An Introduction to OpenAccess

An Open Source Data Model and API for IC Design

Michaela Guiney

Co-Chief Architect
OpenAccess Change Team
Cadence Design Systems
San Jose, CA 95134
Tel : 408-428-5923
Fax : 408-428-5380
e-mail:mguiney@cadence.com

Eric Leavitt

IC Framework Technology
Cadence Design Systems
San Jose, CA 95134
Tel : 408-894-2247
Fax : 408-428-5380
e-mail:eric@cadence.com

Abstract - The OpenAccess database provides a comprehensive open standard data model and robust implementation for IC design flows. This paper describes how it improves interoperability among applications in an EDA flow. It details how OA benefits developers of both EDA tools and flows. Finally, it outlines how OA is being used in the industry, at semiconductor design companies, EDA tool vendors, and universities.

I. Introduction

The purpose of this paper is to provide an introduction to OpenAccess, and how it differs from previous standards efforts in EDA data models. The improvements in interoperability and efficiency for design flows are described. The capabilities provided by the API for application developers are also discussed. Finally, the paper lists some of the companies and universities that are using OpenAccess, and where to get additional information.

II. What is OpenAccess?

OpenAccess is an advanced EDA database designed to enable interoperability among IC design tools through an open standard data access interface (API) and Reference Implementation of that API. The OpenAccess data model spans the EDA design space. It can be used to represent designs from post-synthesis netlists to tapeout.

Standardization of EDA data models have been attempted in the past without significant success. OpenAccess took a unique approach. In addition to defining a standard API and data model, OpenAccess provides a Reference Implementation. The implementation was donated by Cadence Design Systems, who contributed their leading edge IC design database in 2002. Cadence continues to develop and maintain OpenAccess as the Integrator, and has invested over eighty person years in the project. The availability of a high-quality implementation eliminates a barrier to using OpenAccess in design flows.

Changes to the OpenAccess API are managed by the OpenAccess Coalition, a group of over 30 companies who are leaders in the EDA, electronics, and semiconductor business. Twelve members of the OAC are elected to the change team, which champions, refines, and approves changes to the API. Another key differentiator is that changes, once approved, must undergo user-level testing before they are integrated and distributed. This approach ensures the usability and quality of proposed changes. The OpenAccess effort is managed by Si2, the Silicon Integration Initiative.

The first release of OpenAccess was version 2.0, in January 2003. OA 2.1 followed in June 2003. In October 2004, OpenAccess 2.2 release was made available. OA 2.2 is the release targeted for production usage by most companies.

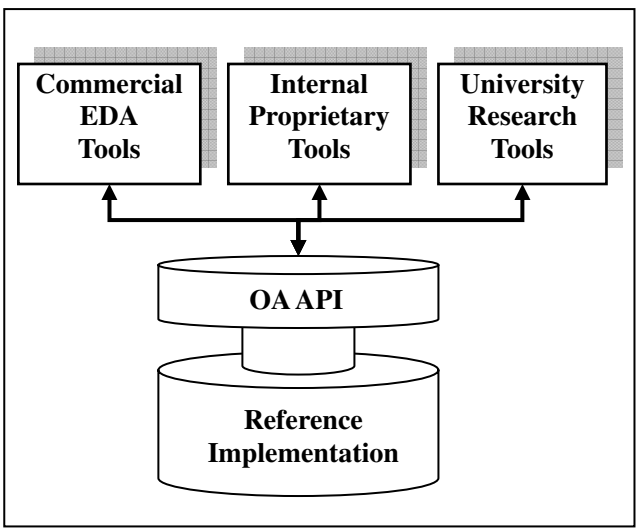

Fig. 1. The OpenAccess API and Reference Implementation : Available to All Tools

III. Why Use OpenAccess?

OpenAccess provides advantages to developers of both design flows and EDA tools. We will discuss both.

0-7803-9451-8/06/$20.00

A. Advantages of OpenAccess in Design Flows

Today, many design flows use common file formats such as Verilog, DEF, GDSII, and SPEF to exchange information between tools. There are two main issues with this approach. First, the various data files for a design are usually incomplete and inconsistent. Second, each application which needs design information must parse one or more of these files and translate it into an internal data structure. This translation is inefficient, and can often misinterpret or lose information, due to ambiguities in the format specification. OpenAccess can solve both of these issues. First, the OpenAccess information model is more complete, unambiguous and consistent than the collection of data formats typically used. In addition, an OpenAccess database for a design can be read by applications through the API much more efficiently than these file formats can be translated, leading to greater efficiency in the overall flow. OpenAccess provides much tighter integration than was previously possible for a design flow with tools from multiple sources. The most efficient approach is for applications to operate directly on the OpenAccess data model. However, some applications may have algorithms that need specialized data structures. In such cases, applications can use OpenAccess data to build the specialized data structures, calculate their results, and store them back into OpenAccess.

Although a well-specified data model reduces the need to modify data in order to ensure it is correctly interpreted by the next tool in a flow, sometimes customer-specific needs require manipulating data and building customized tools. Efficient access to design data via a C++ API enables more efficient flow customization than is possible when limited to accessing data through extension languages. OpenAccess users have seen dramatic reductions in runtime of applications ported to OpenAccess from extension languages and scripts. [1]

The ease of integration opens the door to new possibilities. University research developed on OpenAccess can be quickly integrated into a design environment. Exchange of design data and development tools with business partners can be achieved more easily. The effort required to evaluate new internal or commercial tools based on OpenAccess can be greatly reduced. Overall, OpenAccess can increase the ability to evolve flows to respond to new design challenges.

B. Advantages of OpenAccess for EDA tool Developers

OpenAccess is an object-oriented API written in C++. It was built from the start for open community use. Using C++ ensures a strongly typed interface, preventing many programming errors. Consistency was emphasized during the design of the API, in order to make it easier to understand and use. It also has extensive online documentation.

The OpenAccess information model covers a large portion of the EDA domain. It represents both logical and physical hierarchy and connectivity, as well as an occurrence model which relates the two. It includes custom geometry, routing topology and floor planning information, parameterized cells, scan chains, and technology information. It also provides a

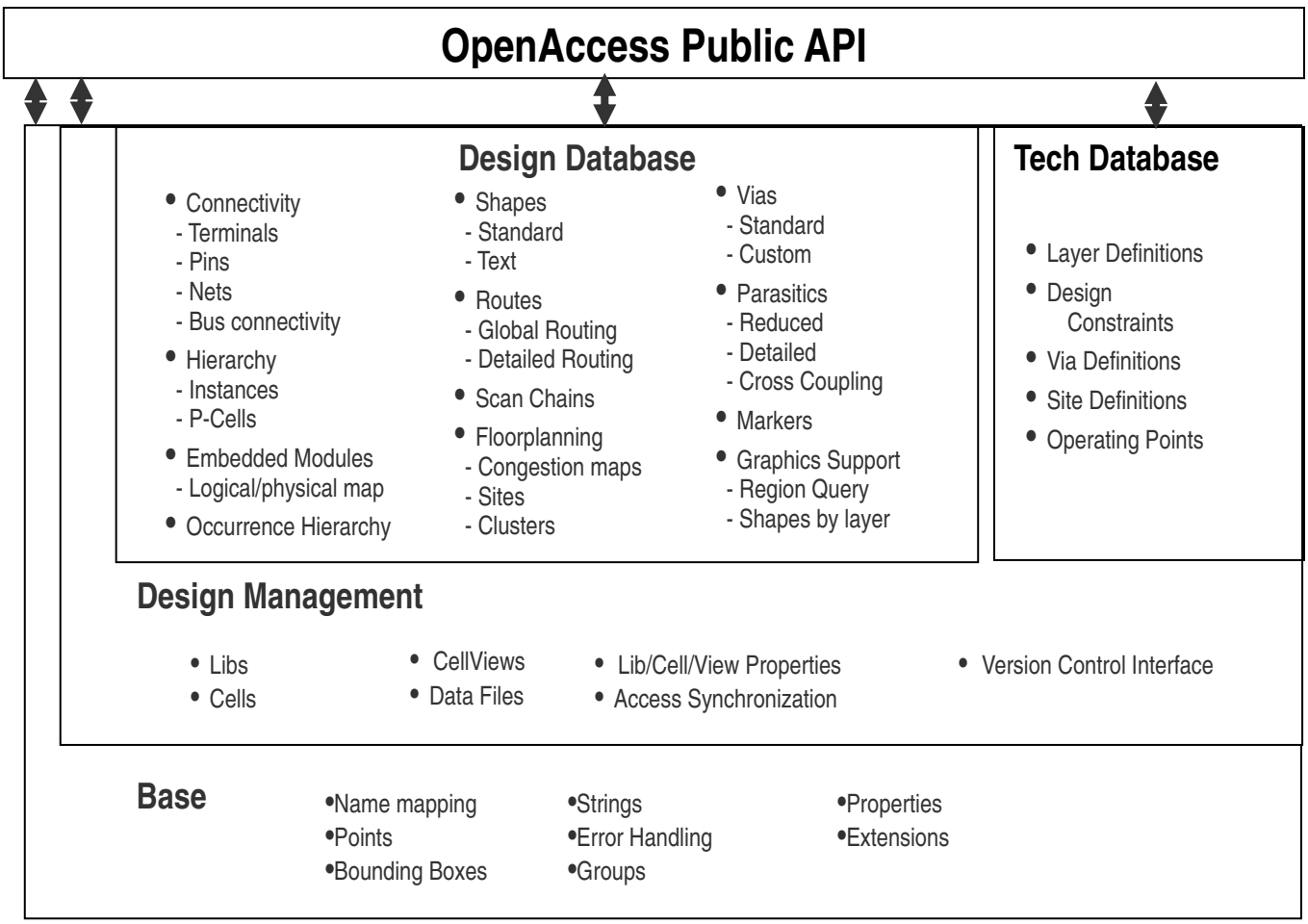

Fig. 2: Information modeled by OpenAccess

parasitic network API which enables access to parasitics on a per-net or even per-subnet basis, to minimize memory usage. An observer mechanism can be used by applications to take action when database objects are modified. Finally, the API supports defining extensions to most built-in objects as well as new kinds of objects; those extensions can optionally be saved. The extension mechanism is highly efficient, and can be used by developers to extend the database to support their application's needs. [2]

The API has been designed to support typical application access to EDA data. Various collections are supported, with filters for different types of usage. The API supports efficient searches, utilities such as Region Query, and name mapping capabilities. Multiple Design Management (DM) implementations are possible via a plug-in mechanism. The Reference Implementation has been tuned for improved performance and memory efficiency. Information and relationships are cached on the fly, depending on application access. Tcl, Python, and Skill extension languages are also available for use.

The breadth of the information model is one reason why many EDA tool developers have chosen OpenAccess as their development platform. The availability, quality and performance of the implementation provided by the Integrator (Cadence), and used in production by Cadence and other major EDA vendors, is another key factor. There are other factors as well. The availability of the source gives developers the best of both worlds. They can rely on the Integrator to provide maintenance and enhancements, but for time-critical bug fixes, they have the source available to them to fix issues themselves if required. Also, the quality and extent of the documentation surpasses that of any other EDA API available today, and certainly surpasses that of most internally deployed databases. Finally, the momentum behind OpenAccess ensures that their OA-based applications can interoperate efficiently and correctly with a growing number of tools.

IV. Where is OpenAccess Being Used Today?

Many companies are actively working with OpenAccess today. Freescale, IBM, AMD, HP, AMI, Renesas, and LSI Logic are just a few. LSI Logic is using OpenAccess in its RapidWorx design flow [3]. IBM is using some OA 2.2 based tools in their custom design flow [4]. Renesas is basing their integrated EDA system on OpenAccess [5] [6]. AMI Semiconductor is using OpenAccess as the basis for their flow to convert ASIC or FPGA designs to their environment [7]. Some companies are developing their own OpenAccess-based tools. Others are using tools from Cadence and other vendors based on OpenAccess 2.2. Cadence's custom design tools, digital implementation tools, extraction and physical verification tools all work with OpenAccess 2.2. In addition, a number of other EDA vendors have products that use OA 2.2, including MatrixOne, Mentor Graphics, Silicon Navigator, Cira Nova, Atrenta, and Gradient.

Universities such as UC Berkeley, Carnegie Mellon University, and the University of Michigan are using OA to build tools and toolkits to further research. They have put a set of tools together into a package called OA Gear. The package includes a viewer, a static timing engine, a placement interface, a functional/logic representation package and an and-inverter graph package. [8] In addition, North Carolina State University is independently utilizing OpenAccess to write tools for 3D thermal analysis and visualization. [9]

IV. Where To Get More Information

For more information on the OpenAccess Coalition, see www.si2.org/?page=69. Materials describing companies' experience with OpenAccess can be found through links to various conference and panel presentations at the Si2 "News" website, http://www.si2.org/?page=12. To download OpenAccess code and documentation, go to www.openeda.si2.org. For more information on the OA Gear project coordinated by Cadence Berkeley Labs, go to http://openedatools.si2.org/projects/oagear.

V. Summary and Conclusions

OpenAccess provides a comprehensive data model, API, and Reference Implementation for IC design. It is being adopted as the foundation by a number of EDA vendors, and is seeing more and more usage at IC design companies. The time is right for companies seeking efficient integration of tools and data in their design flows to investigate OpenAccess.

Acknowledgements

Thanks to the OpenAccess Development Team, the OpenAccess Coalition, the Change Team, and Si2.

References

[1] T. Heiter and J. Glas, "How Infineon Implemented OpenAccess", EETimes, November 11, 2004, http://www.eetimes.com/news/design/features/showArticle.jhtml;jsessionid=NHADQETGW5UI4QSNDBOCKHSCJUMEKJVN?articleID=52601068

[2] D. Mallis, *OpenAccess 2.2: Standard API Tutorial,* 2005. Available at www.si2.org

[3] R. Goering, "OpenAccess adoption challenging but worth it", *EETimes,* November 22, 2004, http://www.eedesign.com/article/showArticle.jhtml?articleId=53701169

[4] K. Barkley, "IBM: OpenAccess Adoption from End User's Perspective", *7th OpenAccess Conference*, November 2005, http://www.si2.org/?page=613

[5] Y. Inoue, "OpenAccess: Reducing Handling Mistakes and Smoothing Logic Design and Layout Design", *DesignWave Magazine*, August 2005, page 99.

[6] I. Kojima, "Design View from Japan: Renesas Leverages OpenAccess Database to Pursue EDA Integration", *NE Asia Online,* http://neasia.nikkeibp.com/archive_magazine/nea/200405/columns_305267.php

[7] K. Reynolds, "AMI: OpenAccess Adoption from End User's Perspective", *7th OpenAccess Conference*, November 2005, http://www.si2.org/?page=613.

[8] R. Davis, "OpenAccess Tools for 3D Integration", *7th OpenAccess Conference*, November 2005, http://www.si2.org/?page=613.

[9] A. Hurst et al, "OpenAccess Gear Functionality: A Platform for Functional Representation, Synthesis and Verification", *7th OpenAccess Conference*, November 2005, http://www.si2.org/?page=613.