INTEGRATED ENGINEERING, GEOMETRIC, AND CUSTOMER MODELING: LCD PROJECTOR DESIGN CASE STUDY

Shaun Abrahamson  
Graduate Research Assistant  
Department of Mechanical Engineering  
Massachusetts Institute of Technology  
Cambridge, MA 02139  
Shaunabe@mit.edu

David Wallace  
Esther and Harold Edgerton Assistant Professor  
Department of Mechanical Engineering  
Massachusetts Institute of Technology  
Cambridge, MA 02139  
drwallac@mit.edu

Nicola Senin  
University of Perugia, Italy  
Nsenin@unipg.it

Nick Borland  
Research Specialist  
Department of Mechanical Engineering  
Massachusetts Institute of Technology  
Cambridge, MA 02139  
nborland@mit.edu

ABSTRACT
This paper describes an integrated product design study conducted with Polaroid Corporation for a liquid crystal display video projector, applying a research system called DOME. The services of distributed objects—encapsulating CAE simulations, component catalogs, manufacturing cost models, geometric and configuration models, customer preference models, and environmental life-cycle assessment—are mathematically related to form an integrated product system model. Software objects providing optimization and decision support are also included in the model to create a design tradeoff environment. As such, designers can obtain sales predictions based upon configuration changes and make tradeoffs with other requirements. Benchmarking suggests there would be approximately a 30% increase in the time to fully evaluate the first design configuration due to the overhead of creating the integrated system model. However, the time to fully evaluate subsequent alternatives may be reduced from months to minutes.

INTRODUCTION
Predicting overall product performance and making informed design decisions can improve quality and speed development by reducing costly design, build, test, and refine cycles. Consequently, Center for Innovation in Product Development (CIPD) industry partners (CIPD 1999) have expressed a strong interest in integrated product design modeling. Integrated models should link design performance attributes derived from predictive models or simulations with tradeoff assessments between, for example, customer satisfaction, market penetration, environmental impact, and return on investment.

Figure 1 illustrates a scenario for the design of a liquid crystal display (LCD) projector—design capabilities are distributed geographically. The original equipment manufacturer (OEM) wants to foresee the design as a whole, trading off cost, sales, technical performance, and environmental impact. In-house divisions of the OEM provide manufacturing cost estimates (using Microsoft Excel) and thermal analysis (using custom software). A mechanical engineer and industrial designer use SolidWorks to define and
detail the geometric assembly, while an environmental consultant uses EcoBilan TEAM software to estimate the environmental friendliness of the product. A marketing model estimates sales using proprietary software (not shown) while an overseas light engine supplier provides the OEM limited technical data about their product. Each design participant uses different software tools, models, and data, ranging from solid modelers to spreadsheets.

In this scenario, the OEM wishes to consider various light engine suppliers. The components from each supplier effect many aspects of the product design, as shown in Figure 2 when a different light engine vendor is chosen. Ideally, the OEM could propose a substitution and foresee the resulting system-wide implications. Hypothetically, effects of this change might include changes in overall product form, the manufacturing cost, internal temperature, the MTBF of the printed circuit board (PCB), environmental impact, and expected sales. All of these changes impact the decision to accept the light engine change.

In current practice, obtaining an integrated view at this level of detail for a single design alternative is, at best, very time consuming, typically requiring months to coordinate and resolve. At worst, obtaining this global system view is deemed intractable and key decisions are left to intuition. The goal of this paper is to apply a prototype research system called DOME (Distributed Object-based Modeling Environment) to explore the feasibility of constructing comprehensive integrated models. These models will be used to predict overall product performance in an effort to reduce the time needed to evaluate design option tradeoffs.

Challenges and desired characteristics of an integrated modeling environment are discussed. Different approaches to integrated modeling are reviewed and the DOME concept is briefly described. The LCD projector model is introduced and used to illustrate how customer preference and sales predictions can be provided to designers along with the prediction of product performance. Finally, estimates comparing the integrated modeling approach to a traditional design process are provided.

BACKGROUND

In product development, the combination of both focused analytical modeling and comprehensive system modeling has traditionally been deemed infeasible (Ulrich et al 1995). This is in part because of the difficulty in modeling some elements of the design, but also simply because of integration difficulties associated with model scope and logistics associated with the connecting different sub-models. This is reflected in the following three statements (Cutkosky et al 1994).

1) It is very difficult, if possible, to formulate a complete explicit product development model, both because of its size, complexity, and its dynamic, uncertain and evolving nature.
2) Different product development domains use different tools, data models, and often even different data management systems
3) All necessary data are not globally available, because of consolidation, logistics or proprietary issues.

In spite of these issues, numerous integrated modeling efforts continue because there is great potential for savings in development time by avoiding expensive, slow physical iterations and focusing on fast and cheap analytical iterations. Other researchers have also described requirements for integration frameworks (Tomiyama 1994b; Molina 1995). Drawing upon this work and data gathered from CIPD industry
collaborators (CIPD 1999), key requirements for an integrated product development system include:

1. Information system architectures that allow distributed users in different environments to participate without investing significant training time into systems, tools, or modeling languages outside of their own expertise.

2. Systems allowing each design participant to use the tools, representations, simulations, heuristics, or models which are most suitable within their domain.

3. Flexibility to allow for the spontaneous and robust growth, extension, change, revision and reuse of integrated models, tools, or resources to solve evolving or new problems.

4. Incorporation of a seamless mix of detailed models and incomplete or approximate models to support both top down and bottom up design.

5. Providing the ability to explore a design solution space, elicit trade-offs between participants and goals, and monitor design evolution in both a manual and automated fashion.

6. Support for decision-making through user feedback or decision theory.

7. Accommodating both tight and loose collaboration, ranging from close colleagues to outside consultants or suppliers while respecting a diverse set of intellectual property and synchronization needs.

We believe that these integrated modeling challenges and desired capabilities are dominated by a variety of communication and coordination issues—communication between designers, communication between designers and tools, and communication between tools (adapted from Branki (Branki 1995)). The work described in this paper focuses on these types of interactions.

Integration can be approached in a variety of ways, but we see research efforts as a continuum bounded by two approaches. One extreme requires all participants to represent their views in a standardized way so that it can be understood by other people or computational models. An example of such standardization is STEP (Owen 1993), which is intended to provide an all-encompassing data representation. The opposite pole focuses upon translating between elements that need to communicate. An example is the development of ontologies to define exchanges between entities. In the case of ontologies, interfaces between elements are added as needed (Cutkosky et al 1994). Within this continuum are intermediate representations such as qualitative process theory, that allow integration by mapping concepts from one modeling domain to a general representation, and then to other domains. For example, Tomiyama (Tomiyama 1994a) uses qualitative process theory as an integrative language mapping each sub-model to a global representation. While this has potential for automating the integration process, designers are required to understand a global representation, which might require revision each time a new interface requirement was identified.

The DOME research tool focuses on letting design participants use the tools with which they are familiar—representations are not prescribed. The assumption is that representations for different domains have evolved to best suit that domain. This approach does not preclude the use of standardized representation, but acknowledges that the development of an all-encompassing representation is very challenging. Like work by Cutkosky (Cutkosky, Engelmore et al. 1996), users are allowed to wrap applications and publish interfaces. However, rather than use ontologies to automate inter-domain integration, the focus is on mechanisms that allow participants to create, connect, and use modeling services.

Overview of the DOME concept for integrated modeling

The integrated product modeling prototype used in this case study attempts to address the issues identified in the previous section. It assumes that different aspects of a product design problem are delegated based upon disciplinary expertise or organizational structure, as suggested by the scenario in the introduction. Each participant uses their own modeling tools and representation to address their part of the problem, and then use a simple publisher application to build an object (module) that provides access to their modeling services through the world-wide-web (WWW). For example, a publisher for an environmental impact assessment application is described in work by Borland (Borland et al 1998). This interface allows other participants to manipulate individual inputs and observe results through a homogeneous web browser-based environment.

The web-based environment also allows participants to define new modules containing mathematical relationships capturing the interactions between the input and output services of different models, and to create new services. These relations create a service exchange network that becomes a distributed computational system model (Pahng et al 1998; Senin 1999b). The propagation of input/output service changes is coordinated by relations so that each sub-model responds to a new design state. This provides behavior not unlike that found in blackboard architectures like “design sheet” (Reddy and Fertig 1996) without an explicit, centralized system model. This is particularly important when all participants in the supply chain will not be willing to expose their data and models in a centralized system.

In addition to providing an environment for publishing models and relating services, special software modules are also created to provide services for optimization, decision support, and tradeoff analysis.

Agent-based approaches for integrated modeling are becoming increasingly popular (Cagan et al 1998) (Parunak 1998). Therefore, it is useful to place this approach in the context of this work. We limit our scope to systems that are
modular, decentralized, and relatively easy to extend or change. Further, DOME does not require central definition of connections between sub-components and assist the user in a variety of ways. In this definition, the DOME framework can be classified as an agent-based system. However, we have not focused on the automated synthesis of models.

**LCD PROJECTOR OVERVIEW**

The objective of the Polaroid LCD design study was to develop an integrated model linking product design changes with real time feedback for predicting technical performance and expected sales. The scenario described in figures 1 and 2 closely mirrors the structure of the integrated model. However, all models currently reside on the MIT CADLab internal network, with different computers assigned to each of the sub-models depicted as geographically distributed. Further the research software used in this case study is based on a second generation DOME kernel using a MOTIF-based user interface (unlike the web-browser-based environment that is currently under development).

Polaroid acts as an integrator, assembling a variety of standard and custom components from different suppliers and outsourcing services such as industrial design or market analysis. Final sub-models included geometric and thermal models, in addition to consultant models for environmental assessment, cost, and market analysis. This section describes the model at various stages of evolution and highlights integration issues that were encountered.

*Evaluation of core technology (light engine) alternatives and suppliers*

Figure 3 shows the graphical user interface for the initial engineering model developed to assess light engines and their suppliers. The purpose of this initial model was to benchmark Polaroid’s current product offering with new light engine technologies. Each circle is the visualization of a module delegated to provide services about an aspect of the problem—characterizing light engines, assessment against technical specifications or metrics, and vendor quality assessment. Lines between modules indicate the existence of service exchange relationships.

The light engine module is a container, encapsulating data and relations defining the characteristics of a given light engine (LE). Figure 4 illustrates the contents of the LE module corresponding to the benchmark design. Many modules, such as the uniformity, provide static data, while heat dissipation is mathematically related to the efficiency and power modules. In this case, if the LE power data are changed the heat dissipation will respond accordingly.

The light engine specifications and vendor intangible modules are decision support modules. These modules are used to evaluate states, providing: an interface for defining preference functions, mechanisms for relating preference functions to design attributes, and services which assess attributes relative to preference functions. Different decision modules can be provided for different decision methodologies—in this application an acceptability-based decision module is used (Kim and Wallace 1997).

The decision module provides a graphical assessment interface, shown in the right of Figure 3 for the LE specifications module. The LE specifications module defines technical expectations based on today’s standards for brightness, uniformity, contrast and artifacts (green or light gray). The vertical red lines illustrate the light engine attributes, received through services of the LE module, relative to preferences. From this assessment, one can see that brightness and contrast of the benchmark LE are poor. If data within the LE module change, relations mapping the LE module services to attributes used by the LE specification decision module would coordinate service propagation so that the decision outcome updates to reflect the new design state. The LE vendor decision module provides a similar assessment of vendor qualities. The preferences defining vendor quality were determined using an on-line Delphi study at Polaroid.

*Figure 4 Modules within the light engine container module. Many of the modules are individual data points, while some compute their values using the services of data modules (such as Thermal Energy). The adjustment of parameters such as uniformity result in changes to the evaluation scores.*

This preliminary model clearly illustrated that different light engines presented interesting technical performance and supplier attribute tradeoffs. Therefore, the next step was to add...
the services of a Manufacturing Cost module. A preliminary estimation module was created based on a simple scale factor of the LE.

Figure 5 shows the new manufacturing cost module. This module needs services for the LE cost and a cost scale factor, which can be mapped through the graphical interface to create a causal service chain. Once connected, the preliminary manufacturing cost estimation is available for any LE chosen from the LE catalog.

As more components were added, a detailed cost model replaced this initial estimation, considering aspects such as component, manufacturing, and assembly costs (figure 6). The cost model is a Microsoft Excel 97 spreadsheet, which requires inputs such as sales volume, material masses, and component costs, and then outputs a total manufacturing cost. The spreadsheet also controls the cost budgets, which are passed back to the DOME environment, for use in the cost evaluations.

This stage of the case study illustrated the need for model flexibility and extensibility. The initial cost estimation module is replaced by a more detailed model, and new modules are added to provide additional information needed by the more detailed cost model. Further, different modeling tools are introduced without sacrificing integration, allowing participants to use the tools of their choice. The costing expert was not required to learn a new environment, so barriers for participation were reduced.

Although the product model was still relatively incomplete, it was already difficult to identify the best product configurations. Therefore, an automated design search module was added to the model, as is shown schematically in Figure 7. Services of independent design variables (catalog choices or continuous data) are made available to the search module so that they can be manipulated. The services of decision support modules are used to construct an objective function. The particular search module implemented for the prototype system moves through the space, identifying combinations of variables that yield higher scores corresponding to global and local optima (Senin, Wallace et al. 1999a).

The result of an automated search indicated that there were attractive alternatives relative to the benchmark design. However, although detailed specifications for the main components were available, housing geometry characteristics were simply estimates (similar to the initial manufacturing cost model) and not tied to a real geometric model.

**Addition of a geometric model and process evaluation**

The model was very dependent at that point upon geometric data that had been provided through simple estimates; therefore a Solidworks™ (Solidworks 1999) geometric model was developed. The improved model is shown in Figure 8. The CAD designer built geometric models for the
different components represented by modules in the engineering module. Assembly and constraint relationships were then defined, and services of the model were made available via a module connected to Solidworks\textsuperscript{TM}. Once these services were connected, each subsequent change in the engineering module resulted in a change in the CAD model, based on information about which components are chosen from various catalogs. In turn, the CAD assembly would reconfigure and provide services, such as housing mass, footprint area, and volume, to the engineering model.

Figure 7 Schema showing how the optimization module interacts with services of other modules in the design model.

As the scope of the model grew, the decisions about how the model should evolve became increasingly difficult. In areas where the existing models were inadequate, it was desirable to procure new services to augment these areas; but it was not always obvious which capabilities should be extended. For this reason, network analysis was used to identify critical areas within the integrated model.

The design structure matrix (DSM) has been used in a variety of applications to formalize and manage the design process (Eppinger 1994 et al; Eppinger et al 1997). A DSM can be created from the service exchanges and modules in the integrated model to guide model evolution (Abrahamson 1999). For this reason the analysis of the integrated model was conducted using the DEMAID tool (DEMAID 1999). The model was recreated in a different representation and the result indicated the critical position of geometric information. A more recent implementation allows this information to be extracted directly from the integrated model, allowing the design process to be managed in real-time, considering the evolving structure of the integrated model.

This analysis indicated a true geometric model was critical at this time, as geometric services (such as dimensions, volumes) were needed by many other sub-models. Material used for costing and thermal calculations can be completed more accurately in response to changes to configurations and part dimensions. Further, it provides much of information that will be needed for a market and customer response model.

Figure 8 Addition of the detailed geometric model. The models allow reconfiguration in reaction to different combinations of components.

Figure 9 Design Structure Matrix for the Polaroid model based on the analysis of the interfaces proposed by each of the participants. The analysis was used to focus attention on the construction of a geometric model.

Environmental impact assessment and sales predictions

The final stage of the integrated modeling study was to incorporate the services of environmental life-cycle analysis and customer response market models. This allows engineers to obtain real-time feedback about anticipated sales implications and environmental impact. The final system model is illustrated in Figure 10.
Environmental impact assessment services are provided by a life-cycle model created using EcoBilan’s TEAM software. Like the Excel and Solidworks models, these services are made available using a custom module, providing designers life-cycle analysis feedback using a variety of impact assessment schemes. The ability to incorporate services from any third party software application is dependent upon that application’s user providing CORBA-compliant (or DOME-compatible) services to other users. Clearly, it is not realistic for each participant to have this expertise, so service publishing programs are needed. Work by Borland (Borland, Kaufmann et al. 1998) details this concept.

Figure 10 The final integrated projector design model, highlighting links to other applications. The link to the environmental impact assessment program TEAM is not shown.

The second extension, providing real-time sales estimates (based upon design attributes), involves the connection to a market model (Savage 1998). This model first converts engineering attributes to customer attributes and uses the preference structure of target segments to predict customer utility. It then computes the probability of purchase based on utility relative to competitors and provides an estimated annual sales volume as a service.

The services of this model were linked with the engineering model to operate asynchronously. While all other service relationships were synchronous so that a service change automatically propagated through the network, the engineer could withhold changes and then explicitly request the sales prediction service on selected options.

ANALYSIS OF THE PROJECTOR DESIGN STUDY

The projector design modeling study took place over a three-month period. Over the course of the project, participants logged their activities so that impact on product design time could be explored. From this data, it was possible to estimate the overhead of model integration and the potential benefits of service coordination and rapid propagation of changes.

Model integration overhead.

The integrated model or service exchange network was built up in a series of steps following the needs of the project. Like a traditional process, participants were allocated different tasks. Each participant’s task can be viewed as an information processing function, where they are given a set of data inputs from which they are to generate a set of outputs. These might include constraints, objectives of other people, overall design goals, or the attributes of various elements.

Approximately 30% of the design team’s effort (totaling roughly 21 person-months) in the evolutionary process was devoted to the explicit definition, negotiation, and verification of services that would be needed or provided by each sub-system. This is conservatively viewed as overhead that would not exist in a traditional design process. Therefore, we estimate the first integrated assessment would take 1.3 times longer due to overhead associated with creating the integrated design model.

Exploration of alternatives

In a traditional non-integrated environment, the delegated tasks are undertaken in partial isolation. At an initial time, the group meets and agrees on a certain state of the design as well as the input-output structure (or task allocations). This input-output structure forms a model or problem topology. From this point on, they will work on their individual parts of the problem for a period, during which a number of changes occur in the model structure and state.

As the project proceeds, information updates will be sought by a participant, asking other team members for new information and there will be lag until a response is received. At the same time, the participant will be making changes that may affect other participants. In both cases, the state of the inputs and outputs is different from those understood initially. It becomes clear that errors and conflicts will emerge between assumptions in different sub-systems, so timely coordination and propagation of information is important.

Each time a new alternative or change is considered, this complex communication (or service exchange) process must be repeated. Figure 11 shows a simplified map of how a service indicating a change in the power supply needs to propagate to get a new sales prediction from the customer model.

Using traditional communication lag times only ((Osborne 1993) and surveys at the sponsor company) it should take approximately three months to obtain a new integrated assessment for a configuration change, such as the use of a different power supply. The same process requires about 20 seconds in the integrated system. Assuming that three major
iterations would occur in a traditional process, we estimate it may be possible to reduce overall design time by up to 50%. Regardless, the speed of the integrated process creates a new possibility for exploring many options by optimizing the product from a system viewpoint.

Figure 11 Simplified network showing the service propagation resulting from a power supply change in response to a change in power requirements.

CONCLUSIONS AND CHALLENGES

The ability to develop models to predict a product’s performance is becoming an increasingly important component of competitive product development. This paper presents a view where integrated models are created by building a computational service exchange network, thereby interconnecting the input and output services of different design participants. Using this approach, the development of an integrated design model is a flexible and evolutionary process occurring in tandem with the development of a product. This is different from the traditional view that integrated models are tools that are programmed completely and then applied to design a number of products.

Integrated product modeling is complex because of the difficulty of formulating a complete explicit product development model, the use of different tools in different domains, and proprietary barriers in the supply chain which prevent data or model consolidation. To overcome these barriers, it was determined that an integrated modeling environment must accommodate distributed participants and allow these participants to use their own tools and provide services with minimal overhead. It must also be flexible so that the integrated model can grow and evolve, provide design support such as decision-making or optimization, synchronize service needs.

The LCD projector design case study was used as an example to illustrate how a service network model created using the DOME research system might address these requirements. The study illustrated the integration of services coordinating the design configuration: engineering performance simulations, the geometric assembly, manufacturing cost estimation, environmental impact assessment, and real-time sales estimation. Additionally, it illustrated how services from modules for decision support and configuration optimization can be integrated with the design model. Based on preliminary data tracked during the study, the overhead of integration increased the time needed for the first integrated assessment by an estimated 30%. The speed of subsequent iterations, however, may decrease from months to minutes, possibly reducing overall design time by up to 50%. A new field study to obtain better estimates for impact on design time is in progress currently.

Although results are encouraging, numerous challenges remain for integrated modeling based upon the service exchange concept. Some of these challenges are outlined below.

Computational speed and complexity

As models are connected the computation involved increases. In many cases adjusting a parameter will trigger multiple models, which will begin re-computing, resulting in computation times from a few seconds to hours or even days. This is unacceptable in a real-time feedback scenario, or for optimization. A solution may be the use of surrogate models for high speed, approximate evaluations.

Comprehension and navigation

Making sense of a distributed service exchange network will require an understanding of how modules and their services are connected, whether for debugging, verification, or learning. Different types of changes include parameter state changes as well as model structural changes. Often, however, no centralized view of the model is available, because of proprietary service relationships. Appropriate debugging, processing and visualization techniques are needed to facilitate these activities.

Service quality and verification

The performance of an integrated model is determined by the quality of sub-models. A measure of quality is required to allow users to display confidence in the model results and communicate service errors so that the results will be interpreted suitably. This is important because in many instances, services from a domain other than one’s own may emanate from ‘black boxes.’

Organizational change

The construction and use of service-based integrated models has organization implications. The approach represents a change in the way people will think about their work and their relationship to the work of others. Although we can address human-computer and inter-computer interactions using software, inter-personal communication and organizational changes may pose the largest barrier to the adoption of this approach.
ACKNOWLEDGMENTS

This case study was made possible through funding by the NSF Center for Innovation in Product Development, grant number EEC-9529140. The authors are also grateful to Polaroid for their collaboration and support, and to Tom Almy, Tina Savage, Ines Sousa, Priscilla Wang, and Jun-Beom Kim for their modeling contributions.

REFERENCES


http://fmadwww.larc.nasa.gov/mdob/users/rogers/Current_research/DeMAID.html


Senin N, Wallace D, Borland N (1999a) "Mixed Continuous and discrete catalog-based design modeling and optimization”, accepted for publication in the 1999 CIRP Design Seminar.


