Rapid Prototyping of Application-Specific Signal Processors (RASSP)

The Library Management Model for the RASSP System Version 2

Advanced Technology Laboratories,
a Lockheed Martin Company
1 Federal Street, A&E Building
Camden, NJ 08102

Date:
June 12, 1995

Abstract

In today's design environments, the ability of the design engineer to maximize reuse is impaired by the fact that there is no efficient way of searching for reusable design objects across multiple sources; and the various sources of reusable data are uncoupled to the design environment. We describe in this paper the approach we have developed for managing reusable design objects in a system for Rapid Prototyping of Application-Specific Signal Processors (RASSP). Our approach consists of (1) developing a design object class hierarchy that classifies the various types of design objects in the RASSP domain, and models the descriptive data associated with the design objects, and (2) developing a commercial library management system which will implement the design object class hierarchy; and provide mechanisms for searching for design objects across multiple libraries, and across a virtual enterprise.

Introduction

The Rapid Prototyping of Application-Specific Signal Processors (RASSP) is an Advanced Research projects Agency (ARPA)/Tri-Service program aimed at dramatically improving the process of design, manufacture, test and procurement of digital signal processors. The RASSP program will deliver an integrated system called the RASSP system, which integrates the CAD tools used in the RASSP design process under a framework referred to as the enterprise framework. An *enterprise framework* provides the facilities and services necessary to integrate the automated processes of an enterprise. In the RASSP system the enterprise framework provides support for workflow management, design data management, library management, computer-supported collaborative work and remote tool access. The workflow management subsystem of the RASSP enterprise system enables a RASSP system administrator to model and enforce a particular design methodology for a project. The data management subsystem of the enterprise framework provides facilities for configuration managing, and controlling access to design data files that may reside at various sites in a computer network. Library management in the RASSP system involves the release, cataloging, and searching of reusable design objects. The library management subsystem of the RASSP system is called the RASSP Reuse Data Manager (RRDM). Sources for reusable design objects in the RASSP system include the following:

- CAD Tool Libraries
- CAD-Tool-Independent Libraries
- Component Vendor Data Books
- Design objects created within a design organization

In today's design environments, the ability of the design engineer to maximize reuse is impaired by the fact that there is no efficient way of searching for reusable design objects across multiple sources; and the various sources of reusable data are uncoupled to the design environment. Also lacking are mechanisms and processes for organizing reusable design objects created within a design organization, and effectively sharing the reusable design objects within the organization as well as with other cooperating organizations. We describe in this paper an approach for integrating the various sources of reusable design objects to provide a single source for searching for reusable design data, and enable enterprise-wide sharing of reuse data. Our approach consists of (1) developing a design object class hierarchy that classifies the various types of design objects in the RASSP domain, and models the descriptive data associated with the design objects, and (2) developing a commercial library management system which will implement the design object class hierarchy; and provide mechanisms for searching for design objects across multiple libraries, and across a virtual enterprise.

In the next section we describe the overall architecture of the RASSP enterprise system. In section 3 we present a scheme for classifying the integration between a CAD tool and the RRDM. In section 4 we present a workflow that may be adopted by an organization for inserting reusable design objects into the RRDM. The various data views of the RRDM is the subject of section 5. In section 6 we describe our methodology for developing design object class hierarchies for modeling descriptive data in the RRDM. In section 7 we describe how the reuse data and the associated descriptive data may be managed in the RRDM. We present the basic query capabilities of the RRDM in section 8. In section 9 we propose a model for cooperation of RRDMs distributed across a computer network. Facilities that need to be provided for release management are described in section 10. In sections 11 and 12 we present our implementation strategy, and describe an implementation of a CAD tool library integration that we have completed. We conclude the paper in section 13 with a summary of the current status of the RRDM implementation.

2 Architecture of the RASSP Enterprise System

Figure 1. The RASSP Enterprise System Data Flow Architecture

Figure 1 shows the data flow architecture of the RASSP enterprise system. A design engineer interacts with the workflow manager to perform activities in a workflow. The workflow manager in turn invokes the appropriate CAD tool for the particular activity, opening the product data files that are specific to the activity on hand. The workflow manager interacts with the product data manager on behalf of the user in a transparent fashion to open, close, save and configuration manage the product data files. The user may also interact directly with the desktop manager to invoke tools outside of the workflow environment. This enables users to do design actions that were not envisioned by the workflow designer. Once a CAD tool has been invoked the design engineer directly interacts with the CAD tool in its native environment to perform a design/analysis. The design engineer invokes the RRDM through CAD tools, and then interacts with the RRDM directly; or may invoke the RRDM directly. The RRDM receives a new reusable design object from a CAD tool, the associated meta data from the design engineer, and stores them in the Reuse Data Repository and the Reuse Metadata Repository, respectively. The workflow manager is implemented using the Intergraph Design Methodology Manager (DMM) [Intergraph, 1993], the product data manager is implemented using the Intergraph Data Manager 2.0 (DM 2.0) [Intergraph, 1995], and the RRDM is implemented using the Aspect Component Information System (CIS) [Aspect, 1994].

3 Levels of Data Integration

A major goal of the RRDM is to integrate the multiple sources of reuse data libraries to provide a single source for searching for reuse data. Once a reusable design object is identified using an RRDM search, the design object may be inserted into a CAD tool's design data space using either a manual insertion process, or an automated insertion process. A *manual insertion process* implies that the RRDM will provide information regarding the location and the format of the design object, and the design engineer will need to retrieve the design object, perform any necessary data translations, and insert it into a CAD tool's design data space. An *automated insertion process* implies that the RRDM will communicate with a CAD tool using an inter-process communication (IPC) mechanism to insert a design object into the CAD tool's design object space. Any necessary data translation will also be automatically performed.

Data translations may be required when reuse data created in a CAD tool and CT-A need to be used in another CAD tool CT-B, and the two tools use incompatible data representations. Data tanslations are performed using standard data representations as intermediate representations. For example, a data flow from CT-A to CT-B is supported by translating the design data produced by CT-A, from the native data representation of CT-A to a representation consistent with a Standard-X, and CT-B reading data represented in Standard-X, and translating it to the native representation of CT-B. A data flow from CT-B to CT-A is made possible using the reverse process. Examples of standards that may be used for data translation include the Electronic Data Interchange Format (EDIF)[EIA, 1993], the Initial Graphics Exchange Specification (IGES) [US PRO, 1993], and the CAD Framework Initiative Design Representation (CFI-DR) [CFI, 1994].

The Individual CAD tools and their associated libraries may be integrated with the RRDM in one of the three levels described below:

- Level 1 Integration: Descriptive Data Integration
- Level 2 Integration: Level 1 Integration plus Design Data Integration
- Level 3 Integration: Level 2 Integration plus Tool Integration

A CAD Tool CT-A is said to have a *level 1 integration* with the RRDM if the descriptive data repository of the RRDM is populated with the descriptive data associated with the reusable design objects in the CT-A library. This level of integration will allow the search of multiple vendor libraries from the RRDM. Once the appropriate design object is identified, the design object is inserted into CT-A's environment using a manual process. *Level 2 integration* of a CAD tool CT-A to the RRDM subsumes level 1 integration, and the reuse data repository of the RRDM is populated with the reuse data objects in CT-A's library. This will allow for the viewing of the design objects using CAD-tool-specific viewers, and viewers for graphical files represented in standard representations such as Postscript, and Graphical Interchange Format (GIF). A *level 3 integration* of a CAD Tool CT-A to the RRDM subsumes level 2 integration, and provides for automated insertion of reusable design objects from the RRDM to the CAD tool.

4 Building Reuse Libraries

Figure 2 shows the default workflow to be followed to add reusable design objects to the RRDM. We have used an IDEF3 [AL, 1992] notation to represent the workflow. The boxes represent individual activities in a process, and the links between the activities represent precedence relationships between activities. The links are annotated with information about the data that flows between two activities -- the state of the data and the type of data seperated by a "*". A junction box, represented by a box with a vertical line parallel to the left edge and an 'X' within the box, is used to model alternate paths within a workflow. The arrows coming into the bottom edge of an activity box indicate the mechanisms that are involved in the activity, usually the job classification of the individual(s) performing the activity. The workflow is implemented using the workflow manager of the RASSP enterprise framework and may be customized by a RASSP administrator for a particular organization.

Figure 2. The workflow for reusable design object definition

5 Data Views of the RASSP Reuse Data Manager

The RASSP Reuse Data Manager (RRDM) will provide multiple vies for the user into the reusable design object repository:

1. Descriptive Data view
2. Design Data View
    - Databook view
    - Package view
    - Logic Symbol view
    - Simulation Model view
    - Geometry view
    - Schematic/Drawing view
    - Pin Property view
    - Specification & Documentation view
    - LMS Catalog view
    - Test Data view

The Descriptive Data View of the RRDM is provided by the RASSP Design Object Class Hierarchy (RDOCH). The initial version of the RDOCH is shown in figures 3 and 4. The RDOCH models the descriptive data about the design objects that may be created/used within a RASSP Design Environment. The interior nodes of the hierarchy are abstract classes, and the leaf nodes are concrete classes include "Microprocessor", "SRAM", "Image_Processing_Software", etc. The hierarchy will be further refined and developed by analyzing how design objects in the libraries associated with system design tools, architecture design tools, hardware design tools, software design tools, etc. may be modeled.

The goal of the RDOCH is three-fold:

1. Provide a framework for classifying design objects in the RASSP domain
2. Capture all the descriptive data that is relevant for design objects in the RASSP domain using an object-oriented paradigm. Figure 5 shows the descriptive data schemas of four nodes in the RDOCH.
3. Provide a framework for searching for reusable design objects using an object-oriented paradigm. The user may execute a search based on the descriptive data, locate the appropriate design object, and then access the design data associated with the design object. The design data associated with a design object may include a data sheet, a simulation model, a schematic, etc. The methodology for the development of the RDOCH is described in the next section.

The *design data views* provide an alternate mechanism to access the design data. For example a user may go to an online data book and use the table of contents to locate an appropriate data sheet. The RRDM will also maintain limited descriptive data to locate the design objects within each design data view. Figure 6 shows an initial set of descriptive data for Packages and Data Books. The attributes "Function", "Manufacturer", "Use List", and "Change History" are part of the descriptive data schema of every design data view.

Figure 3. The RASSP Design Object Class Hierarchy

Figure 4. The Electronic Design Object Subtree of the RASSP Design Object Class Hierarchy

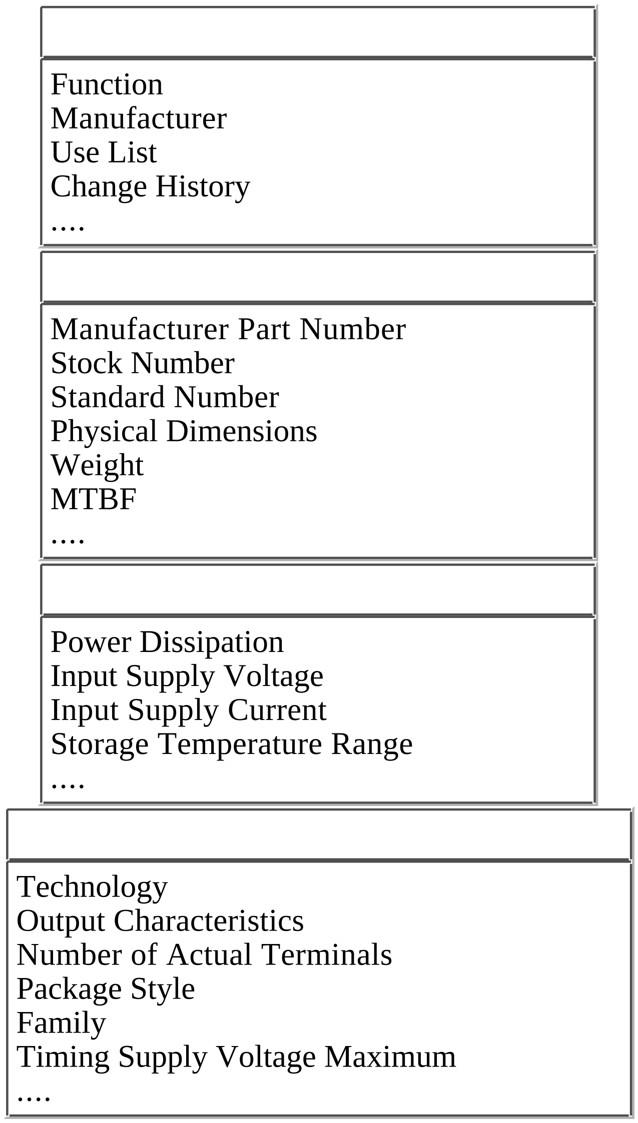

Figure 5. Example Descriptive Data Schemas for nodes in the RASSP Design Object Class Hierarchy

Function Manufacturer Use List Change History JESD-30 Code Surface Mount Package Style Package Body Material Package Shape No. of Potential Terminals No. of Actual Terminals Terminal Form Terminal Count Sequence Package ID ....

Function Manufacturer Use List Change History Document Title Publication Date Volume Order Number Application Information Data Book ID ....

Figure 6. Example Descriptive Data Schemas for Design Data Views

6 Methodology for Class Hierarchy Development

An important criterion for the RASSP Design Object Class Hierarchy (RDOCH) is that it should be general enough to be adopted as an industry standard. As a standard hierachy, it should lend itself for extensions without requiring destructive changes to the hierarchy. Destructive changes include deletion of classes in the hierarchy. Adding new classes to the hierarchy and adding attributes to the existing classes will be allowed. These restrictions are necessary so that future releases of the RDOCH will be upward compatible with the previous releases; i.e., previous integrations of CAD tool libraries with the RDOCH will continue to be valid.

To achieve the above goals we have adopted the methodology for reuse data classification described in International Electrotechnical Comission Draft International Standard 1360-1 for classifying electric components [IEC, 1994]. We summarize below the salient features of the methodology:

- select the type of reuse data to model
- source data from commercially available libraries
- determine the set of potential descriptive data attributes to be managed
- for each attribute, specify the semantics and valid values. The semantics of the attributes is captured by defining for each attribute the following information:
    1. Definition
    2. Note

3. Remark
4. Figure
5. Formula
6. Source Document

7 Reuse Data Management

Reuse data management in the RRDM involves both the management of the descriptive data about the reusable design objects, the management of the reusable design objects, and querying of reusable design objects that are part of the RDOCH.

An RRDM will be initialized with a standard RDOCH as described in section 5. A RASSP administrator at a particular organization may customize the class hierarchy to accomodate new types of design objects generated at the organization, or to add more descriptive data to an existing class. To be consistent with standard RDOCH the following types of changes to the class hierarchy are not permitted:

- deleting a class
- changing the superclass of a class
- changing the type of an attribute of a class
- deleting an attribute of a class

The RRDMs will provide facilities for creating instances of a class in the RDOCH, modifying the values of the attributes of the instances, and associating reusable design objects with instances of classes that model the descriptive data.

The RRDM will provide facilities to search for reusable design objects that are part of the RDOCH. A search will specify a class in the hierarchy, and the search may be a *deep search*, in which case the entire sub hierarchy rooted at the specified class is searched, or it may be *shallow search* in which case only the specified class will be searched. The query expressions will involve the attributes of the class on which the query is specified; and may include logical operators such as *not, and, or, and xor*; as well as comparison operators such as *less-than, greater-then, and equal-to*. Support for wild card completion of string-valued attributes of a class will also be provided.

8 Distibuted Architecture

The RASSP System supports a *virtual enterprise* where the different parts of the enterprise may be distributed across different sites in a wide-area network. Each site in the virtual enterprise may have a complete instantiation of the RASSP system with a suite of CAD tools, the RRDM and the product data manager. The reusable design objects generated at a site will be made known to the RRDM at that site.

The RASSP system will support two paradigms for interaction between RRDMs at the various sites in a wide-area network -- a tightly coupled federation of RRDMs and a loosely-coupled federation of RRDMs. All the RRDMs in a *tightly coupled federation* use the same class hierarchy and class definitions. Any updates done to the class hierarchy at one site will be propogated to the other sites in the federation as well. The reusable design objects inserted into the RRDM at one site will not be replicated at the other sites at the time of insertion. A reusable design object residing at site s1 will be replicated at another site s2 only if and when the reusable design object is inserted into a design object residing in site s2. In this case the descriptive data associated with the reusable design object is made part of the RRDM at site s2 as well. The RRDMs in a tightly-coupled federation are organized hierarchically with the federation at the root and the individual RRDMs as the leaves of the hierarchy. Thus a design engineer may query a group of RRDMs by specifying a node in the hierarchy.

The RRDMs in a *loosely-coupled federation* do no share a common class hierarchy or common class definitions. Thus a design engineer will not be able to query a group of RRDMs in the federation. The design engineer working at a site may however query the other sites in the federation one at a time. A reusable design object residing at a site s2 that is inserted into a design object residing in site s1 will be replicated at site s1. The descriptive data associated with the reusable design object cannot be automatically made a part of the RRDM at site s1, because of the differences between the descriptive schemas of the two sites. The librarian at s1 may however insert the descriptive data for the reusable design object, in the RRDM at the site s1 by executing the workflow shown in figure 2.

9 Release Management

Release Management in the RRDM involves the release of new versions of a reusable design object and the notification of the users of the design object about the new version. Associated with each reusable design object in an RRDM is a use list, which stores information about each instance where the design object has been used. Each record in a use list contains information such as the site id, the project id, the contact person, the id of the design object in which the design object has been inserted, date of use and the expiration date specified by the design engineer who uses the design object. The expiration date indicates the date after which the use record may be removed from the use list. A RASSP library manager may specify an upper bound and a lower bound for the duration for which use records need to be stored by an RRDM. These bounds will over ride the expiration date specified by the design engineer.

A design engineer may submit a new version of a reusable design object for inclusion in an RRDM. The new version of the reusable design object may represent a correction made to a problem that existed in a previous version, or it might represent a different implementation of the reusable design object. The RRDM groups the different versions of a reusable design object by assigning them the same design object id, but different version ids.

A version of a reusable design object may be deleted from an RRDM. If the use list of the design object is non empty, then it is marked as deleted and would actually be deleted only when the use list becomes empty. A reusable design object marked as deleted would not be included in the queries described in section 8.

10 Implemenation Strategy

The RRDM is being implemented by Aspect Development as an extension to the Component Information System (CIS), a commercially available product. In order to implement the RRDM, Aspect is making the following four object oriented extensions to the CIS system:

- data manager extensions
- object viewer development
- object editor development
- metamodel editor development

The data manager of the Cis system will be extended to manage default objects, template objects, and temporary objects. A default object is a model for the objects in a given class. The properties and values for the dafault object may be used as a basis for creating a new object. Template objects are added to the database, but marked to be deleted or moved at a later time. Temporary objects are useful for creating test cases for converting existing data.

The object viewer allows users to search for reusable design objects using a powerful parametric search. The object viewer uses two windows, one for entering parametric search data and another for displaying results. The result of such a search is a list of all individual reusable design objects that match the specified parameters. The user can then pick individual reusable design objects on the list and view all of the parameters and cross-references associated with each reusable design object.

The object editor allows users to enter objects into RRDM or to modify existing objects. Updates typically involve adding, deleting, or editing object parameters and test items associated with a given object, changing user permissions for a given object, and copying an object to a different class. The object editor also allows users to add mappings between objects.

The metamodel editor allows users, typically database administrators or librarians, to add and delete classes, that is, to modify the hierarchical class structure. The metamodel editor is being implemented during the first two builds of the RASSP enterprise system.

11 Implementation Case Study

Aspect Development and Mentor Graphics have jointly completed a prototype of an interface between RRDM and two Mentor Graphics tools, the Library Management System (LMS) [Mentor Graphics, 1994a] and the Design Architect (DA) [Mentor Graphics, 1994b]. This interface is an example of a level 3 integration (section 3) between the RRDM and a CAD tool.

The primary functions supported by this integration include:

1. the capability to launch the RRDM from within the Mentor DA/LMS tool suite and search for applicable reusable design objects
2. facilities for updating RRDM with information generated about reusable design objects in the LMS tool.
3. the capability to create/modify Mentor LMS descriptive data files based on data contained in the RRDM
4. the capability within the RRDM to maintain mapping information to indicate which RRDM attributes correspond to which attributes in the LMS descriptive data files on a per class basis. This is required for replicating updates made to the descriptive data in LMS to the RRDM, and vice versa.

12 Status

To date Aspect Development has completed the basic functionality of the object viewer and the object editor. Aspect and Mentor Graphics have jointly completed an initial integration of Mentor DA and LMS with the Aspect CIS. An initial RASSP Design Object Class Hierarchy has been completed (figures 3 and 4). The integration of CAD tool vendor libraries with the RRDM, and the extension of the RDOCH to include the design objects in the CAD tool libraries, are being initiated. Development of a prototype metamodel editor using a modified object editor has been initiated. The implementation of use lists, version management capabilities, and tightly-coupled federations of RRDMs are currently under way.

References

[AL, 1992] Armstrong Laboratory, *IDEF3 Process Description Capture Method Report* , AL-TR-1992-0057, Wright Patterson Air Force Base, Ohio, 1992.

[Aspect, 1994] Aspect Development, *CIS 2.0 User's Guide* , Mountain View, California, 1994.

[CFI, 1994] CAD Framework Initiative, *Design Representation Programming Interface* , Austin, Texas, 1994.

[EIA, 1993] Electronics Industries Association, *Electronic Design Interchange Format Version 3 0 0* , Washington D.C., 1993

[IEC, 1994] International Electrotechnical Commission, *Standard Data Element Types with Associated Classification Scheme for Electric Components -- Part I: Definitions, Principles and Methods* , Draft International Standard 1360-1, Netherlands, September 1994.

[Intergraph, 1993] Intergraph Corporation, *Design Methodology Manager -- User's Guide* , Huntsvile, Alabama, 1993.

[Intergraph, 1995] Intergraph Corporation, *DM/Manager -- User's Guide*, Huntsville, Alabama, 1995.

[ISO, 1993] International Standards Organization, *Product Data Representation and Exchange: Overview and Fundamental Principles*, ISO 10303-1, Fairfax, Virginia: U.S. Product Data Association, 1993.

[Martin Marietta, 1994a] Martin Marietta, "The Configuartion Management Model for the RASSP System", Moorestown, New Jersey, 1994.

[Mentor Graphics, 1994a] Mentor Graphics, *Design Architect Reference Manual*, Wilsonville, Oregon, 1994.

[Mentor Graphics, 1994b] Mentor Graphics, *Library Management System -- Users and Reference Manual for Designers*, Wilsonville, Oregon, 1994.

[US PRO, 1993] U.S. Product Data Association, *IGES -- Initial Graphics Exchange Specification*, Fairfax, Virginia, 1993.