Migration from DFII 4.3.4/SunOS 4.1.3 to DFII 4.4.1/Solaris 2.5.1

Thomas Harriehausen

Siemens AG, Semiconductor Division
HL CAD SYS
P.O. Box 80 17 09, D-81617 Munich, Germany
Phone: +49 89 636-24762
Fax: +49 89 636-23650
Thomas.Harriehausen@HL.Siemens.DE

International Cadence User Group Conference
October 5-9, 1997
San Diego, California

Migration from DFII 4.3.4/SunOS 4.1.3 to DFII 4.4.1/Solaris 2.5.1

Thomas Harriehausen
Siemens AG, Semiconductor Division
HL CAD SYS
P.O. Box 80 17 09, D-81617 Munich, Germany

Keywords

Migration, Solaris 2, automounter, CDE, DFII 4.4, TDM, Design Data Management, ClearCase, SKILL

Abstract

The performance of Sun's recent UltraSPARC UNIX workstation family made many of our designers request this new hardware. With Solaris 2.5.1, Sun offers a stable operating system for it. As a result, Siemens Semiconductor's central CAD department in Munich has to provide its proprietary IC development system "ASCIA" for a mixed SunOS 4.1.3 and Solaris 2.5.1 environment.
Currently, IC design tool from Cadence's DFII 4.3.4 are part of several ASCIA subflows. Meanwhile, DFII 4.4.1 seems to have reached a level of stability that allows its productive use. Thus, a new DFII 4.4.X based fullcustom design flow is currently under development in the CAD department. TDM was chosen to manage all design data in the new flow.
With this contribution, we want to share our considerations and first results related to this double migration process with other Cadence users.

1 Introduction

1.1 ASCIA 4.1

The architecture of Siemens Semiconductor's ("HL's") proprietary "all-in-one" IC development system "ASCIA 4.1" is described in /1/. Meanwhile, more than 1000 designers at almost 30 locations worldwide are using this system for analogue, digital and mixed-signal designs. This means that ASCIA is one of the world's most widely spread productive IC development environments. ASCIA supports fullcustom and semicustom design methodologies and contains device, standard cell and macro libraries for many technologies like bipolar, CMOS (analog, digital), BiCMOS, smart power and several special purpose technologies.

Originally, it was planned to make ASCIA 4.1 an "all Cadence" environment to minimise tool/data interface problems. Since we need a leading-edge system, we have to integrate more and more "best-in-class" tools from third parties. Meanwhile, components from more than a dozen EDA tool suppliers plus much proprietary software are part of our system.

1.2 Hardware, Operation System, GUI

The hardware and operating system for which ASCIA 4.1 was initially released (in 12/94) are Sun Workstations running the BSD based SunOS 4.1.3. The GUI used is OpenWindows 3. A year ago, our custom-

ers were mainly using Sparc 20 workstations for computing purposes and design data storage.
In 10/95, we began planning the migration of ASCIA 4.1 to the SystemVR4 based SunOS 5.5 which is mandatory for Sun's new UltraSPARC workstations[1]. We expected that ASCIA 4.1 (consisting of kernel, tools and libraries) would be available for Solaris 2.5.1 at the end of 1996.

1.3 DFII

In this paper, the term "**DFII**" relates to the whole IC design tool suite from Cadence. The DFII generation currently released in ASCIA is 4.3.4. In 11/95, we presented a draft project plan for migration of ASCIA 4.1 to DFII 4.4 to our local Cadence office because we knew we would need intense support from Cadence. We intended to first make ASCIA 4.1 also available for Solaris 2 and afterwards build a new Solaris 2 based design system called "ASCIA 5" which should contain the DFII 4.4 design tools and use TDM as data management tool.

2 Migration from SunOS to Solaris

In this document, the term "**SunOS**" will be used for "SunOS 4.1.3" while "**Solaris 2**" will be used for "Solaris 2.5.1/SunOS 5.5.1".

2.1 Motivation

There is a very simple motivation to use UltraSPARC workstations instead of SuperSPARCs or HyperSPARCs: They are faster (factor 2-3 compared with a Sparc 20) without additional cost. Also, Solaris 2 is running quite stable and contains several enhancements concerning architecture and system administration.

Most of our VLSI designers always want to work with the fastest available workstations. The problem the CAD department faced was that many of these users were not aware of the fact that Solaris 2 is not fully compatible to SunOS. Simple UNIX commands which are contained in many ASCIA and user specific scripts like "ls", "ps" and "mail" have to be handled differently. Sun's "Binary Compatibility Package" (BCP) provides some means to execute old binaries under Solaris 2. But this does not help always. Some of our older design tools don't run under Solaris 2.

Another reason to switch to Solaris 2 is the new industry standard GUI CDE (Common Desktop Environment) which is Motif based and will replace Sun's OpenWindows in the long term. Generally, we are very satisfied with OpenWindows 3 and Sun's OpenLook desktop utilities. But Cadence's DFII performs better (e.g. cut&paste) and faster in a Motif based environment.

To minimise our QA and support effort, we decided to:

- Adapt all our own ASCIA software source code so that it runs under both operating systems.
- Try to get identical tool versions of all commercial ASCIA tools for both OS from the EDA vendors and make all workstations automatically use the software variant which fits its operating system.

---

1. By the way: Many of our users were not aware of the fact that Solaris 2 can also be installed on the existing SuperSPARCs.

2.2 Experiences

2.2.1 Automounter Problem

In 1/96, we started our UltraSPARC/Solaris 2.5 evaluation. First tool tests showed amazing performance increases. But soon, we ran into a problem which seemed "killing" for our plan: The new "automountd" did not handle hierarchical mounts correctly. The effect was that required software components suddenly "vanished" from the end user's point of view so that CAD system components seemed to be missing. When we reported this bug to Sun, we heard that it was a known one (bug-ID 76506082) and that we would get a bugfix soon. The patch we received didn't work. As a workaround, we had to disable the automounter's unmounting. This causes additional UNIX system administration effort.
In 2/97, we verified that the bug is fixed in Solaris 2.6-beta and requested a backport to Solaris 2.5. In 10/96, Sun promised to deliver it till 9/97.

2.2.2 Printing

Printing was an other big problem in our mixed SunOS/Solaris 2 environment. It only could get solved by installing the BSD based spooler "LPRng" /3/ which is public domain software.

2.2.3 Migration of proprietary Source Code

We received special support from Sun for migration of the proprietary parts of the ASCIA software (which is mainly implemented in C, csh, SKILL and Perl) because we were one of the first big customers tackling the migration. The ASCIA 4.1 kernel was available for Solaris 2 in the beginning of 1996. Since then, the portion of Ultras used by our designers has been increasing dramatically.

2.2.4 DFII 4.3.4 in a mixed SunOS/Solaris 2 Environment

We had to learn that it is impossible to get functionally identical versions (including up-to-date hotfixes) of our EDA tools for the two OS variants. The most complex case is the DFII software. Cadence told us that there is no process in place to provide (almost) identical hotfix streams for the two operating systems. But they are willing to provide this service through consulting work. Currently, it looks like we will have to pay Cadence extra money to get what we need. Is our request so unique?
The most recent DFII version available in ASCIA is 4.3.4.50.154 for both OS versions. But hotfixes for some tools have more or less different versions. The effect is that the behaviour of some of the DFII tools depends on the workstation a designer uses. This is exactly what we wanted to avoid.

When we started testing DFII 4.3.4 in a mixed SunOS/Solaris 2 environment, we faced problems similar or identical to those described in /2/. I do recommend to read this document very much.

Unable to create CDBs on Solaris 2 Machines

Our first Solaris 2 pilot users weren't even able to create CDBs with a DFII running on the local host. This was a Cadence locking daemon (cdsd) bug which is fixed since 4.3.4.50.137.

Loopback File System Problem

Background: When Cadence introduced the new “filepath” and “filehost” library properties in a hotfix release of DFII 4.3.2, we had to deactivate their evaluation: Many DFII users complained about access problems after moving/copying CDBs. They were confused by the strange message in the CIW or ignored it and could not work any more. If Cadence had spent just a few words of explanation in the message printed to the CIW about how the “dmSetLibLockPath” command has to be executed or provided a simple GUI to semi-automatically perform the task, the deactivation would not have been necessary.
Mid 7/96, we were directly informed by Cadence US via the Mixed Signal TAB about potential CDB corruption caused by the “loopback file system (lofs) problem” in the Solaris 2 version of DFII. After we pointed out to our local Cadence office that we use lofs and that some CDB corruptions were probably caused by this bug, we received a hotfix for it. We were blamed by Cadence that the problems would not have happened if we had not deactivated checking of the above mentioned properties.

Timeout during Spectre Netlist Creation

This bug (local machine SunOS, rlogin to Solaris 2 machine, display redirection to SunOS machine, start netlister on remote machine) occurred in our environment in 1/97. The message “waiting for m6-file to finish” is printed about 20 times before a time-out. It is caused by an IPC problem which may potentially affect all simulators. We were told that this is a long-known problem (PCR 171123) which is caused by a bug in Sun’s OpenWindows server and thus cannot be solved by Cadence. Interestingly, the bug is fixed in DFII 4.4.1 by re-implementation of the layer to use sockets. But the code is said to be not back-portable to 4.3.X. Cadence: “There are only the workarounds either to use Solaris or SunOS on both ends.” The solution would be to use Standard MIT X. As Sun’s X11/NeWS is blamed to be the problem, we asked whether Cadence has reported the problem to Sun. After a long time, we found out: no, they haven’t and they don’t plan to do so!

Stranded Locks on Solaris Machines

When a DFII tool running on a Solaris 2 machine crashed, locks on CDBs were not released. This bug could be fixed by installing a new (4.3.4.50.142 or later or from 4.4.X) version of the cdsd on all Solaris 2 machines. We solved this problem in 12/96.

We wonder why Cadence has not managed to collect all the internally available information about possible problems and solutions when running DFII 4.3.4 in a mixed SunOS/Solaris 2 environment and provided a “cookbook” for their customers' CAD system administrators. It looks like most of the AEs have to go through a similar painful learning process together with their customers. Cadence's suggestion “move to DFII 4.4 to get rid of the problems” did not help us at all.

3 Migration from DFII 4.3.4 to DFII 4.4.1

3.1 Motivation

Even before the official release of DFII 4.4 with 9504 in mid 1996, some Cadence guys told our customers that they needed DFII 4.4 as soon as possible because of its functional enhancements. They avoided to explain that as a prerequisite, the whole design environment has to be converted. Because many custom-

ers felt that the move from 9502 to 9504 could not be bigger than from 9404 to 9502, the CAD department was put under pressure to offer "the new powerful DFII 4.4 tools" as soon as possible.

As a result of our tests of a DFII 4.4 beta version in 1/96 which were already presented in /1/, we decided to wait for DFII 4.4.1 rather than playing the ungrateful role of a DFII 4.4 pioneer. It seems this was a good decision. Meanwhile, our local Cadence AEs and consultants had the chance to get acquainted with TDM so that they are now able to answer our questions.

In the CAD department, we see three reasons to migrate:
- New (5.X) architecture of the Cadence libraries (CDBs)
- Availability of the design data management tool TDM
- Access major functional enhancements of DFII tools coming in the future

3.2 Experiences

3.2.1 Management and QA of DFII Hotfix Streams

In the CAD department, we have one tool supporter for each released CAD tool. Several different people are responsible for the DFII tools we offer in ASCIA. Until now, we have invested many man-years of work for testing, ordering bugfixes and re-testing the most recent version of the DFII 4.3.4 software. Our parallel QA approach (everybody starts testing at the same time until a killing bug is found, after bugfix is installed, all people possibly affected have to start testing from scratch again) has to be improved.
Together with local Cadence consultants, we want to set up a more sophisticated QA concept to reduce our effort. A prerequisite is reliable information from Cadence about the tools likely to be influenced by the contents of the hotfix tarkit we receive. We are looking forward to see how this information can be provided.

3.2.2 Cadence Libraries (CDBs)

The few (but important) design data views in ASCIA which are stored in CDBs are the ones which cause most trouble. Although an experienced Cadence consultant told us that he has seldom seen corrupted CDBs, they are part of our daily business. The easiest way to restore a corrupted CDB is to use a backup. But a UNIX-level backup of a CDB which is currently being edited will often be corrupted, too. So we wrote our own CDB backup software. Dozens of Cadence users must have done so before. We wonder why Cadence is not willing to provide such functionality as part of the standard software.

A 5.X CDB is no longer a "real" database but a structure in the UNIX file system. We expect that access to CDB data in multi-user mode will become much faster and that fatal CDB corruption problems of the kind we know will no longer occur. But sooner or later, a portion of our users will recognise that manipulation of a 5.X CDB will no longer destroy it immediately. This will create a big challenge for our and Cadence's support people.

3.3 Migration of SKILL Code

ASCIA 4.1 contains far more than 100,000 lines of SKILL code in about 50 individually versioned packages. Because our SKILL code represents a value of several million dollars, we are shipping it as autoloading context files.

Most of these packages will still be needed in our DFII 4.4 environment. Exceptions are e.g. the CDB DDM customisation packages. Contrary to Cadence's advice to use the 4.4 migration to re-implement all productive SKILL code using the new SKILL features (e.g. SKILL++, MP-SKILL), we decided to just substitute all code segments which will no longer work in 4.4 in the first step. Cadence's system architects seem to assume that all 4.3.X SKILL code is deleted after migration to DFII 4.4. We did not find or hear of any concepts how parallel 4.3.X/4.4.X SKILL code streams should be managed. Therefore, we checked several migration scenarios:

1. Duplicate the packages and start an independent development branch for DFII 4.4
2. Keep a common source code basis within each package version from which DFII version specific code is generated after each change via a postprocessor
3. Separate code segments which are DFII version dependent from the DFII version independent ones. Keep the version dependent code segments in separate files but within the same package. Decide at loading time of a package which code variant (i.e. set of files) is used.

The first alternative is best suited for the development phase because it offers a maximum of flexibility and a minimum of impact to the ASCIA 4.1 community.
The last approach is preferable if functional compatibility between ASCIA 4.1 and the new fullcustom flow is desired.

During our DFII 4.4 beta test, we ran into several SKILL problems /1/ which we reported to Cadence. Meanwhile, Cadence's SKILL migration utilities were enhanced. Our local Cadence experts assured us that the migration would be almost straight forward now.
Since we decided to let Cadence do the SKILL migration, we hear that many problems have to be solved which were not expected in the beginning. To minimise effort, the first of the scenarios mentioned above was chosen.

4 TDM Introduction

4.1 Role of Design Data Management

We are living in a strange era: On management level, EDA visionaries praise the "plug and play" approach of IP blocks to build "systems on a chip". At the same time, EDA tool users are hampered by an ever increasing number of vendor/tool specific data formats and databases. The lack of standards and powerful design data management functions is one of the biggest deficiencies of today's EDA software.
CFI didn't help. Will VSIA help? The problem is not solved by defining VDHL or Verilog as "the" design data standard because it covers only digital designs. I am convinced that design of purely digital chips will be an ability of tenthousands of people worldwide in a few years. It will be very difficult to earn money in that market. But for many years from now, design, production and test of complex, optimised, mixed signal or RF ICs will remain an art only few design groups in the world are capable of.

Meanwhile, many of our IC designers and managers have understood the importance of design data management (DDM). The uniform structure of all technology and design libraries for all flows in ASCIA 4.1 is beginning to be accepted as an ingenious feature and not just a matter of course. Without this architectural property, easy design reuse and the option to select an optimised flow for each application domain would not be possible.

On the other side, many EDA tool vendors (like Cadence and Synopsys) are pushing the approach of "lean", application domain specific product development environments ("PDEs"). For a company like Siemens Semiconductors which offers a broad product line and thus needs several different IC design flows, this approach contains two risks:

- Firstly, effort for creating and maintaining individual environments will multiply efforts in the CAD tool and library groups (and thus - consulting turnover of the EDA tool vendors). This effort may be over-compensated by the increased productivity of designers working in such a specialised PDE.
- Secondly, any work flow or design reuse approach which requires traversing more than one of these specialised PDEs may face tremendous obstacles because different entities of design data have to be exchanged between different DDM systems. This means additional implementation effort, risk of data inconsistency and reduction of user's productivity due to necessary data conversion steps.

Thus, we would like to preserve a common infrastructure including the DDM level for all our flows.

4.2 ASCIA 4.1's Design Data Management

In ASCIA 4.1, DDM features like design partitioning, individual workspaces, versioning and configuration management are implemented on UNIX level without any special software. This approach requires only little administrative overhead and allows the fastest possible data access. The drawback is that advanced DDM features are not available.

4.3 Requirements

Together with Cadence, we have executed a DDM requirement survey among the worldwide ASCIA 4.1 user community. It was based on a systematic questionnaire and resulted in the formulation of 25 DDM/ project administration related enhancement requests. Some very important ones are:

- It should not be necessary to duplicate (sometimes big amounts of) data in case of small differences between versions of design data entities
- It must easily possible to define configurations of data to e.g. handle multiple variants of a design
- For work-in-progress projects, a differentiation must be possible between users who are editing the design data and users who may only reference (i.e. read) it
- VHDL code development must be supported
- All current ASCIA 4.1 data use models must be available, again
- Support of globally distributed design teams which are only connected by a WAN
- The DDM tool must be 'core competency' of its supplier
- Tool must be able to work with DFII and handle CDBs
- Easy to use by designers
- Acceptable performance, even if several tenthousand files must be managed
- Uniform data management for all views
- Low installation and customisation effort
- Powerful administrative utilities must be in place for backup, integrity check, repair etc.
- Interface to design flow management and design tracing software must be available
- Interface to defect tracking system (e.g. ClearQuality) must be available

4.4 Solution Alternatives

We have been looking for a DDM system which meets the requirements listed above. As expected, we didn't find the ideal solution.

4.4.1 Standard UNIX Tools

Although many of our VHDL designers are using them, sccs/rcs/cvs are not powerful enough. We don't want to implement and maintain a system on top of one of these tools which makes it useful for our more general purposes.

4.4.2 ClearCase and related Products

ClearCase from PureAtria /4/ (now Rational Software) is a widely spread data management tool for software development. When we asked them, both PureAtria and Cadence did not know of any customer in the world who has successfully managed Cadence CDBs with this tool.
We heard that a ClearCase database (VOB) becomes very slow in case it contains more than a few thousand objects. We verified this behaviour. But we also found out that the problem can easily be solved by splitting up one VOB into several smaller ones. Drawbacks of ClearCase are the facts that it requires modifications of the UNIX kernel because it uses a special file system (MVFS) and that the data in VOBs is inaccessible in case ClearCase does not run. (The latter is just normal for a real DBMS.) A very interesting feature of ClearCase is its interface to "ClearGuide", a workflow management system. PureAtria's product "MultiSite" on top of ClearCase might be a solution for our distributed design teams.

4.4.3 TDM

Cadence has repeatedly declared that they don't consider DDM part of their core competency. An early version of TDM in DFII 4.4 which we evaluated /1/ showed severe deficiencies concerning performance, stability and documentation. Until today, we did not find any documentation about TDM's "philosophy" (or isn't there any?) or how to make best use of all the features and implementation alternatives TDM offers. Our doubts about TDM's future within Cadence were answered in 6/96 by an official commitment signed by James H. Hogan stating "Design Management support is an essential part of the Cadence technology offering. Cadence has committed to include this functionality as a core competency in its five year technology roadmap." And we also learned "Cadence currently is investigating the integration of high-level, commercial Design and Project management environments as an extension to DM. A clear roadmap will be available in September this year." We haven't seen that roadmap till today. We consider Mr. Hogan as our guarantor of TDM's future at Cadence.

4.5 Our Choice

Did we have a choice? The DFII 4.4.1 tools can only use TDM as DDM tool. So for the new DFII 4.4 based fullcustom flow, we had four alternatives:

- Use no DDM at all. This would have been a clear setback compared with ASCIA 4.1
- Use TDM and customise and extend it to fulfil our customers' needs. We want to minimise customisation effort. This is an unsolvable contradiction.

- Try to use ClearCase instead of TDM. Although this might theoretically be possible, it would require enormous effort because we would have to mimic TDM's interfaces for the current DFII tools. We would have to teach ClearCase how to handle CDBs.
- Use ClearCase on top of TDM. Both tools are powerful and complicated. Both tools require intense user training. Introduction of just one of the tools makes things complicated enough for the CAD people and our users. This approach is not feasible.

Naturally but not happily, we decided to implement the second alternative. But we will try to keep our investments in TDM customisation as low as possible because of TDM's potentially transient nature. Cadence advised us to use both "base TDM" and "full TDM" to keep the amount of policies needed to implement our data use models minimal.

We have chosen ClearCase as DDM tool for our new HDL/synthesis based flow which is almost "Cadence free" and thus does not need to handle CDBs.

4.6 Data Use Models in ASCIA and their Re-Implementation with TDM

In ASCIA 4.1, three data use models concerning CDB data are implemented via SKILL code packages[1]. Common to all use models are the following UNIX-level settings:

- Only members of a projects's team have access to the project's data at all.
- All ASCIA subproject version specific configuration data (e.g. technology used, reference libraries available) is read-only for designers and can only be edited by a project administrator.
- ASCIA specific tool wrappers extract the information they need automatically from the ASCIA specific configuration files mentioned above. E.g., the DFII's library search path is automatically set up according to the ASCIA reference libraries defined in the configuration files.
- All design data is readable and writeable for all team members.

We will try to re-implement the following ASCIA data use models with TDM. Some of the most fundamental changes are

- ASCIA's versioning of design data on UNIX level will be replaced by "full TDM" releases.
- While the DFII only managed CDBs in ASCIA, TDM will manage all design data in the new fullcustom flow. Locking of files (i.e. handling of multi-user access) will still be only performed for the CDB data processed by the DFII tools.
- Definition of the CDBs visible for a designer will become more complicated. From a generic configuration file similar to the one in ASCIA, a "project.lib" file must explicitly be derived by the project administrator with the names of all CDBs contained in the reference libraries (at the time the file is derived). In each workarea, the "cds.lib" file (which is owned by the designer) will "SOFTINCLUDE" the "project.lib" file (which is owned by the project administrator) and additionally list all CDBs in the workarea and all CDBs in the local design which shall be shared by all designers. If a designer creates a new CDB which shall be visible for the other team members, s/he has to perform a "tdmsubmit" command., All other designers who want to see that CDB immediately will have to perform a "tdmup-

1. Please remember that only a few of ASCIA's views are stored in CDBs. Handling of the views stored outside CDBs on UNIX file system level is not affected by these use models.

date" and include a reference to that CDB in their "cds.lib" files. This a clear setback compared with the easy mechanism in ASCIA/DFII4.3.4 which assured that after start of the DFII, a user will automatically have an up-to-date list of all CDBs available.

- We don't know yet how to consistently remove a file from TDM's data repository.

4.6.1 "Sharing of Checkouts" (DMS)

This use model provides only very little DDM comfort. But it is almost fool-proof and necessary for those users who think they don't need to learn how to work efficiently with a powerful and complex system like ASCIA. Furthermore, this is the only reasonable use model for designs which are developed by globally distributed teams which have to exchange/replicate their data between sites automatically, e.g. using UNIX "rdist". DMS implements a "sandbox" like use model. All designers may read, edit and delete all objects (checked in or checked out). The effect of DMS is that at each point in time, exactly one common configuration of the CDB's design data exists. The biggest drawback is that wrong data one designer saves affects the rest of the team almost instantly. One of the major advantages of DMS is that people who have never heard of "check-in" can successfully contribute to a design.

DMS shall be re-implemented with "full TDM" by using one common workarea for all team members. We will have to find a policy when and by whom an update of that common workarea shall be allowed.

4.6.2 "Readonly Checkouts" (DMR)

This use model is almost identical to DMS. The differences are that checkouts can only be edited and checked-in objects can only be deleted by their owner.

DMR shall be re-implemented using "full TDM" similar to DMS with the restriction that design team members do only have read access to other team member's files on UNIX level.

4.6.3 "Private Checkouts" (DMP)

This use model is only available for very small design teams working at the same site with good knowledge of CDB data management. It implements workspaces in CDBs with private checkouts, i.e. only the owner of a checkout has read and edit access to it.

Because this use model is desired by many of our users, it may become the "default" use model in the new flow. Thus, we will have to make it very easy to use even by designers with little experience. DMP shall be re-implemented with "full TDM" providing individual workareas for all team members.

4.6.4 View specific Access Restrictions (DMX)

As a functional extension to DMS, view specific lists of user names can be maintained to restrict access to cellviews. After the initial release of this feature, we learned that some of the DFII tools did not regard the access restrictions set via user triggers. This underlines the need for a clean, uniform DDM layer below all of the DFII's tools.

It is not yet clear how view specific access permissions will be handled with TDM.

4.7 Design Migration

Because ASCIA 4.1 and our new fullcustom flow contain tools from various vendors, migration from the current to the future environment is not just a matter of CDB conversion. Complete designs will have to be transferred from ASCIA 4.1's DDM to TDM. Cadence's "libcvt" is a kernel for migration of ASCIA 4.1 projects to fullcustom flow projects. The versioning level of ASCIA 4.1 is mapped to "full TDM" releases. Because our CDB reference libraries' CDF parameters are changed, too, we will need a design transfer center which migrates our users's designs. A push-button solution is not feasible.

4.8 Performance[1]

Performance and stability of TDM has been improved dramatically from 4.4 to 4.4.1. We have measured the elapsed times for execution of the most frequently used TDM commands on an UltraSPARC with 128 MB main memory. At the time the tests were done, only one user was active in the design and on the machine. All data was stored on the local host so that no data had to be transferred via the network. For the last measurement, also the index files used by TDM were cleared. We do not yet understand the results of that measurement.

TDM Command	Number of Files			
	11,400	5,700	1,455	104
tdmcheckin -i	44 min	22 min	5 min	16 s
tdmcheckin	49 min	22 min	6 min	17 s
tdmcheckout	49 min	23 min	5 min	16 s
tdmsubmit	50 min	16 min	4 min	14 s
tdmupdate (no changes)	6 min	2 min	1 min	7 s
tdmupdate (all cells changed)	44 min	13 min	2 min	12 s

Table 1: Duration of some TDM commands executed on local host

4.9 Backup

Until today, Cadence did not supply procedures for on-line backup of CDB data with the standard software distribution. The DFII's design manager offers no CDB database administration utilities. "cdsAdmin" is a pain because the DFII tool supporter needs UNIX "root" privilege (which he has not!) to remove stranded locks. In the CDB 5.X/TDM world, things will hopefully get better. But again, we learned that Cadence does not yet provide procedures for backup, consistency check and repair of TDM managed data. Looks like a good business opportunity for Cadence consulting.

1. These measurements were performed by Joachim Glas from the CAD System Team.

5 Outlook

We expect that ASCIA 4.1 and SunOS 4.1.3 will be productively used by our customers and will thus have to be maintained for at least three more years. This also means we will have to maintain a heterogeneous environment with respect to both the operating system and the DFII versions used.

Right now (8/97), the alpha version of our new full custom design flow which uses DFII 4.4 design tools and “full TDM” is available for pilot use. Hopefully, it will be possible to present some of our pilot users' experience at the CUG conference 1997.

Today, the DFII 4.4.1 tools can only use TDM to perform DDM related tasks. Cadence's plan to provide a general data management (GDM) layer is one further step towards a modular, open system. We do not know the strategy behind GDM: Is it meant as an open interface to third party data management tools to get rid of the unsolved DDM questions? Or is it a signal that Cadence has understood that advanced DDM-related functionality will in the near future become at least as important as good design tools? I fear the former comes close to truth because there are still too many people in our industry which think a bunch of tools and libraries makes up a design flow. In that case, who will take care of the challenges associated with terms like “mix and match of IP blocks”, “virtual company”, “globalisation of the development process” and “teleworking”? Will it be an EDA company? Will the EDA tool vendors be able to adapt emerging standards from the office/consumer market like Multimedia, WWW and Virtual Reality to tackle the DDM problem domain?

6 Conclusion

The effort to make our IC design system ASCIA 4.1 with DFII 4.3.4 available for SunOS and Solaris 2 with almost identical tool functionality is much higher than expected. We have been facing many problems with DFII 4.3.4 in a heterogeneous SunOS/Solaris 2 environment. Most of them, but not all, have been fixed - one by one. We have spent man-years testing the various hotfix versions we received. Hopefully, we will be able to release the Solaris 2 version of ASCIA in the next 2 months. We would have saved much effort by waiting for a mature Solaris 2 version of DFII 4.3.4 and the availability of practical know-how about how to set up this kind of environment at Cadence.

Our next goal is to build a new, DFII 4.4.X based fullcustom design environment as a successor of the corresponding ASCIA subflow. The first pilot users have just started their work in an “alpha” environment. For DDM, it offers only minor add-ons to the basic TDM functionality. The first impression we share with our pilot users is that the structure of a project environment and the mechanisms to manage configuration files and design data has become quite a bit more complicated compared with ASCIA. We will have to spend several man-years of effort to implement the most urgent needs of our customers.

One year ago, Cadence intensively tried to push us into DFII 4.4. Today, being massively involved in the development work, we rather get the advice not to hurry too much but to prepare all aspects of the transition very carefully.

We will try do get as much benefit out of DFII 4.4.X’s new features like TDM with a minimum of customisation. However, the effort for migrating an existing DFII 4.3.4 design environment (CDBs, tech files, CDF parameters, SKILL code, etc.) to DFII 4.4.1 is considerable. Together with Cadence consultants, we have spent man-years to understand TDM and start to implement a new infrastructure with simi-

lar DDM functionality we already have in ASCIA. We would have appreciated to get direct access to Cadence people with practical experience in setting up a customised TDM environment.

In the end, our customers are asking us: “What do we have to pay you (the CAD people and/or Cadence consultants) to be able to use the most recent Cadence tools in the flows we need?” We have to tell them “Well, the problem is not to make the tools available. But with an effort of xx man-years, we have to build a new environment before you can use them and we/you will have to migrate all your design data if you want to re-use it.” And the reaction may be “Why should I pay so much money for those tool enhancements?”

Acknowledgements

The author wishes to thank Walter Eckardt, Joachim Glas, Gunther Hadwich, Marek Rouchal, Spiros Vaboulas and Holger Wende of Siemens Semiconductor’s CAD System Team for doing most of the development work and providing descriptions and solutions for the mentioned problems - together with several consultants from our local Cadence office in Munich.

References

/1/ Harriehausen, T.: “ASCIA - A Multi-Vendor IC Design System”
Proc. 1996 Int. Cadence User Group Conf., Sept. 8-12, 1996, Phoenix, AZ, pp. 207 - 215

/2/ Lazar, C. J.; Taylor, J. M.: “Integrating the SunOS 5.5 based Ultrasparc into a SunOS 4.X Design Environment”
Proc. 1996 Int. Cadence User Group Conf., Sept. 8-12, 1996, Phoenix, AZ

/3/ Powell, P.; Mason, J.: “LPRng-An Enhanced Printer Spooler System”
Proc. Ninth System Administration Conference (LISA ‘95), Sept. 18-22, 1995, Monterey, CA

/4/ http://www.pureatria.com

About the Author

Born in 1958 in the north of Germany, Thomas Harriehausen finished his studies of Electrical Engineering and Computer Science at the University of Hannover with the diploma degree in 1987.
His primary interests were in microelectronics, microprocessors, telecommunication, operating systems and software engineering.

His diploma work was about new algorithms for automatic layout generation for bipolar ICs.
For his Ph.D. thesis, he developed a system for adaptive, real-time simulation of thermal networks.

Since 1992, he is with Siemens Semiconductors (“HL”) in Munich where he started as a support engineer in the central IC CAD department. He is one of the architects of HL's “all-in-one” IC development system ASCIA 4.1.

He became leader of the “CAD System Integration” group in 1995. Since 1996, he is responsible for CAD “Systems and Training”.