Design of Collaborative Engineering Data System (CEDS): an Application Case of Process Integration*

Chih-Min Fan[1], Shi-Chung Chang[1,2], Henry Chang[3]
1. Graduate Institute of Industrial Engineering, National Taiwan University, Taipei, Taiwan, R.O.C. 10617
2. Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, R.O.C. 10617
3. IBM Thomas J. Watson Research Center, Yorktown Heights, New York, USA
fanchihmin@yahoo.com.tw, scchang@cc.ee.ntu.edu.tw, hychang@us.ibm.com

Abstract – This paper presents a collaborative engineering data system (CEDS) design for supporting semiconductor technology lifecycle management in three aspects: change impact analysis, knowledge perspective modeling, and service configuration. The design innovation lies in the generic engineering data object (GEDO), which is designed for dynamic object modeling to wrap, structuralize and integrate all disparate information along an engineering chain. A proposed CEDS lifecycle control mechanism then guarantees the operation efficiency and quality during the CEDS implementation and run-time phases. An application of CEDS to collaborative construction of process integration knowledge serves as a conveyer of ideas.

INTRODUCTION

Rapid development of manufacturable and profitable semiconductor products in the evolution of sub-wavelength technology has posed stringent challenges to the vertical integration among design houses, design service providers and fabs. The process complexity rapidly increases, the coupling among design, engineering and manufacturing gets much tightened, product lifecycle continues to shorten and capital investment is sky-rocketing. Successful engineering collaboration among design, technology development and manufacturing for quick time-to-market and then volume production with high yield is critical to business success [1].

There are three key aspects in collaborative engineering for semiconductor technology lifecycle management as shown in Figure 1:

A1. <u>the efficiency of technology development</u> to facilitate quick time-to-market of products using new-technology;

A2. <u>the effectiveness of technology transfer</u> to maximize the capability for product yield-ramp-up, and

A3. <u>the quality of design and engineering services</u> to guarantee the product manufacturability.

These require a collaborative engineering data system (CEDS) that facilitates integrated management of heterogeneous engineering data among various organizational entities engaged in technology lifecycle.

* This work was supported in part by the National Science Council, Taiwan, R.O.C., under grant NSC- 93-2213-E-002-041 and by IBM under the Shared University Research Award.

This paper presents the design and lifecycle control of a collaborative engineering data system (CEDS). Its application to the collaborative construction of process integration knowledge serves as a conveyer of ideas.

CHALLENGES OF ENGINEERING COLLABORATIONS

Take process integration as an example. The process integration iteratively characterizes and optimizes both the devices and the processes, which is one of the most important tasks during technology development and early product manufacturing stage. Figure 2 illustrates the disparate data/information related to process integration. In the current practice, although a good part of the data has been well organized and presented via the engineering data analysis (EDA) platform, unstructured or miss-linked engineering data/information is still abundant. Three problems arise:

P1. <u>Change management for efficient technology development (TD)</u>:
Changes are the norm during technology development. It is difficult to synchronize all the changes of dependent data with unstructured formats and missing links. For example, the changes in test chip, design rule and related documents are very likely asynchronous with each other, which could result in mistakes on follow-up actions and hinder the efficiency of technology development. The challenge is how to, with high efficiency and quality, evaluate the full impacts by a single change and conduct all synchronized activities driven by this change.

P2. <u>Knowledge management for effective technology transfer</u>
During technology development, engineering knowledge is often kept in TD engineers' minds or buried in unstructured documents. Only significant final-results will be transferred to product manufacturing stage. Much of the experiment data and first-cut analysis information such as the possible relationships among yield-BIN, E-test, and metrology data will not be in the transfer, which may be useful to manufacturing engineers for rapid product yield-ramp-up. The challenge is how to effectively represent, reserve, and communicate all the knowledge data as well as structures along the technology lifecycle.

P3. <u>Configuration management for the quality of design and engineering services</u>
For different products and customers, design and

engineering services are configured differently to meet the requested service quality. Whenever there are new changes in data sources, a service provider needs to provide an active change-notification and a new version of service configuration to customers who previously received the data. The challenge is on the efficiency of version control, where the number of versions grows exponentially with the number of service configurations, the number of data sources, and the number of data versions.

As the semiconductor technology node goes to 65nm, problems P1~P3 are getting more severe than ever due to the unprecedented growth of engineering-data in volume, dependencies, and changes [1].

CEDS REQUIREMENTS

The original idea of CEDS is to integrate, structuralize, and visualize all the engineering data, information, and knowledge. The ultimate goal of CEDS is to build a platform providing various engineering services so that all engineers can collaborate on it to efficiently solve problems P1~P3. To achieve this goal, there are two fundamental requirements:

R1. Disparate data and information technology (IT) services integration

At various levels of the data structure, CEDS should be able to access at least three kinds of data: structured engineering data on manufacturing execution system (MES) and engineering data analysis (EDA) platforms, semi-structured document in document control center (DCC), and disparate documents in individual departments and personal file vaults. In addition, the CEDS should be able to access and reuse other kinds of IT services like data analysis and simulation tools developed in-house.

R2. Dynamic object modeling (DOM)

The CEDS design should have the fundamental capability of dynamic object modeling to wrap, structuralize and integrate all disparate information along an engineering chain. In addition, three advanced methods are required to address P1~P3 respectively.

M1. Change-impact analysis on all related objects to improve both the quality and efficiency of change evaluation and conduction;

M2. Perspective modeling to facilitate various knowledge representations and state-transition control to ensure knowledge lifecycle management, and

M3. Bill of document generation to support various configurations of design and engineering services.

CEDS INFRASTRUCTURE

Figure 3 conceptually demonstrates the engineering collaboration environment after the implementation of CEDS, in which the requirements of R1 and R2 are realized by the following two designs respectively.

Service-Oriented Framework (SOF)

The framework is designed to support the efficient integration of all the disparate data and IT services to fulfill R1. As shown in Figure 4, seven kinds of services are currently identified: UI Tool, Analytic/Simulation Tool, Authority, DCC, EDA, DB and File services. The integration of these services is realized by using the IBM WebSphere Business Integration (WBI) server [2].

Generic Engineering Data Object (GEDO)

Innovation of our design lies in GEDO (Figure 5), which is designed for the DOM in R2. GEDO combines the concepts of resource description framework (RDF) [3] and the hierarchy and flexibility of XML schema for consistent and scalable definition of data object types, attributes, structures and linkages. A set of basic methods for manipulating data objects is identified and provided in GEDO. The interface technique of "drag-and-drop" is adopted for composition of user interface and data binding. The design of advanced methods M1~M3 are realized on top of GEDOs. As a result, users of different background can use GEDO to define various object types and perform various manipulations on them without taking any coding efforts.

CEDS LIFECYCLE & ISSUE MANAGEMENT

CEDS Lifecycle

In the development and implementation of each new engineering application on CEDS, there are four stages:

S1. Analysis: To describe the engineering purpose, sketch the conceptual modeling, and conduct process analysis on the application problem.

S2. Design: With the basic understanding of a new application through the first analysis phase, an engineer needs to further conduct thorough design following the UML format [4]. Specific outcomes of this phase are the use case diagram, class diagram, sequence diagram, etc.

S3. Configuration: Use the GEDO template editor to configure the class identified in the design phase as a GEDO template. The configuration consists of six steps: input template information, define attributes, design UI forms, conduct data binding, implement action links, and conduct validation.

S4. Run-Time Manipulation: In the run-time phase, CEDS will generate GEDO object instances based on the GEDO template configured in phase S3. An engineer with authorized roles and uses will then be able to manipulate and manage the GEDO object instances on CEDS in the run-time environment.

Issue Management

There could be a lot of issues occurring in each CEDS life stage. The issues can be classified into two types: engineering innovation issue (EII) and CEDS system issue (CSI). Most of the engineering collaboration process is to resolve EII, which has no standard operation procedure (SOP). Therefore, it needs the dynamic configuration and manipulation of GEDO to support resolving EII. However, during the process of configuration and manipulation on GEDO, it also results in CSI. Most CSI are accompanied with the EII. The control of CEDS lifecycle is therefore through the fine management of both EII and CSI through the issue submitting, monitoring, and control mechanism.

A PROCESS INTEGRATION EXAMPLE

For the purpose of illustration, consider a process integration example. The collaboration is between engineers from E-Test (ET) integration and Defect Analysis (DA) on constructing a key node table (KNT) for knowledge management [5, 6].

S1 Analysis

The KNT is a result of knowledge engineering. Figure 6 depicts a KNT example, where the columns correspond to E-Test items while the rows correspond to process steps. The values in individual KNT cells empirically correlate the abnormal E-Test data (symptoms) to major process steps (root causes). In each cell of a KNT, the number in the upper-left(upper-right) sub-cell represents the level of correlation between a small(large) deviation of a E-Test data item and the corresponding process step; the higher the number, the stronger the correlation. The sign in the lower half of a cell provides the direction of E-Test data deviation should a process step be faulty. Both the upper numbers and the lower signs are called dependencies of individual E-Test data items with their corresponding process steps. Each power user of KNT specifies the cell contents in the KNT that the users would like to see. A power user also specifies KNT rules that check the consistency among KNT entities, i.e. the cell contents in a KNT, input by different engineers.

Figure 7 gives an overall conceptual model for the collaboration between ET and DA engineers on integrating the processes of ultra-shallow co-silicide junctions. In specific, objects and their links circled in Figure 7 correspond to constructing the dependency knowledge in a KNT.

S2. Design

Figure 8 further elaborates on the collaboration process between ET/DA engineers and power users for KNT dependency knowledge construction. For conciseness of presentation, ET/DA engineers will be referred as engineers and power users as users in the later discussions. There are four stages of the process:1) KNT template configuration and rule setting by users, 2) KNT entity input by engineers, 3) collaborative consistency check of KNT entities and collaborative management of KNT issues and 4) consensus KNT entity setting. The first two stages are performed by individual users and engineers respectively. In Stage 4, users set the consensus KNT entities. Stage 3 is the key to facilitate the collaboration.

The first collaboration via stage 3 is the consistency check of KNT entities defined by engineers with respect to rules specified by users. The loop from “Update KNT Entities” to “Check KNT Event Rules” in Figure 8 facilitates another level of collaboration. The loop allows the feedback from specific engineers to go to all the collaborating users and engineers for not only consistency check but also KNT issue management. .

The second collaboration via stage 3 is the management of KNT issues. The issue management life cycle consists of four stages: identification, notification/presentation, review and action/comments. The feedback from “Comment on KNT Issues” to “Notify Related Engineers About KNT Entities & Issues” closed the collaboration loop, which includes (1) collection of comments, reasoning and actions on issues (2) transparent presentation/notification of feedback to all parties and (3) systematic tracking of issues.

S3. Configuration

The collaboration process for KNT construction depicted in Figure 8 helps us identify the KNT related GEDOs and their corresponding attributes and methods. All the identified GEDOs will be configured by a GEDO template editor. Figure 9 demonstrates an example of using the GEDO template editor to configure the GEDO template of a KNT entity via 6 steps with individual functional links displayed in the left column of the window.

S4. Run-Time Manipulation

During the run-time phase of CEDS, engineers may manipulate the KNT related GEDOs for collaborative KNT construction. For example, the ET engineer can apply both basic and advanced methods (displayed in the left column of Figure 10, a window of KNT Entity GEDO instance manipulator) to edit the dependency knowledge between E-Test and process control items.

CONCLUSIONS

In summary, the GEDO modeling facilitates the integration of structured and unstructured engineering data. The former includes MES and EDA data while the latter includes information/knowledge spread over engineering documents in the forms of reports, comments, notes, etc. Combining GEDO with computing services such as change impact analysis, perspective modeling and bill of document generation, CEDS provides a platform for collaborative engineering. In addition, both specific designs of GEDO template editor for dynamic object modeling and CEDS lifecycle control mechanism greatly enhance the CEDS operation efficiency and quality.

REFERENCES

[1] International Technology Roadmap for Semiconductors (ITRS), 2004. http://public.itrs.net/

[2] WebSphere Business Integration Server, http://www-306.ibm.com/software/integration/wbiserver/

[3] Resource Description Framework (RDF), http://www.w3.org/RDF

[4] G. Booch, J. Rumbaugh, and I. Jacobson, The Unified Modeling Language User Guide, *Addison Wesley*, 1998.

[5] C. Fan, R. Guo, S. Chang, and J. Wei, “SHEWMA: an End-of-line SPC Scheme Using Wafer Acceptance Test Data,” *IEEE Transactions on Semiconductor Manufacturing*, vol. *13, no. 3,* pp. 344-357, 2000.

[6] R. Guo, C. Tsai, J. Lee and S. Chang, “Intelligent Process Diagnosis Based on End-of-line Electrical Test Data,” Proceedings of *International Electronics Manufacturing Technology Symposium*, 1996.

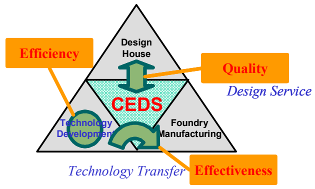

Figure 1: Three key aspects in collaborative engineering for semiconductor technology lifecycle management

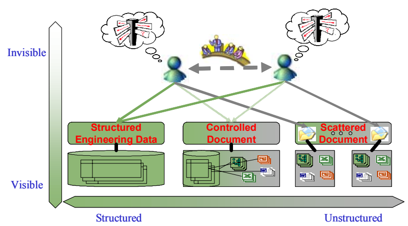

Figure 2: As-Is: process integration with disparate engineering data

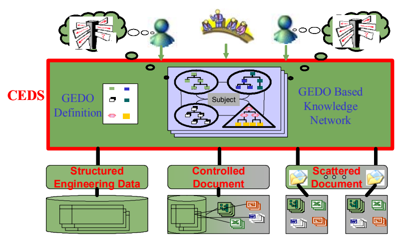

Figure 3: To-Be: process integration with integrated engineering data provided by CEDS

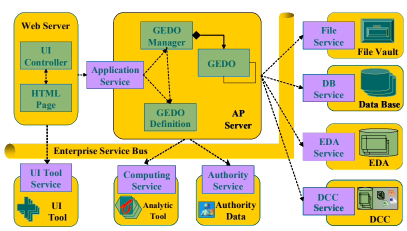

Figure 4: CEDS Service-Oriented Framework

	GEDO Template	↔ Binding ↔	CEDS Resource
View	GEDO UI Form	XML	UI Components
Control	GEDO Method Access		Advanced Methods Basic Methods
Model	GEDO Data Modeling		Meta Data

Figure 5: Concept of GEDO template configuration

Data Deviation

Key Node	Vt_NL	Vt_N	Isat_N	BV_N	Rs_N+	Rc_N+
P+ S/D Photo (Blanket)	1 1 +	1 1 +	1 1 -		1 1 +	1 1 +
NLDD-Implant (Dosage)	0.22 0.78 -	0.22 0.78 -	0.78 1 +	0.22 0.78 -		
N+ S/D Implant (Dosage)	0.22 0.22 -	0.22 0.22 -	0.22 0.78 +		1 1 -	1 1 -

Figure 6: An example of Key Node Table

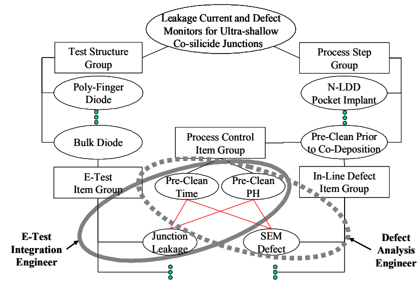

Figure 7: An example of process-integration conceptual modeling

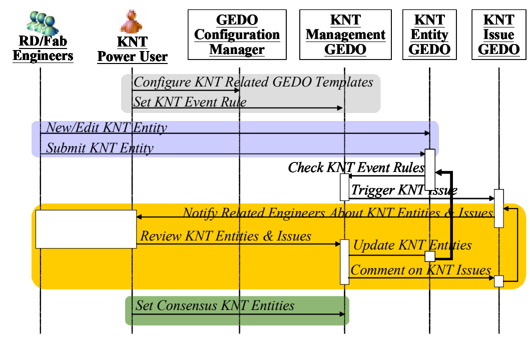

Figure 8: Exemplary collaboration process for key node table construction

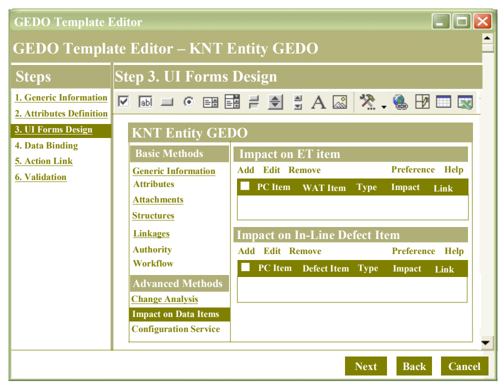

Figure 9: GEDO Template Editor

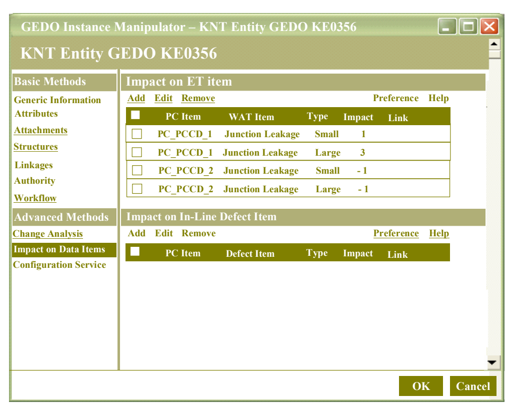

Figure 10: GEDO Instance Manipulator