CONNECTOR FOR DESIGNSYNC DFII

OBJECTIVE

Connector for DesignSync® DFII provides design data management for Cadence® data, in either CDBA (Cadence DataBase Access) or OpenAccess formats. It extends the design data management capabilities for **Software and System-on-a-Chip Designer.**

OVERVIEW

Connector for DesignSync DFII integrates Software and System on-a-Chip Designer with the Cadence DFII (Design Framework II) graphical integrated circuit (IC) design environment, also known as Virtuoso, recognizing and efficiently managing Cadence library design data. The Cadence DFII IC graphical design environment is modified with the addition of **Software and System-on-a-Chip Designer** menus and commands. Designers are able to perform Design Data Management (DDM) operations such as check-in, checkout, or tag without leaving the familiar Cadence graphical environment, and without having to manage the actual collections of files and directories on disk, which represent Cadence objects such as schematic diagrams.

EDA Data Awareness – Cadence Library Recognition

Data awareness is important because data created and modified by EDA (Electronic Design Automation) tools, such as a Cadence schematic diagram or physical layout, is typically not stored on disk as a single file. Rather, a design object such as a schematic diagram consists of a specific set of files. In order for a DDM system to maintain a version history of changes to the schematic diagram, this set of files must be managed as a group. The group of files is also referred to as a "co-managed" set, or "collection object." These collection objects are stored in a larger directory structure called a "library." A Cadence "library" consists of "cells," and a cell may contain multiple "views," which are models used for different purposes, such a schematic view, a layout view, etc.

Software and System-on-a-Chip Designer recognizes Cadence DFII libraries on disk so as not to confuse them with ordinary directories and files. Collection objects are managed transparently to the end user. So, when a user issues a command to checkout a version of a schematic diagram, the appropriate versions of each of the member files of the collection object are checked out automatically. The member files of the collection are each individually version controlled, and a mapping is maintained between the version of the design object and the versions of the member files which constitute the version of the object.

Storage of data in the DDM repository is efficient because only member files of a collection which are modified in an edit operation are stored in the new version of the design object. And because the design object is managed as a "collection," the tool prevents direct modification to individual member files, which can result in the corruption of the object as a whole.

The User Interface

With the **Connector for DesignSync DFII**, Cadence data recognition allows users to operate on familiar constructs such as libraries, cells, and views. **Software and System-on-a-Chip Designer** menus are included in the Cadence graphical environment and its commands appear in menus in the Cadence Library Manager tool. Commands are integrated in the Cadence environment using both the Cadence SKILL API, and the Generic Data Management (GDM) layer. Menus are added using the SKILL API, and AutoCheckout/AutoCheckin functionality is enabled through the GDM layer. Where appropriate, Synchronicity® commands are overlaid in Cadence menus in the Library Manager tool.

In addition to being able to operate on libraries, cell, and views, DDM operations may also be performed on a data "Category." For example, if a standard cell library has been categorized by types such as "FlipFlops", all the "FlipFlops" could be checked out for edit in a single operation.

A unique capability is also provided to perform data management operations based on the "hierarchy" of the design. Knowledge of a design is obtained by walking its instance hierarchy using Cadence SKILL functions. For example, a Cadence library may include ALU and MULTIPLIER design blocks. If the top level ALU schematic is fetched, one might issue the "Synchronicity > Tag > Hierarchy" command which would identify and tag each version of each instance of the lower level schematics. The level of hierarchical depth can be controlled. For example, one might want to tag the hierarchy of a standard cell design, yet not descend into the transistor level representations of the standard cells themselves. Once a hierarchy is tagged, it could be fetched into a new workspace.

The result would be that one or more libraries could be fetched, with valid Cadence library structures created on disk, but the local libraries would only include the cells/views which constitute the ALU design hierarchy. This is an example of the capability to construct a workspace with a subset of the data contained Cadence libraries as stored in the DDM repository.

Some operations are more efficient if performed using **Software and System-on-a-Chip Designer** products rather than running the commands from within the Cadence environment. For example, when starting a new project, it is typical that multiple libraries are put under revision control. Due to the restriction within Cadence that one must select a library before performing any operations, checking in multiple libraries requires that each be checked in separately. If a project consists of 50 libraries, for example, this is a tedious operation at best. Using **Software and System-on-a-Chip Designer** products, all 50 libraries can be checked in by running a single command.

HIGHLIGHTS

Key features and capabilities include:

Cadence Library Recognition and Processing

Cadence library directory structures are recognized as such. Users manipulate familiar constructs such as cells or views, while the tool automatically processes the underlying files and directories which comprise data views. Designers needn't be concerned with detailed version control bookkeeping which is handled automatically. If design data is managed in a module, the "whereused" command provides a report indicating all of the modules and member configurations that contain specific design object version.

Locking Model Enforced

Because Cadence design data is binary in nature, a strict locking model is enforced. This prevents the situation in which two designers are making changes to the same version of a cell view, because once the first designer checks in the changes, there is no automated capability to merge the changes of the second designer.

Operating on Libraries

DDM operations may be performed on an entire Cadence library. For example, a library may be tagged with a label which signifies a given release, or quality level. Or a library may be updated with changes checked in by other team members.

Operating on Cells

DDM operations may also be performed at the library "cell" level. For example, all views, or a subset of the views of a cell could be checked out for edit.

Operating on Views

DDM operations are commonly performed on individual data views in the course of daily design activity. A "Synchronicity" menu in each Cadence graphical editing tool such as Virtuoso provides convenient access to DDM commands such as check-in, tag, or version history.

Key Benefits:

- Industry leader in the management of Cadence DFII design data
- Integrated into the Cadence DFII graphical design environment — designers work in the tools with which they are familiar
- Aware of the unique structure of a Cadence data library
- Manages Cadence data, along with non-Cadence project data, letting companies connect to and manage the entire design chain with a unified design data management system

Operating on Categories

If Cadence cell data is organized using "Categories," DDM operations may also be performed on the categories. For example, if the cells in a standard cell library have been categorized, all the cells in a category could be checked out together.

Operating on Design Hierarchies

A unique capability enables the processing of a design hierarchy. A hierarchy is defined as the set of instance records contained in each level of a design hierarchy. If a design hierarchy is tagged, for example, a workspace could be created which contains the set of libraries which store the design data, yet each library would contain only the cell views which comprise the hierarchy of the design. Other cell views which are stored in the server which are not part of the specific design hierarchy would not be fetched.

Operating on Files

Some DDM tools which are integrated in the Cadence design environment do not provide access to manage files which are not contained in a Cadence data library. An example might be a "cds.lib" file which stores a mapping between library names, and the location on disk of each library. While not contained in any library, this file is required by Cadence to open the library in the graphic environment. A "File" selection in the "Synchronicity" menu provides access to any file located in a workspace, such that data, which is vital to the project, but is not located within a Cadence library, can be managed as well.

Operating on Subsets of Data

It is not necessary to fetch an entire Cadence library into a workspace in order to work on a subset of the data. Only the data of interest can be fetched. A valid Cadence library structure is created in the workspace, but the local library contains only a subset of the data as stored in the managed library in the data repository.

Library Status Browser

The Library Status Browser displays design management information which cannot be displayed in the Cadence Library Manager, such as version, check-in status, date/time of last update, "locked by", etc., and includes the most common Connector for DesignSync DFII commands. The Status Browser also displays information about modules which have been populated into a workspace. Icons provide an intuitive graphical indication of the status of a managed object such as whether it is locked, up-to-date, locally modified, or unmanaged. If "locked" and "locally modified," the icon for check-in will be displayed.

Graphical Comparison of Views

Any two versions of a view, or any two independent views, may be graphically compared. Differences are highlighted using a haloing effect in any window that displays the view. It finds added, modified, or deleted graphical objects, changes to attributes or properties for instances, and path styles and label texts for physical objects. A description of the differences can also be output to a test file.

Pre-defined Tag Lists

Lists of tags can be pre-defined, making them available in a pull-down list for the tag command. Users can select tag names from the list, maintaining consistent naming conventions, and eliminating potential spelling errors.

Status Information Displayed in Editor Banners

When a data editor such as Virtuoso is open, revision control status information is displayed in the banner. The information displayed is configurable, and can include the cell view version number, version tags, branch tags, lock status, and modification status.

Viewing Workspace or Vault Status

The status of data located in a workspace, or in the data repository, may be viewed. Viewing status in a workspace lists cell views which have been locked for edit, and reveals whether or not views have been locally modified in the workspace. All modified views can be selected for check-in in a single operation. When the data repository is viewed, all views which are checked out for edit are listed.

Viewing Version History

The complete version history of a cell view may be viewed.

Branching a Library

Support for branching scenarios is included. A library may be branched using the "Synchronicity > Branch Library" menu entry. Branching allows for modifications to be made in isolation from the main branch of development, perhaps to fix bugs, or to explore alternatives.

Opening Multiple Versions of the Same View

Two versions of the same Cadence view may be opened simultaneously. If one view is opened for edit, data can be copied from the other. This is particularly useful in a branching scenario when data from one branch needs to be merged into the data on another. Because the underlying data is binary, and thus cannot be merged using automated means, the ability to have two graphical versions of a cell view open simultaneously provides the means for performing the merge manually.

Distributing a Library across Multiple Data Repositories

A single library can be distributed across multiple data repositories by using **Software and System-on-a-Chip Designer** module-based data storage. Local efficiencies are maximized at each design center because the majority of data transfer activity (check-in/checkout) occurs locally.

GUI and Command Line Support

Although most day-to-day data editing operations are performed from within the Synopsys Custom Designer graphical design environment, DDM operations on Synopsys library data can be performed using either the **Software and System-on-a-Chip Designer** GUI or command line. Multiple libraries can be processed using one command, which is not possible from within the Synopsys environment.

Documented SKILL Functions

A SKILL Programming Interface Guide contains descriptions of the SKILL API functions which are available for use in the Cadence DFII design environment. SKILL variables may be set to customize your environment.

Our **3D**EXPERIENCE® platform powers our brand applications, serving 12 industries, and provides a rich portfolio of industry solution experiences.

Dassault Systèmes, the **3D**EXPERIENCE® Company, provides business and people with virtual universes to imagine sustainable innovations. Its world-leading solutions transform the way products are designed, produced, and supported. Dassault Systèmes' collaborative solutions foster social innovation, expanding possibilities for the virtual world to improve the real world. The group brings value to over 190,000 customers of all sizes in all industries in more than 140 countries. For more information, visit **www.3ds.com**.

3DEXPERIENCE®

DASSAULT SYSTEMES | The **3D**EXPERIENCE® Company

Americas
Dassault Systèmes
175 Wyman Street
Waltham, Massachusetts
02451-1223
USA

Europe/Middle East/Africa
Dassault Systèmes
10, rue Marcel Dassault
CS 40501
78946 Vélizy-Villacoublay Cedex
France

Asia-Pacific
Dassault Systèmes K.K.
ThinkPark Tower
2-1-1 Osaki, Shinagawa-ku,
Tokyo 141-6020
Japan