance, CUSTOMER can be DOMESTIC or FOREIGN. FOREIGN CUSTOMER can be EUROPEAN or ASIAN.

The following diagram displays Entity Subtypes in an ERD:



Entity Subtype Requirements

Add a Partitioning to an Entity Type before creating Entity Subtypes within it.

Entity Subtype Inherited Characteristics

An Entity Subtype's characteristics and all Entity Type's inherited characteristics apply to the subtype. For example, all customers have a value for Attribute Name, but foreign customers must also have a value for the Attributes Country Code and Import License Number. The subtype also inherits Identifiers and Mutually Exclusive constraints; you can add others if they apply only to the Entity Subtype, not to its parent.

Entity Subtype Classifying Attribute

The Classifying Attribute is the mandatory basis for subdividing the Entity Type into Entity Subtypes. Classifying Attribute value determines the Entity Subtype placement. For Customer, Foreign and Domestic are classifying values of the classifying Attribute Nationality.

BLOB attributes may not be selected as a classifying attribute.

Chapter 4: Business System Definition

This chapter discusses the various business system definition concepts.

Business System Definition Concepts

In Analysis you work with elementary processes by:

- Discovering and defining them in your Process Hierarchy Diagram
- Sequencing them in your Process Dependency Diagram
- Detailing them in your Process Action Diagram

You define a business system by grouping these elementary processes into potential systems. You use either the Business System Definition (BSD) Tool or the Standard Matrices to name and define the system.

Grouping Techniques

The various grouping techniques are as follows:

- **Affinity Clustering**

  Specifies grouping by how often processes and data are used together.

- **Event Response**

  Specifies grouping by the executions of one or more elementary processes responding to specific events.

- **Logical Constraint Analysis**

  Specifies grouping by the prerequisites for implementing each process. Ordinarily, you will implement processes in the order they occur in the business life cycle and business systems in the order data is created by the processes.

- **Primitive Function Groupings**

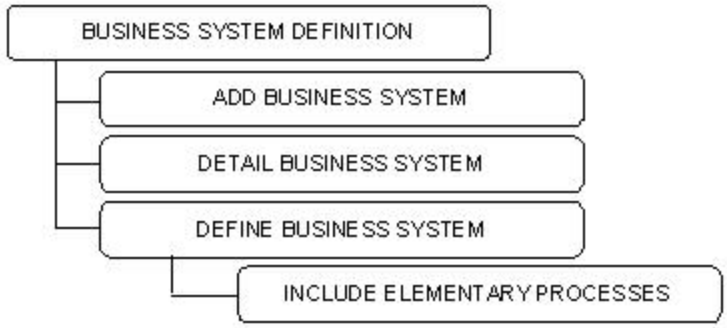
  Specifies grouping by categories and content of functions.

Definition Tool

When you access the BSD Tool to define a new business system or view or change an existing business system, CA Gen displays the unexpanded activity hierarchy associated with the current model. You must expand the hierarchy to work in the hierarchy, as shown in the following figure.

On the BSD screen, processes that are already included in the selected business system are displayed in yellow with a black background. Elementary processes that are already included in a different business system are displayed in white with a black background. Elementary processes not included in a business system are displayed in white on a blue background.

Business System Definition Tasks



Chapter 5: Creating a Business Plan

This chapter discusses how to create a business plan using a matrices analysis. It also discusses how various organizational units are affected by the need of certain information.

Using Matrices Analysis

Matrices Analysis helps you create a business plan by analysis of objectives, strategies, critical success factors, information needs, performance measures, and so forth.

Business Objectives and Strategies

Add specific intentions and methods to the Objective/Strategy matrix.

Add cell values where a strategy supports an objective. Indicate a numeric degree of support between 1 (minimum) and 9 (maximum).

Critical Success Factors

Open the Objective/Critical Success Factor matrix.

Add the critical success factors. Indicate a numeric degree of impact a factor has on an objective between 1 (minimum) and 9 (maximum).

Intermediate Strategy Goals

Create a user-defined matrix with Objective and Goal.

Add the goals to the matrix. Indicate a numeric degree of impact a goal has on an objective between 1 (minimum) and 9 (maximum).

Key Performance Measures

Create a user-defined matrix with Objective and Performance Measure.

Consider each objective, and mark any cell where the Performance Measure is associated with the objective. Indicate a numeric degree of impact a performance member has on an objective between 1 (minimum) and 9 (maximum).

Information Needs Impact

Evaluate how objectives, strategies, goals, performance measures, critical success factors, and organizational units are affected by the need for certain information. This also helps establish priorities when considering information systems development.

Information Need/Strategy

Open the Information Need/Strategy matrix. Indicate a numeric degree of impact an information need has on a strategy between 1 (minimum) and 9 (maximum).

Information Need/Performance Measure

Open the Information Need/Performance Measure matrix. Indicate a numeric degree of impact an information need has on a performance measure between 1 (minimum) and 9 (maximum).

Information Need/Critical Success Factor

Open the Information Need/Critical Success Factor matrix. Indicate a numeric degree of impact an information need has on a critical success factor between 1 (minimum) and 9 (maximum).

Information Need/Organizational Unit

Open the Information Need/Organizational Unit matrix. Indicate a numeric degree of impact an information need has on an organization between 1 (minimum) and 9 (maximum).

Information Need/Information Need Category

Open the Information Need/Information Need Category matrix. Ideally, there is at least one information need for the Strategic, Planning, Control, and Operational categories. Indicate a numeric degree of importance for an information to a category between 1 (minimum) and 9 (maximum).

Organizational Units and Strategies

Open the Strategy/Organizational Unit matrix. All organizational units in the Organizational Unit Hierarchy Diagram appear in the matrix.

Indicate a numeric degree of involvement for a unit to a strategy between 1 (minimum) and 9 (maximum).

In later stages, use this matrix to determine the principal users of key systems and to quantify the users' stake in the systems.

Success Factors and Organizational Units

Open the Critical Success Factor/Organizational Unit matrix.

Cell values indicate if the organizational unit is active or passive. Active organizational units can positively manage these critical success factors. Passive organizational units can influence, or indirectly impact a critical success factor. Recommended cell values are 1 (active) and 2 (passive).

Performance Measures and Organizational Units

Open the Performance Measure/Organizational Unit matrix. Each organizational unit can be associated with one or more Performance Measures. This confirms that the planned systems will satisfy the needs of the responsible organizational units.

One or both of the matrix axes are populated if the Organizational Hierarchy Diagram exists, or the performance measures are entered in a different matrix.

Business Functions and Information Need Categories

Create a user-defined matrix with Information Need Category (Planning, Control, Operational) and Function.

Indicate a numeric degree of importance for an information category to a function between 1 (minimum) and 9 (maximum).

Chapter 6: Clustering Matrices

This chapter discusses the various clustering matrices used to define a business system.

Clustering Methods

Clustering Methods help you scope a project by identifying the Entity types and functions that make up a project or a business system.

Matrices Used

Two matrices can be used with Interaction Clustering: the Business Function/Entity Type matrix and the Entity Type/Elementary Process matrix. In both matrices, the cell values indicate if the function or elementary process Creates, Reads, Updates or Deletes the Entity Type.

Results

Interaction Clustering automatically groups together the activities that act upon the same data objects. Thus, a cluster contains one or more business functions and the Entity Types that the functions create and update.

The boundaries of a cluster define the smallest group of Entity Types and activities that form the scope of a system.

Entity Type/Business Function Matrix

The Entity Type/Business Function Matrix identifies the business functions involved with Entity Types, and records the functions' expected effects: CRUD (Create, Read, Update, Delete) values.

Entity Type/Business Function Matrix

Cell Values: = Not referenced C = Create D = Delete U = Updates R = Read Entity Types	Assess Credit	Track Design	Gain Commitment	Identify Needs	Develop Tech Sale	Conceive Product	Establish Sale Target	Manage Public Relations	Plan Research	Develop Specifications	Analyze Research	Manage Product Group	Negotiate Sale	Communicate	Develop Proposal	Set Prices	Manage vertical Market	Amend Contract	Implement Product	Purchasing
Commitment			C							C			C		C			C		C
Purchase Order			U							C			U		U			C		C
Vendor																				C
Vendor Employee																				C
Vendor Product																				
Product						C										R			U	
Product Application						C													U	
Product Group						C						U							C	
Customer Survey			U														C		C	
Market																	C			
Portfolio Strategy			U									C				C				
Product Price																C				
Advertising Medium								C						U						
Competitor				C																
Competitor Product				C																
Market Need				C							U									
Market Research									C		U									
Sales Target							C													
Design Project		U			C															
Geographic Zone																				

Clustered Entity Type/Business Function Matrix

The Clustered Entity Type/Business Function Matrix rearranges the rows and columns of the Business Function/Entity Type matrix or the Elementary Process/Entity Type matrix to show a staircase of clusters from top left to bottom right. See the Business Function/Clustered Entity Type Matrix.

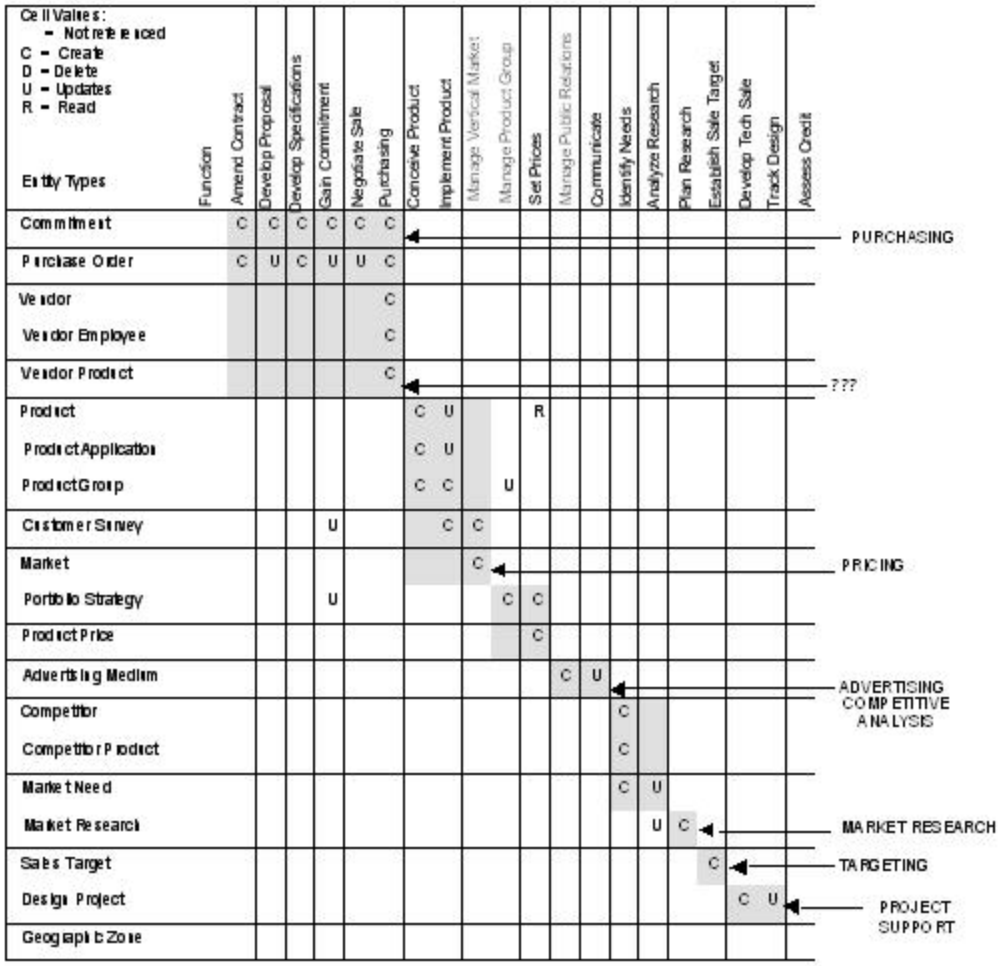
Clusters are based on Create and Update cell values. Functions and Entity Types are grouped together when the function creates or updates the Entity Type.

Clustering a matrix is an iterative process. Most often, a matrix is clustered, reviewed with the users, modified, and clustered again.

Sample Clustered Entity Type/Business Function Matrix

Cell Values: = Not referenced C = Create D = Delete U = Updates R = Read Entity Types / Function	Amend Contract	Develop Proposal	Develop Specifications	Gain Commitment	Negotiate Sale	Purchasing	Conceive Product	Implement Product	Manage Vertical Market	Manage Product Group	Set Prices	Manage Public Relations	Communicate	Identify Needs	Analyze Research	Plan Research	Establish Sale Target	Develop Tech Sale	Track Design	Assess Credit
Commitment	C	C	C	C	C	C														
Purchase Order	C	U	C	U	U	C														
Vendor						C														
Vendor Employee						C														
Vendor Product						C														
Product							C	U			R									
Product Application							C	U												
Product Group							C	C		U										
Customer Survey				U				C	C											
Market									C											
Portfolio Strategy				U						C	C									
Product Price											C									
Advertising Medium												C	U							
Competitor														C						
Competitor Product														C						
Market Need														C	U					
Market Research															U	C				
Sales Target																	C			
Design Project																		C	U	
Geographic Zone																				

Named Clusters Matrix



Matrix Without a Create Cell

There is no create cell value for the Entity Type Geographic Zone. Also, the function Assess Credit does not create or update an Entity Type. See the Matrix Without a Create Cell.

This raises questions that the users can answer about the expected effect of a function on an Entity Type.

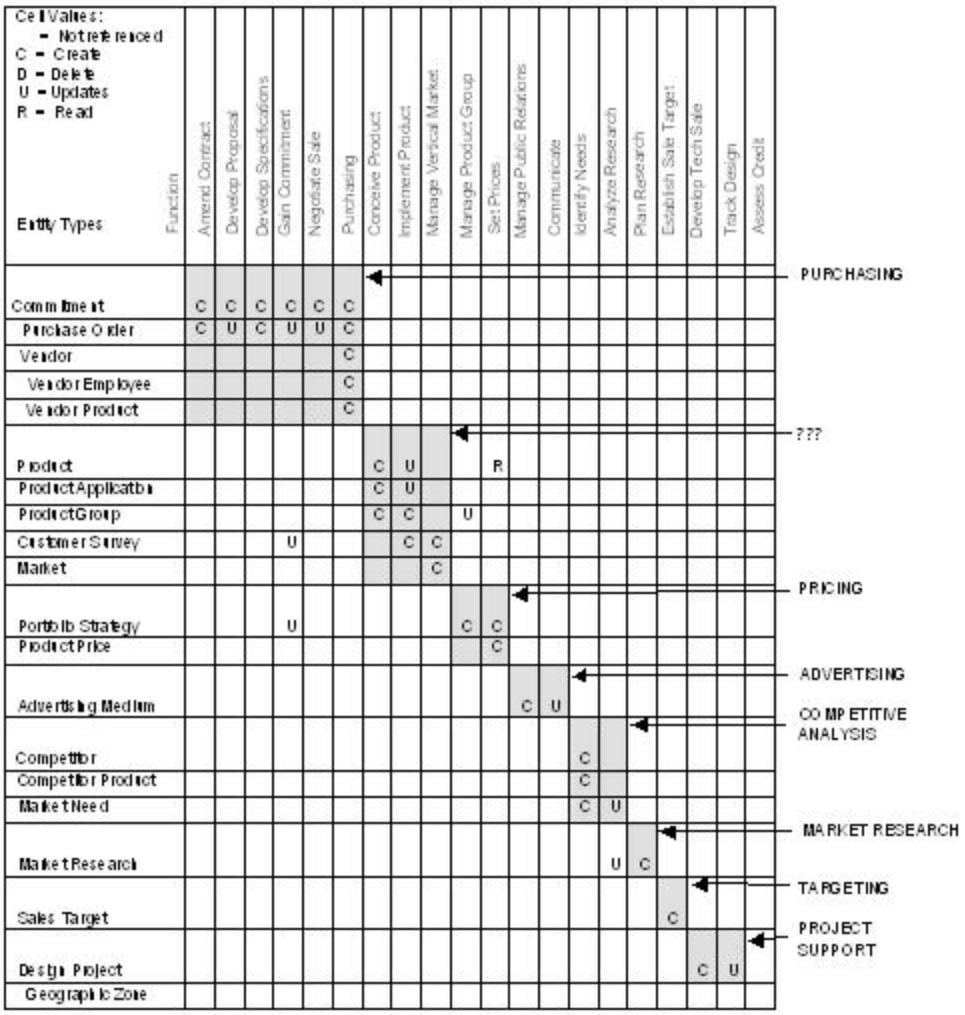
Matrix Without a Create Cell

Cell Values: = Not referenced C = Create D = Delete U = Updates R = Read Entity Types / Function	Amend Contract	Develop Proposal	Develop Specifications	Gain Commitment	Negotiate Sale	Purchasing	Conceive Product	Implement Product	Manage Vertical Market	Manage Product Group	Set Prices	Manage Public Relations	Communicate	Identify Needs	Analyze Research	Plan Research	Establish Sale Target	Develop Tech Sale	Track Design	Assess Credit
Commitment	C	C	C	C	C	C														
Purchase Order	C	U	C	U	U	C														
Vendor						C														
Vendor Employee						C														
Vendor Product						C														
Product							C	U			R									
Product Application							C	U												
Product Group							C	C												
Customer Survey				U				C	C											
Market									C											
Portfolio Strategy				U						C	C									
Product Price											C									
Advertising Medium												C	U							
Competitor														C						
Competitor Product														C						
Market Need														C	U					
Market Research															U	C				
Sales Target																	C			
Design Project																		C	U	
Geographic Zone																				

Read Interaction Dependencies

The Pricing cluster reads the Entity Type Product that is created and updated by the second cluster which is unnamed. See the Matrix below Showing Read Interaction Dependencies. It is simpler to implement the unnamed cluster first, but you may have to implement the Pricing cluster first. If so, you must determine how to access an existing products file until the second cluster is resolved.

Matrix Showing Read Interaction Dependencies



Using Affinity Clustering To Define a Business Area

You can define a business area by analyzing the results of affinity matrices and creating several other matrices.

Define Function/Entity Types Interaction

1. Identify Entity Types and add them to the data model.
2. Identify functions and add them to the Activity Hierarchy Diagram. Define the expected effects of activities on Entity Types.
3. Open the Entity Type/Business Function Matrix and select Affinity Clustering.
4. The CA Gen software creates two result matrices: the Business Function/Business Function matrix and the Entity Type/Entity Type matrix.

Identify Activity Clusters

1. Open the Business Function/Business Function matrix. Hide all cell values less than seven.

   The remaining cell values identify the functions with the highest degree of affinity. Each cluster of cells identifies an activity cluster. Note the functions in each activity cluster.
2. With the Business Function/Business Function matrix displayed, you can select Matrices from Analysis or Design in the CA Gen Main Window. Open the Activity Cluster/Business Function matrix.

   This allows you to review the activity clusters displayed in the Business Function/Business Function matrix while you add the cluster names to the Activity Cluster/Business Function Matrix.

   Assign each function to one activity cluster by marking the cell with an X.

Identify Data Clusters

1. Open the Entity Type/Entity Type matrix. Hide all cell values less than seven.

   The remaining cell values identify the Entity Types with the highest degree of affinity. Each cluster of cells identifies a data cluster. Note the Entity Types in each data cluster and write down a name for each data cluster.
2. Open the Data Cluster/Entity Type matrix.

   Add the data clusters to the matrix.

   Assign each Entity Type to one data cluster by marking the cell with an X.

Assign Clusters to Business Areas

1. Open the Activity Cluster/Business Area matrix.

   Add business areas to the matrix.

   Assign each activity cluster to a business area by marking the cell with an X.

2. Open the Data Cluster/Business Area matrix.

   Assign each data cluster to a business area by marking the cell with an X.

Display Clustered Function Groups

1. Open te Activity Cluster/Business Function matrix.

   Note the order in which functions appear.

2. Open he Entity Type/Business Function matrix.

   Move the functions so that they are in the same order as they appear in the Activity Cluster/Business Function matrix. This produces a matrix with groups of clustered functions.

Include Functions in Business Areas

1. Open the Activity Cluster/Business Area matrix.

   Note the activity clusters in each business area.

2. Open the Activity Cluster/Business Function matrix.

   Note the functions in each activity cluster.

3. Open the Entity Type/Business Function matrix.

   Use the Define BA action to include relevant functions in each business area

Display Clustered Entity Type Groups

1. Open the Data Cluster/Entity Type matrix.

   Note the order in which Entity Types appear.

2. Open the Entity Type/Business Function matrix.

   Move the Entity Types so they are in the same order as they appear in the Data Cluster/Entity Type matrix. This produces a matrix with groups of clustered Entity Types.

Include Entity Types in a Business Area

1. Open the Data Cluster/Business Area matrix.

   Note the data clusters in each business area.

2. Open the Data Cluster/Entity Type matrix.

   Note the Entity Types in each data cluster.

3. Open the Entity Type/Business Function matrix.

   Use the Define BA action to include relevant Entity Types in each business area

Verify Business Areas

1. Open the Business Area/Business Function matrix.

   Verify that all functions are included in one and only one business area.

2. Open the Business Area/Entity Type matrix.

   Verify that all Entity Types are included in one and only one business area.

Chapter 7: Analyzing Current Systems and New Business Areas

This chapter discusses the various techniques used to analyze current information systems and business areas.

Using Matrices Analysis

Matrices Analysis helps you analyze current information systems, business areas, and distribution of systems in multiple locations.

Information systems analysis helps you specify function and Entity Type support needs, system information needs, and current information systems and data stores by location and unit.

Business area analysis helps you group business functions and Entity Types into natural business systems and natural data stores. You can verify the division of business areas into business systems, and you can divide the business into business areas automatically with Interaction Clustering.

Distribution analysis helps you determine the need for one-site or multi-site support.

Current Systems and Data Stores

Open the Current Information System/Current Data Store matrix. Ensure that:

- Each current information system uses at least one current data store
- Each current data store is used by at least one current information system

This guards against major deficiencies in the inventory of information systems and data stores.

Current Support of Business Functions

Open the Business Function/Current Information System matrix. Look for:

- Current information systems not supporting functions
- Functions not supported by current information systems

A system not supporting a function means either a business function is missing from the matrix, or the matrix does not identify completely what the business does.

A function not supported by a current information system means the current environment does not support the function, or the matrix does not identify completely how the systems support the functions.

Indicate a numeric degree of support for a function by a system between 1 (minimum) and 9 (maximum).

Entity Types Currently Supported by Data Stores

Open the Entity Type/Current Data Store matrix. Look for:

- Current information systems not supporting Entity Types
- Entity types that are not supported by current information systems

A system not supporting an Entity Type means either a business function is missing from the matrix, or the matrix does not identify completely what the business does.

An Entity Type not supported by a current information system means the current environment does not support the function, or the matrix does not identify completely how the systems support the Entity Types.

Indicate a numeric degree of support for an Entity Type by a system between 1 (minimum) and 9 (maximum).

Locations of Current Information Systems

Open the Current Information System/Location matrix. Use it for distribution analysis when you determine where to situate computing elements and where to make information systems available.

Organizational Units that Use Current Information Systems

Open the Organizational Unit/Current Information System matrix. Use it to determine where to situate computing elements and where to make information systems available.

Current Data Stores Impact

Evaluate how subject areas, locations and organizational units are impacted by current data stores.

Subject Area/Current Data Store

Open the Subject Area/Current Data Store matrix to review, at a high level, the data stores that support principal areas of interest to the business.

Current Data Store/Location

Open the Current Data Store/Location matrix and review the location of data stores. Use this analysis as you determine where to locate computing elements and where to make data accessible.

Organizational Unit/Current Data Store

Open the Organizational Unit/Current Data Store matrix and review the organizational units that use each data store. Use this as a completeness check and to compare the current and future system coverages.

Functions and Entity Types in Business Areas

Use the Business Area/Business Function matrix to assign functions to business areas. Use the Business Area/Entity Type matrix to assign Entity Types to business areas.

Information Needs Supported by Functions

Open the Business Function/Information Need matrix. Indicate a numeric degree of impact an information need has on a function between 1 (minimum) and 9 (maximum).

Current Systems and Business Areas

Open the Business Area/Current Information System matrix. Recommended cell values are:

- 1 = Business area is responsible for analyzing this system.
- 2 = Business area is responsible for analyzing this system, and the area currently is supported by this system.

Ranking Business Areas

Following are the ways to rank the business areas:

- Review the Performance Measure/Business Function matrix in conjunction with the Business Area/Entity Type matrix. Determine which business area, when implemented, will have the greatest impact on the performance measures.
- Review the Entity Type/Information Need matrix in conjunction with the Business Area/Entity Type matrix to assess which business area, when implemented, will provide the greatest increase in satisfaction of information needs.

Locations

Locations are important characteristics of organizational units, functions, activity clusters, and data clusters. You can evaluate how each is affected by its location.

Organizational Unit/Location

Use the Organizational Unit/Location matrix. To indicate that an organizational unit exists at a location, enter an X in the cell.

Business Function/Location

Use the Function/Location matrix. To indicate that a function is used at a location, enter an X in the cell.

Activity Cluster/Location

Use the Activity Cluster/Location matrix. To indicate that an activity cluster exists at a location, enter an X in the cell.

Entity Type/Location

Use the Entity Type/Location matrix. To indicate that an Entity Type is used at a location, enter an X in the cell.

Data Cluster/Location

Use the Data Cluster/Location matrix. To indicate that a data cluster is accessed at a location, enter an X in the cell.

Location/Location

Use the Location/Location matrix. Use cell values to identify traffic flow characteristics (quantity, periodic or continuous). Periodic traffic may lend itself to batch implementation, continuous to online.

Index

E

F

G

I

L

M

O

P

R

S

U