Interoperability Beyond Design: Sharing Knowledge between Design and Manufacturing

D.R. Cottrell
Si2

T.J. Grebinski
SEMI Data Path Task Force Chair

Abstract

The nature of IC design has is necessarily evolving to a more data-centric design flow in which EDA tools share a common information in a design database without the negative cost and quality impacts of data translation from sequential files. In support of this new paradigm, a collection of mainstream companies within the IC supply chain have sponsored the development of an open industry data model and application program interface for IC design tools, along with a database that fully implements this. This technology, called OpenAccess, is now available and being adopted by the IC design community.

Another industry effort is in operation with the goal of greatly improving the cost and efficiency for IC photomasks. That effort is exploring a new paradigm similar in nature to OpenAccess in that a common data model and data access language is proposed. This data model would span both the design and mask-making communities, and possibly expand into wafer fabrication over time. Thus, it has become known as the Universal Data Model (UDM).

This paper discusses some of the rationale for the UDM and highlights the attributes of the OpenAccess technology that make it the ideal base on which to build an open industry UDM.

1. Introduction

Quality in the design of integrated circuits has improved dramatically over time. Improvement has come from new and improved design tools (e.g. synthesis, static timing analysis, faster simulators, etc.), improved design methods (e.g., RTL level design, scan design and ASIC, etc.). Also, improvement has come as the result of better communication of information between design processes and tools. The industry has shifted away from organizations whereby different teams have narrow a responsibility to the overall design to a broader approach whereby responsibility spans over a broad spectrum. For example, the considerations of manufacturing test within the logic design, and of physical placement and global routing within synthesis. This integration of what were considered discrete steps in the design has had a dramatic effect on design quality, and the progression necessarily continues as IC features shrink.

In support of this integration of design steps, design flows are necessarily becoming more data centric. The exchange of information between design steps by sequential files is giving way to central representations of information that is incrementally accessible by broad collections of design and analysis tools. This can be evidenced in products from a number of commercial EDA vendors and as proprietary solutions within a number of companies performing advanced IC design. Unfortunately, each of these solutions is limited in the scope of EDA tools it supports because of its technological and intellectual property limitations. These limitations preclude the use of these solutions across the broad spectrum of tools required for IC design and limit the choice of tools that can be used within a design flow using any particular solution. This is a major concern for companies pushing the limits in advanced IC design technologies. This will be a major concern for the commodity design marketplace when their design points match what is now considered advanced.

1.1. OpenAccess based design systems

OpenAccess is an opportunity to achieve data centric design systems but which afford the customer choice in the design tools that populate the flow. The OpenAccess technology is rich is breadth and depth and provides a wealth of extensibility features. Further, the OpenAccess technology, which consists of a datamodel for IC design information, an API (application program interface) definition, and complete source code for a production quality database supporting these, provides:

1. an open-technology available to any party under non-discriminatory terms — commercial or private, industrial or university, competitor or customer, and worldwide;

0-7695-1881-8/03 $17.00 © 2003 IEEE

2. the using party ample rights to develop and distribute unrestricted derivative works using the API and database; and,

3. technology that is being used by the tools of the company providing it, and under the same terms and conditions that it is offered under.

OpenAccess is providing a level playing field for participation in the solution of IC design information communication across the industry and where companies can still compete in the critical areas of new and improved tools and design methods. Further, the OpenAccess technology is provided in a manner that allows providers of EDA tools to provide complete support for their customers without dependence on other organizations. With the availability and use of this technology, the industry can provide even broader levels of integration across the growing spectrum of IC design and analysis tools. Now point tools from commercial EDA companies, internal development and university research can be integrated within the data centric flow providing customers with the choice they require to be competitive. Now, it is possible to integrate across all design and analysis steps in the design flow. But is this sufficient?

Design has most often been considered to include those steps in the process that begins with the personalization and interconnection of RTL (register-transfer level) elements and ends with the design rules checking and logical-physical verification of polygons that specify the fabricated IC. From there, a GDSII stream file would be cut and the process then handed off to the mask maker. Mask making has been considered to be a job-shop operation, where the specification of the polygons and their placement and the mask order (e.g. SEMI P10 standard) is sufficient. Thus, a clear separation of responsibility between design and mask making was possible, and with no reason to consider information centricity across them. This simple hand-off paradigm, whereby the GDSII and P10 were sent to the mask shop, was sufficient. But this is now becoming problematic [1].

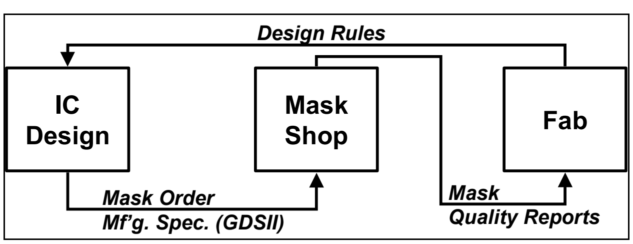


Figure 1: The Classic Hand-Off Paradigm

1.2. Mask making

Classically, masks were WYSIWYG, easy to make and inexpensive — just a necessary commodity in the overall IC supply chain. Today, the physics of light has made mask making a far more complex task and a critical enabler to wafer fabrication. This also is making a substantial impact on the amount of data required to express an integrated circuit design for manufacture. Very small and dense integrated circuit elements and features within a design today do not resolve well on either a reticle or wafer. Currently, the wavelength of the light used for exposures is 248 nm and moving to 193 nm. whereas the circuit elements may be at 130 nm or smaller. Hence, corrective shapes must be added to the original design to account for the optical distortions introduced by the relative size of the wavelength of the light used to expose features with the features themselves. Introduction of these resolution enhancement techniques (RET) accelerates the increase in data volume and thus accelerates the need for a substantially greater computational infrastructure capable of handling such quickly increasing quantities of design and manufacturing data. And, the quality of results, as measured by the increase in data, can be effected by knowledge of the both the intended design and the manufacturing processes used to fabricate it[2][3]. This increase in data size results in proportional increases in the transmission time for the handoff to the mask maker, the data preparation time (fracturing etc.) in the mask shop and the mask writing times, and complicates the overall data management problem.

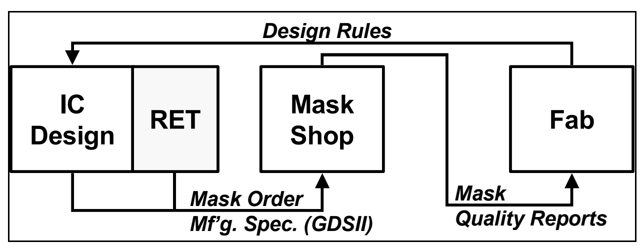


Figure 2: Today's Hand-Off Paradigm

This is becoming of critical concern because mask making cost is heavily weighted by data size (ref. figure 1) and becoming an impediment for low volume wafer starts.

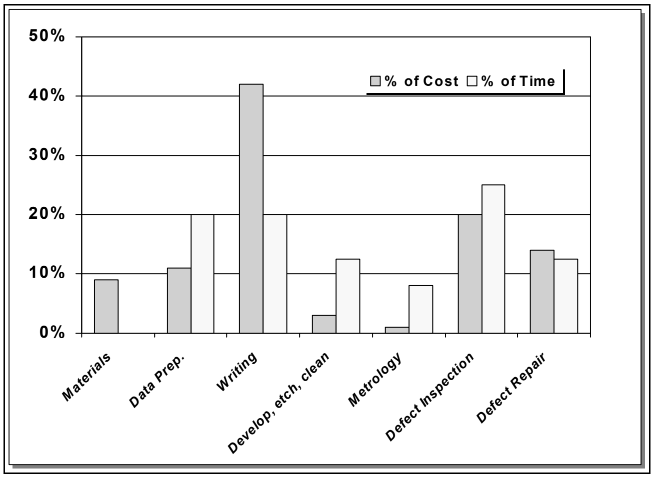


Figure 3: Mask Cost and Turnaround Factors[4]

Steps are being taken to address this situation. Two important steps are underway under the auspices of SEMI (Semiconductor Equipment Manufacturers, Inc.). The first is the development of a replacement to GDSII but that represents the same information in at least one-order of magnitude less space. This new format for design handoff to manufacturing is called OASIS (Open Artwork System Interchange Standard [Draft] Standard)[5]. A parallel effort is underway to develop an alternate paradigm whereby a common base of information about both the design and manufacturing can be shared across all disciplines in the IC supply chain. This effort has been given the name Universal Data Model[6] and is depicted in the following figure.

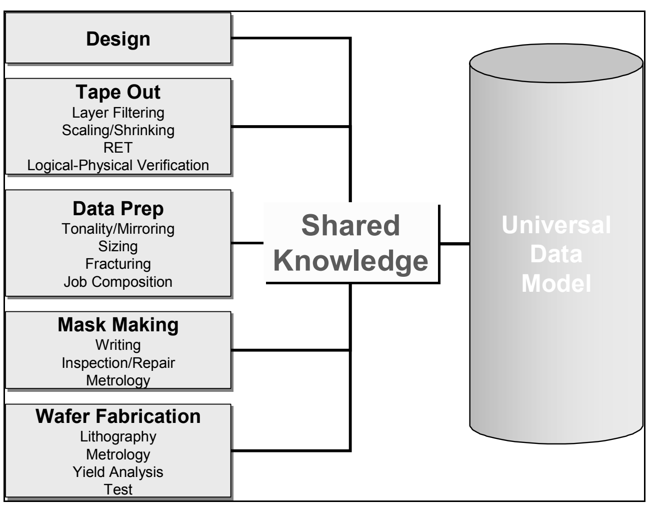


Figure 4: Universal Data Model Paradigm

This paper discusses the features and virtues of the OpenAccess technology that make it the ideal base on which to build this as an open industry standard usable by and between design and manufacturing.

2. OpenAccess Language Features

OpenAccess[7] differs from GDSII or OASIS stream in that it is a language based interface to information as opposed to a standard file format. The OpenAccess language is a set of C++ functions that operate on a hierarchical set of class definitions for that information. Collectively, this is referred to as the OpenAccess Application Program Interface (API). This API offers numerous technological and operational advantages, some of which are described in the following paragraphs.

2.1. API vs. format

Exchange of information by means of a [stream] file in which the data is stored in a standard format means that any application accessing that data must embody code written to that format. Thus, if the format changes in some way, so must the application code that operates on it. An API is language-based. It specifies how to request information but hides its stored format; thus it insulates applications from the data format allowing changes and improvements to be made to the storage methods without effecting the applications. With standard file format based interchange, change to the format comes with a cost impact to all applications using it. For example, OASIS will yield an order of magnitude or more decrease in the size of a stream file. This will result in a significant improvement in transmission and mask data preparation times. To reap this benefit, software and tools now reading GDSII must be recoded to parse and operate on the new OASIS file format. Should yet another creative improvement to that format be proposed, then again the applications would need to be recoded.

Through an API interface, applications are shielded from changes to the stored form of information, and data representation improvements can me made transparently to the applications. Only when new information is added to the data model is the application effected and then, only if the application needs to operate on that new information. This means that changes can be made to the data format to improve its compactness in a more continuous manner and on a competitive basis without impact on the applications..

2.2. Community management

To be successful, the UDM can not stagnate and remain unchanged. In order to meet the growing technology needs, it too must grow. However, in order to be a resource that serves the industry at large, its change to must be managed by the industry. No single company can control what changes are made to the UDM or how they are implemented, nor can they control when such changes are released.

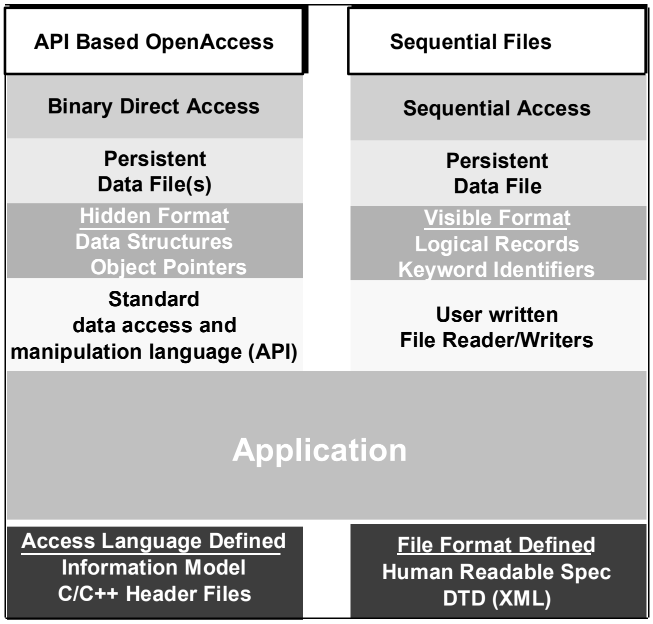


Figure 5: API vs. File Format Standards

OpenAccess provides a procedure for managed change to the technology under the control of an elected body of stakeholders from twelve companies (called the Change Team). The process dictates that the Change Team be a mix of companies from different parts of the IC supply chain in order to assure an objective view of all change proposals. The process used within this body is a democratic one that provides each participating company an equal vote.

Change to the OpenAccess technology may be proposed by anyone in the industry and, in fact, this is encouraged. All elements of the OpenAccess technology are provided to any person or enterprise worldwide, and on an equitable basis without bias. Further, they are provided in source format, which allows for anyone to make and test changes and recommend them for adoption by the industry. (In order to assure that there is only one OpenAccess API, any change to the OpenAccess technology that modifies the syntax, behavior or semantics of the API must be adopted by the Change Team before that change can be distributed externally.)

2.3. Extensibility

Often it will be desirous to add information to the UDM that is of a proprietary nature. There may be some new technology characteristics or constraints that give a company and its partners a competitive edge. In such cases, it is not possible that any community change process would be acceptable.

GDSII and OASIS both provide for the addition of user defined properties. Formally, these properties are attributes that are associated with specific objects in the information model. The format of these properties is user defined and most often they are sets of keyword/value pairs.

The OpenAccess API provides a rich capability for making extensions to the data model that are private to the creator any anyone he chooses to inform. OpenAccess provides for user defined properties on (virtually) any of the objects in its information model and further, provides a facility to collect sets of objects under a single named *group* and operate on that group as a single object. For example, the collection of all physical shapes on a specific layer may be collected into a *group*. Additionally, the OpenAccess API includes extensibility classes that permit applications to create high-performance, application defined database extensions that work the same as the built-in attributes of the database. Thus, the performance and memory requirements for such extensions are like those of native objects in OpenAccess yet they are not publicly visible.

The OpenAccess extensibility API allows applications to extend the attributes on an existing object in the model and to add new independent objects with their own unique attributes and relations with other objects. Extension attributes may be from a list of types including integers, floating point, strings, and pointers to other objects and application defined storage areas.

2.4. Information richness

It is no longer correct to consider the mask set to be a simple necessity in the IC supply chain. Today, masks truly enable the wafer fabrication process. Mask making is no longer a job shop operation but is a key element of the entire flow from design through manufacturing. The industry must find ways to support continued improvement in the mask engineering process beyond simply trying to make the information passed it more compact.

One problem today is that mask makers are not given complete information about the intended design. Instead, they are delivered only data that represents the shapes to be cut into the mask set, and order information, such as critical dimension (CD) acceptability and points where inspections are to be made to assure that the CD is met. Nothing is communicated about the design intent (such as net configuration, or criticality of timing along a path), or which shapes constitute the actual design vs. a design correction made by a RET process and thus, traceable knowledge of how the layout data has been modified for manufacturing. This information could be a valuable resource to the inspection and repair of masks and overall improvement to mask making operations.

The OpenAccess data model contains complete information about the intended design. A complete hierarchical net list, design constraints, physical layout, parasitic values, etc. may all be represented in OpenAccess and RET added shapes may be identified unique from shapes intended in the original design. All of this information can then be made accessible within the mask making or wafer fabrication processes. Further, OpenAccess is extendable to contain additional design information of use by manufacturing and manufacturing information that may be valuable in the design process.

3. OpenAccess reference database features

The OpenAccess package is provided freely to the industry. This package includes the API specification and a reference database implementation that fully supports the API. This reference database is fully tested, and of production level quality and performance. It was written for use on multiple hardware platforms and for either 32 or 64-bit architectures.

It is not a goal of OpenAccess to standardize the use of this reference database code, but only to achieve industry acceptance of the API as a standard. The reference database is supplied to reduce the barrier of entry to OpenAccess and it may be used by commercial applications and within IC design and manufacture flows. Following are some of the attributes of this database that support its use by the industry.

3.1. Built for public use

OpenAccess was developed with an open source model in mind. Thus it was created with the intent that all of its internals would be publicly visible and so that the public could make change to it. The reference database code is not a result of evolution except for its novel techniques and algorithms. The reference database code is highly structured and extremely modular. This code is made available with a complete suite of tests (which assure its correctness and can be used by alternate implementations of OpenAccess conformant databases). This code is also well documented and is, in fact, used to auto-generate the API specification.

3.2. Incremental access

The term *stream* is associated with GDSII because of the inherent sequential nature of its data format. The OpenAccess paradigm is one of direct and incremental access. That is, an application can request information relative to a single specific object (say the list of pins of a specific net) and the database will return only that information. In performing this operation, the OpenAccess reference database does not load or parse the entire set of information it contains. This database is able to load only the set of objects and attributes necessary to fulfill the request into memory. In this way, the overall performance relative to parsing a sequential file in its entirety is greatly improved and the processing time can track with the amount of information the application must deal with rather than the amount of information in the database itself.

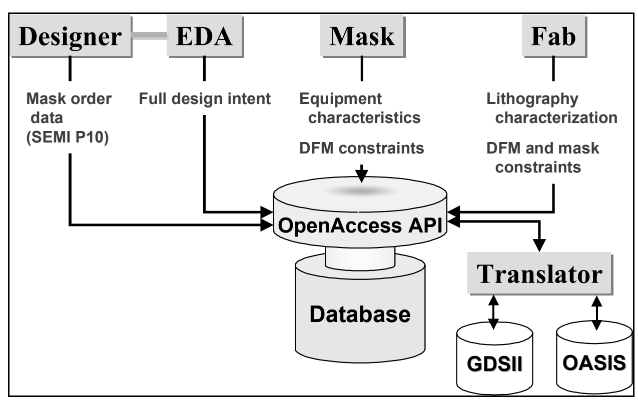
Similarly, information (new or changed) may be written back into the OpenAccess reference database incrementally without necessitating a complete rewrite of the entire database (such as is required with a sequential file). This attribute is also a key element of improved performance across a flow of applications and is of particular importance when the data is shared across multiple disciplines such as design and manufacturing.

3.3. Thread safe

Many feel that one key to improvement in the mask creation times is the use of parallel processing in the mask data prep operations. The OpenAccess reference database is thread safe, meaning that multiple threads within an application can operate on the data in parallel without the risk of one thread contaminating the data on another.

4. Conclusion

OpenAccess provides most of the necessary technical and business requirements required by the UDM. There will no doubt be new information objects and features to be added, but there is a well-defined open community process in place to support this.



With this new paradigm industry will have a direct communication link for information across design and manufacturing. This will be used to improve processes and reduce cost and cycle times while improving quality throughout. Thus, it will become a framework on which to work smarter and not just smaller. This paradigm will not require that the industry move to it together, as it supports a graceful migration via translation to and from existing stream files (GDSII and OASIS). This paradigm is highly extensible and supportive of future innovations, while minimizing impact on applications using it.

---

[1] R.C. pack, et.al., *GDS-3 Initiative: Advanced Design-through-Chip Infrastructure for Sub-Wavelength Technology*, SPIE, 2001

[2] A.B. Kahng and Y.C. Pati, *Subwavelength Lithography and its Potential Impact on Design and EDA*, 1999 Design Automation Conference

[3] M.A. Lavin and W.C. Leipold, *VLSI Manufacturing Shape Data Preparation*, MicroNews, Third Quarter 1999, Vol. 5, No. 3, IBM Corp.

[4] J. Lin and B. Lin, Mask Supply Workshop, SPIE BACUS Conference , October 1, 2001

[5] T.J. Grebinski, *SEMI IC Design/ Photomask Data Path Task Force*, BACUS, 2002

[6] T.J. Grebinski, *The Anatomy of a Universal Data Model*, BACUS, 2001 (www.si2.org/eda-mask)

[7] *OpenAccess Project*, www.openeda.org