E-Design Based on the Reuse Paradigm

L.Ghanmi [1], A.Ghrab [1], M.Hamdoun [1], B.Missaoui [1], G.Saucier [2], K.Skiba [1]

[1] CSI / INPG, Grenoble, France
[2] Design & Reuse, Grenoble, France

Abstract

This paper gives an overview on a Virtual electronic component or IP (Intellectual Property) exchange infrastructure whose main components are a XML "well structured IP e-catalog Builder ™" and a" XML IP profiler™ While the first module is a e_publishing and an exchange management module the second has as role to extract from the design directories the IP files and to trigger their transfer to the user site possibly via an IP distribution server under the catalog control. Direct Design file extraction from commercial configuration systems such as CVS and Clearcase is supported; notice also that the architecture supports if required a network of IP distribution servers preventing from a performance bottleneck when exchanging IPs; both modules have been implemented respectively in Java Servlet and as a Java client/server application.

1. Introduction

Due to design engineering resources shortage together with the integration power of sub micron technology, the necessity of design reuse practices is presently worldwide accepted. Reuse of previously designed blocks called Virtual Electronic component or IP (for intellectual property) is the key facet of collaborative design as it means that design blocks previously or concurrently implemented are provided by separate teams in order to achieve a final design in an acceptable time frame .
Many efforts have been made for establishing world wide accepted rules for high quality reusable block design (VSIA Standardization [10] and RMM guidelines [9]).
Many publications have been made the past years on this hot topic, concerning mainly the fundamental reuse methodology, the reuse management view and the e-commerce aspect supporting the IP exchange.

This paper focuses on the implementation of intranet and internet technologies allowing collaborative design in an enterprise as well as reusable block or IP exchange via internet, the paper will present the most innovative features on the 2 following aspects :
- Flexible IP cataloguing for both internet and intranet usage.
- Optional IP import from decentralized design centers to IP distribution servers in an intranet environment.
- IP file Download to the user site either directly from design centers or via IP distribution servers following user request to the catalog

All the methodology and tools presented here have been developed by using extensively XML tagging [20, 21, 22] as this is the well adopted standard for electronic information exchange and gives amazing functional capabilities for managing e design based on IP reuse .

The paper is organized into 2 sections one on the underlying catalog technology which handles also the Download requests, the second section handles the actual IP file or IP version exchange

2. IP cataloging technology

A catalog considered here is defined by the catalog infrastructure, the contents which a set of IPs, the functions that can be performed on the catalog items (searching, support and info broadcasting around the catalog for instance) as well as the online catalog modification process.

2.1 IP Catalog infrastructure and well structured IP catalog

Definition 1

A catalog infrastructure is defined by a pair (T,F) where :
- T is a so called taxonomy or IP classification;
F is a set or binary of library formats, each format being defined by a XML DTD (Document Type Definition).
Note that the use of DTD instead of XML Schema is not jeopardized by its rigidity as the format is used here as a description standard for reuse and the capabilities expressed by DTD are satisfactory

Definition 2

A well structured IP catalog is defined by a catalog infrastructure as defined above, a set of IPs and two applications *F* and *MD* associating respectively to each IP_i :

- A format element of F : $F(IP_i)$;
- A set of XML meta data : $MD(IP_i)$;

The wording "well structured" is related to the use of XML DTD providing a good support of the functions described later on such as searching, life cycle catalog modification capabilities, format bridging etc…

2.2 Classification or taxonomy associated with a catalog infrastructure

Definition 3

A taxonomy "T" is defined by :

- A rooted tree or a DAG (Directed Acyclic Graph)
- A mapping associating to each node N_i of the DAG a subset S_i of IP with the following rules :

At the top or root is associated the whole set of IPs;
The subset S_i associated to a node is partitioned into disjoint subsets associated to its successor nodes.
The partitioning process implies that an IP will belong to only one successor node. These rules make the classification process somewhat rigid as an IP cannot be attached to several sub categories and this may be a drawback in practice.
Several types of classification exist. The most commonly used criteria are the application fields and the function performed by the IP.

2.3 IP Formats and format library

Definiton 4

An IP format "f" is defined as a XML DTD defining and structuring the set of so called catalog meta data associated with an IP.
Practically these meta data correspond to information about IP that should help a user to understand the functions performed by an IP, its performances and quality allowing for instance to sort a set of IPs to find our the relevant one.

Data typing in an IP format

A DTD is by definition a hierarchical declaration of XML elements, in the IP format description used here the leaf elements are text type elements and are considered as different level of titles.

Three types are supported for the leaves :
Text types , Attributes, URL addresses

- Text types may correspond to single text fields or extensible text areas.

- Attributes may be the following :

◆ Multi valued attributes for which the set of values are predefined and enumerated during the declaration, a simple case being Boolean attributes taking 2 values. IP hardness for instance is an attribute that can take 3 values namely "hard", "soft" or "firm".

◆ Multi valued attribute with non numerable set of values.
For instance an attribute pointing to a target technology has by definition a non numerable set of values as new technologies may appear permanently, in this case the attribute values are expressed by character strings.

◆ Parametric value attributes, in this case the values are any real number and are associated with a large variety of units such as nanoseconds for timing, megabits per second for throughput, watts for power dissipation etc; comprehensive processing of these data types has to be available as these attributes are targeted during the search process, comparison and range containment checking has to be available.

- URL addresses allow to establish a link to data pre stored on the server hosting the catalog at a specific URL address; the data may commonly be data sheet, product sheet, any kind of documentation as well as design files; the URL address may be a directory address that can be browsed during a catalog "View" action or a file address possibly the zipped file of a directory to be downloaded by a "Download" action attached to the catalog.
In the environment described here both a directory address and a downloadable file may be declared for URL leaf type.

Format library and heterogeneous catalogs

Definition 5

A heterogeneous catalog is a catalog whose set of formats "F" contains more than one element.

F is then also called a library of formats underlying the catalog

A format library associated with a catalog is very convenient.
First it may be useful to look at an IP through specific views. For presales or marketing purposes it is important to give some relevant data but no too much of details; we may speak about a marketing format. An example of such a format is implemented in D&R web site and triggers a consistent number of client/provider matching.
Another reason for having a format library is to afford flexibility in associating a format to an IP, it may be understood easily that the same format describing a multi processor and a PLL cannot seriously be considered; thus there may be different formats associated with IP categories.

2.4 Catalog XML meta data

Catalog data in this approach means any kind of descriptive data assisting in understanding the functions and selecting the IPs with respect to an application and version independent.
It is easy to understand that an IP is delivered through its normal design life cycle and successive versions will be made available. A catalog contains "long term" data; a link to another specific module called IP profiler which is in fact an "IP version profiler" (see below) will import in the catalog version specific data. It is considered as a link or an import primitive and will be discussed later on in this paper (section 3). Are commonly available as catalog data the following categories of data :

-Identification data supporting a first level of searching.
- Functional description (key features).
- Claims which are targeted to demonstrate performances of the IP or its quality.
- Link to data sheet and product sheet and all levels of documentation.
- Version history or summary.
- Delivery data description.

All this information is translated as values included in the XML DTD. Thus to each IP is associated as said in the introduction a XML file called the meta data of the IP; the whole catalog metadata is the union of the XML files of all the IPs.

2.5 Functions supported by a well structured XML catalog

A large variety of functions is supported in a well structured XML catalog; for paper length reduction purpose we will only comment 3 aspects considered as significant; other aspects such as data uploading, catalog viewing, file downloading or info broadcasting , user support around a well structured catalog will be discussed in companion publications;

The 2 functions commented here are:
- Infrastructure and well structured catalog creation with extension to the creation of decentralized catalogs.
- Life cycle catalog modification.

2.5.1 Well structured Catalog creation

A well structured catalog creation environment has been developed in cooperation with D&R

Infrastructure creation

The following primitives have been implemented in a so called IP catalog Builder™ written in Java Servlet;
- Hierarchical taxonomy creation with "cut and paste" capabilities to provide more flexibility in this creation process.
- Format declaration through a XML DTD on line construction supporting the leaf data types defined in section 1.3.

In IP Catalog Builder™ a sophisticated environment has been provided; it supports a working space environment where the catalog manager can store a format so called "private format" for working on it during working sessions and before releasing it definitively and entering IPs.

Well structured catalog creation

This is done through a simple command associating to a node of the taxonomy a format; the format associated at the top node can be considered as a "by default format", a node inherits the format of its closest predecessor to which a format has been associated.

2.5.2 Life cycle catalog structure modification

This is an important point, once a catalog has been created and IPs descriptions have been entered it is important to have the ability to modify some catalog features without having the burden to reenter the data; by catalog modification we mean catalog structure modification mainly format modification and taxonomy. When modifying a format the new format becomes replaces the previous one.

Definition 6

A catalog modification is a permissible modification if no XML meta data have been altered or destroyed for already stored IPs and no data type violation has appeared.
A catalog modification is semi permissible if and only if a single identified XML element value has been altered.

Property 1

Permissible and semi permissible catalog modification actions respect the hierarchy relation of the initial format DTD and the data typing of the leaves.
As an example the following permissible catalog modifications have been implemented in IP Catalog Builder™
- Flattening taxonomy levels.
- Flattening 2 levels of titles in a format.
- Changing a title or a subtitle.
- Adding a new XML "child" element in the IP format DTD; this file becomes empty and open for data uploading for the already stored IPs.
- Permuting the ordering of 2 XML "child" elements of the same parent element.

A semi permissible that has been implemented in Catalog builder™ is the modification of the type of a leaf.

When doing that, the meta data of the already stored IPs are destroyed as there is a data type violation; the corresponding fields are empty and open for upload for all IPs of the catalog .

3. Importing and Downloading IP version in intranet environment

IP based e–Design is an important facet of collaborative design which has been widely discussed in the literature the last years [12, 13, 14]. This paper will focus on the features that are strongly linked to the IP reuse features

3.1 Problem statement and general IP Exchange architecture

In this section we consider an intranet environment dedicated to manage the IP reuse focusing on the file exchange for blocks designed in distributed design centers.
The overall architecture is depicted in *figure1.a:*

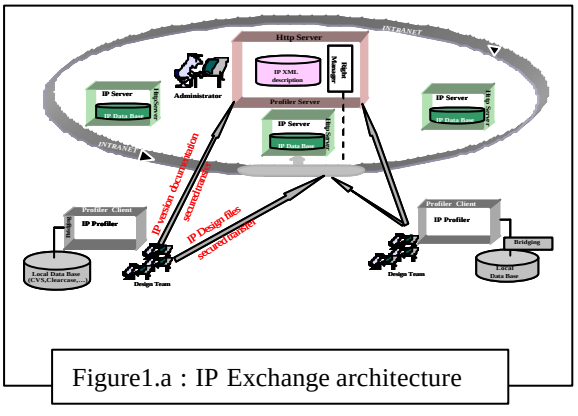

Figure1.a : IP Exchange architecture

The design centers represented at the bottom work in their design environment using the conventional design tools such as Synopsys, Mentor, Cadence .etc...environment at different levels of abstraction or for different design or qualification steps of the design.
They use their local data management styles by using well established configuration management tools such as CVS or Clearcase[15].
At a given point a Reuse manager may decide that an IP release is strategic and should appear in the intranet IP catalog which is a well structured XML catalog as described in the previous section.
As far as the IP design file storage is concerned 2 options are available : either the design files stay stored under the configuration server (CVS server for instance) at the designer site or the design files are compressed and transferred to one or several IP distribution servers.
It is expected that a designer selecting this IP in the IP catalog can trigger the download of this IP in his design environment in a convenient way

This section addresses the basic mechanisms to allow this download under the catalog control.

Should the files have to be stored on IP distribution servers a Java client/server application included in the Profiler will trigger this transfer.

Note that several distribution servers are depicted in *figur1.b*; it is extremely important that this exchange environment supports a network of distribution servers to prevent server congestion. Remember that to an IP is associated up to a gigabit of design files and a single server would not be realistic at all.

In both options when a user request is made to the catalog the client/ server mechanism triggers the transfer of the selected IP files either from the configuration server or from the IP distribution server to the user station, all these transfers are secured through proprietary mechanisms

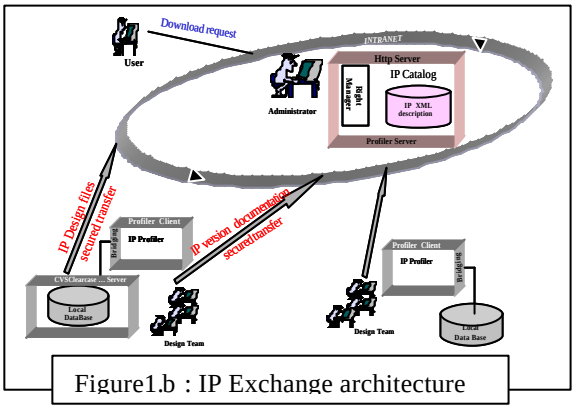

Figure1.b : IP Exchange architecture

3.2 IP Profiling

3.2.1 Profile Definition

Definition 11
An IP profile is a XML DTD describing the delivery directory associated to an IP version allowing its reuse; its leaves are linked by path declaration to the delivery files of the IP.

File packages
While the IP profile is somewhat similar to a catalog format description differences are significant especially through stronger typing capabilities in the IP profile.

First the delivered files may be grouped into so called *packages* which represent the lowest granularity in terms of subset of files that can be exchanged or downloaded individually; as an example a simulation package comprises the

simulation model itself as well as the simulation library, the test stimuli and the test report.

A second typing concerns the definition of package types; namely 3 types have been defined :

- *Documentation packages* containing information data.
- *Design packages :*

Whose leaves are linked to the models or views of the design at different level of abstraction (system view, RTL view, netlist view, layout view). The design views at different levels are linked by a top down dependency relation as between 2 successive levels of abstraction a design action has taken place and is expressed within the DTD associated to the profile

- *Qualification packages :*

Attached to a design level; as indicated the purpose of this package is to reports on a qualification action and its leaves are linked to simulation, testing, verification. Files or reports to a given level of abstraction several qualification packages may be attached

- *Claim package :*

It may be convenient to distinguish special documentation packages containing strategic types of "result information" expressed in terms of parametric attributes values; they play an important role when searching or comparing IPs

A profile so called a "soft IP profile" from D&R is shown for illustration (figure 2).

The data types accepted at the leaves include the data types of the catalogs, text type, multi valued attributes and the URL link.

But a new type appears especially in the design package which is a so called "bridging path" commented later on and which is fact an element of the mapping of the native design directory to the XML delivery directory described in the profile.

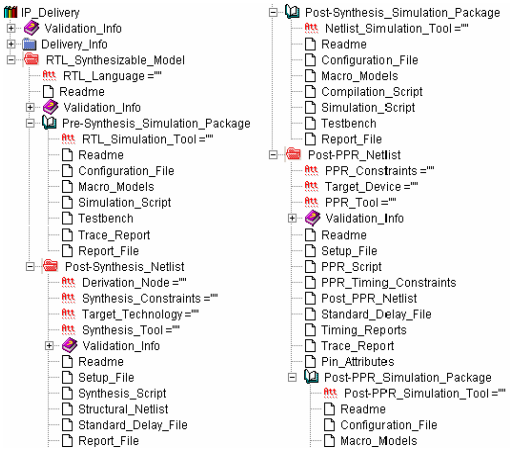

Figure2 : Soft IP Profile

3.2.2 Package parameterization

To a package is associated a set of XML attributes called package parameters; they aim at characterizing the specific package include in the described IP version . These parameters specify for instance the architecture or configuration options of the IP version, the environment conditions under which the files have been generated. Commonly it includes elements such as the design tool set, the various scripts, the libraries, the target technology.

The IP manager in charge of the IP profiling is free to select these attributes.

It means also that the same package may be re-instantiated with the same version with another set of parameter values, as an example a netlist model commonly exist in different formats (VHDL, Verilog, EDIF); thus this netlist model will be declared with the 3 values of the format attribute; similarly various simulation reports may be available using different simulation tools (Model Sim or VSS); thus different simulation packages may be attached to a design level.

Extended Package definition

Following the previous remarks the following definition can be stated:

A *package* in the Profiler is defined by its XML label, its type, a set of parameterization attributes and the set of delivery files belonging to this package(documentation and Design files)

The syntax of a profile has been explicitly defined and is not reproduced here for space limitation

3.2.3 Profile selection

As shown above the definition of a profile is made fully customizable and should follow the corporation Reuse policy

The profiling technique can be used in different perspectives and several strategies (bottom up or top down) can be used in a corporation

- It can be used as a documentation info when exchanging an IP; this means that when proposing an IP for reuse or exchange, the files delivered by the designer should by properly described by a XML profile so that the receiver interested by such a IP can clearly understand the delivered files by looking and browsing in the catalog; for a specific IP the profile could be freely chosen by the designer and entered manually as a DTD.
- A profile can represent the directory structure that a company wishes to impose as a standard to facilitate IP Exchange and the directory of the existing IPs will be "mapped" or "bridged" to the standard one (see below).

3.3 Publishing an IP version

3.3.1 Publishing a specific version

In the profile consists in filling in:

- Characterization attributes associated with the packages up to the final claim package
- Documentation packages which are commonly linked to documentation files (pdf, ps, html, …)

3.3.2 version summary

As said above an IP version is described by the set of packages at different levels of abstraction and ends up with a claim package stating the final result.

Thus a version summary table gives the list of packages associated with the successive versions labeled by their parameter values.
Note that versions are incrementally uploaded and downloaded.
Figure3 gives a version summary for 2 successive versions targeting 2 distinct technologies namely one on a Xilinx technology and one of an Altera technology

Figure3 :version Summary

3.3.3 Importing the published data in the catalog

The transfer of the information data is triggered form the catalog by a so called “Import IP command”; this triggers the transfer of the documentation files as well as information contained in the version summary table.
Once this is done when selecting an IP, the user can see not only catalog data as described in section 2 but all the published data relative to a specific version and may choose to download it.

3.4 Importing IP Design files version or Bridging mechanism

Importing IP design files or an IP version means assigning a URL value to a Delivery package or a directory or to the whole IP

This bridging mechanism can be
- Manually achieved or by an on line process each URL address is entered by the designer.
Interactively by an assisted mapping, the assistance is providing by displaying on the screen the native directory and the design can designate for each native leaf the leaf in the XML profile
- Associated with a search process in the configuration server
Figure 4 gives the interface that will show up if a package or a subset or the whole IP has already been configured under CVS or Clearcase for instance.

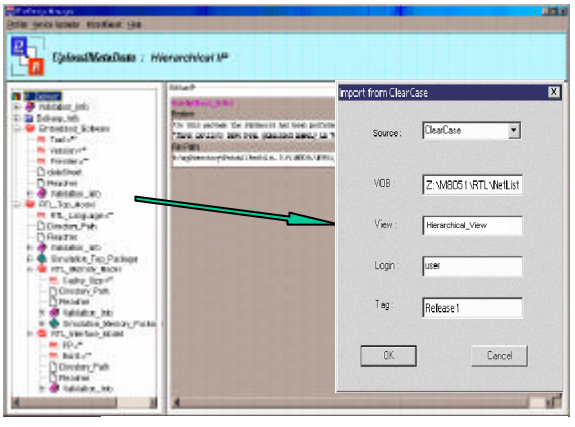

Figure 4 : Clearcase bridging

3.5 Exporting the files to IP distribution severs

We assume in this section that the exchange architecture is based on the existence of several IP distribution servers. The profiler which is a Java client/server application whose application part runs on the designer site and the server part on the central catalog management server as well as on all IP distribution servers.
To import an IP to a distribution server a command :
- Generates the compressed file for the IPs.
- Takes as parameter the identification of the IP distribution sever.
- Triggers this secured transfer; in this case a trusted environment is created based on a Virtual private network (VPN)

3.6 Downloading to the user site

The Download to the users site is triggered after an user has selected the IP version he is interested in.

The publishing data or information is directly downloaded from the catalog server while come the design data may either from an IP distribution server or from the native configuration server at the designer place

4. Conclusion

A very important aspect of collaborative design is the sharing of design blocks whatever they come from an internal design site in an intranet environment or externally acquired blocks; in both cases the net based exchange plays a fundamental role. This paper has addressed 2 major aspects :

The first one is e-publishing or information exchange; it is obvious that any exchange is first dependent of easy accessible, accurate and comprehensive information; thus an XML internet/intranet IP cataloging methodology and related Java based technology which benefits from 4 years of experience and improvement is presented; this catalog is resident on a "master" server and will trigger and control actual design files exchange described in the second part
The second part addressed the key issue namely the IP design file transfer in an intranet environment from design sites to a user either directly from the configuration system of the designer or through one or several IP distribution servers; thus the geographical distribution is supported both for the design centers and the servers storing the IPs thus refraining from traffic congestion.
The exchange methodology and architecture presented in this paper comprise all the ingredients of a successful e -Design environment and various
applications such as project management; IP entry XML encapsulation portal for a large corporation have successfully demonstrated the robustness of the technology.

References

[1] B.Behman, B.Babba, G.Saucier
« IP Taxonomy, IP Searching in Catalog»
Design, Automation and Test in Europe (DATE), February 1998, France
[2] Ph. Delforge
«IP Business models, IP reuse ; IP Market and IP corporate database»
IP'98 Europe Proc., pp 11 – 16, Mar. 1998
[3] R.Seepold
«Reuse of IP and Virtual Components »
Design Automation and Test in Europe (DATE), Munich, March 1999
[4] I.R. Pearson
« The factors determining the value of Intellectual Property (IP) being developed for use in the semi conductor industry»
IP'98 Proc., pp19-26, Mar. 1998
[5] P. Clarke
« IP-core catalog adds Net access, e-commerce»
Electronic Engineering times,p.88, April 6, 1998
[6] R.Rafidinitrimo, P.Coeurdevey, G.Saucier
« IP Key Features and IP Catalog Management System»
Design, Automation and Test in Europe (DATE), March 9-12, 1999, Munich, Germany
[7] R.Rafidinitrimo, M.Hamdoun, B.Missaoui, SA.Senouci
« Rights Management for IP Exchange»
International Workshop on IP-Based Synthesis and System Design, December 1999, Grenoble, France
[8] J. Kim, Y.lee, K.kown, C.Lee, KETI
« IP Database and Catalog System »
International Workshop on IP-Based Synthesis and System Design, December 1999, Grenoble, France
[9] Michael Keating (Synopsys) and Pierre Bricaud (Mentor Graphic)
«Reuse Methodology Manual for System on Chip Design » November1998
[10] Virtual Socket Interface Alliance
www.vsi.org
[11] A.Reutter, B.Moβner, W.Rosenstiel
« Design of reusable modules for reusable designs »
Workshop on IP Based Synthesis and System Design. 15./16.12.98
[12] Katz, R., H., and Lehman T. J.,
<<Database Support for version and Alternatives of Large Design Files>> IEEE, Tans. On software Engineering , Vol. SE-10,No.2, March 1984, pp. 191-200.
[13] Katz, R., H., and Chan E.,
<<Managing Changes in aComputer-Aided Database>> Proc. Int. Conf. On Very Large Databases, Brighton, 1987, pp. 445-446.
[14] Katz, R., H., and Mario J. S,
<<Version of Schema for design using the word wide web>> 32ndACM/IEEE Design Automatic Conference , 1995.
[15] D. Leblang. The CM Challenge : Configuration Management thats work, chapter 1, pages 1-37. In Tichy Tic94, 1994.
[16] C.Lytle, R.Beachler
« Using Intellectual Property in Programmable Logic»
ED&TC (Electronic Design & Test Conference), User Forum, p.125-129, Mar.1997
[17] M. Koegst, P. Conradi, D. Garte and M. Wahl
« A Systematic Analysis of Reuse Strategies for Design of electronic Circuits»
Design Automation and Test in Europe (DATE), February 23-26, 1998, Paris, France
[18] M. Dalpasso, A. Bogliolo, L. Benini
« Specification and Validation of distributed IP-Based Designs with javaCAD»
Design Automation and Test in Europe (DATE), Mars 9-12, 1999, Munich, Germany
[19] J. Haase
« Design Methodology for IP Providers»
Design Automation and Test in Europe (DATE), Mars 9-12, 1999, Munich, Germany
[20] W3C Technical Report and Publication
[21] XML Base www.w3c.org/TR/xmlbase
[22] Canonical XML www.w3c.org/xml/
[23] A.michard
"XML Language et application"