

SECTION 2

PART I: SURVEY OF CURRENT RESEARCH AND TECHNOLOGIES

2.1 INTRODUCTION

We identified four technological domains related to TO authoring, and more importantly TO validation. These domains included (a) product data management (PDM), (b) virtual design engineering and manufacturing, (c), concurrent engineering, and (d) human form modeling. Virtual design tools produce the models necessary to validation simulations. PDM systems deliver this information to the validation application. Concurrent engineering practices and supporting software, including human modeling applications, allow authoring and validation to be pursued while products are still in their design stage.

2.2 TECHNOLOGY DOMAINS RELATED TO TECHNICAL ORDER VALIDATION

2.2.1 Product Data Management

PDM [3] is an enterprise-wide framework aimed at modeling and tracking all of the data concerning produced goods and services as well as related processes. PDM was initially developed to organize and store data pertaining to engineering activities in a company producing industrial, transportation, and consumer goods. A PDM system stores the design, manufacturing, and maintenance data for each product in a uniform framework, and also manages the processes critical to a product's life cycle. PDM is increasingly used on a larger range of products such as buildings, bridges, factories, cable networks, software, and services. Since PDM is a general framework, it can be used for any production activity. Furthermore, its scope is wide since anyone who deals with products consumes or creates PDM data.

Product data generally consists of specifications, configuration data, CAD/CAM/CAE files, manufacturing data, revisions, and maintenance manuals. However, it also extends to financial and marketing documents. In any case, a PDM system can be scaled up or down to manage specific disciplines of a company. PDM covers the entire life cycle of a product - design, testing, manufacturing, support, and maintenance. In particular, it supports concurrent engineering functions.

The core of a PDM system is its *data vault*. This *metadatabase*, or database of databases, inventories every product datum in the system by maintaining an associated *metadata* record. It contains format, location, ownership, security, and revision information. The data itself is stored in an application-specific database. CAD/CAM/CAE software and other applications can directly access a PDM system to store and retrieve data.

The major functions of a PDM system are listed below:

- **Uniform Data Referencing and Access:** Objects or data records can be referenced with a unique name by different databases. They can also be accessed uniformly by various applications.
- **Process Management:** A PDM system can model and manage data workflow. It can trigger and monitor specification, design, approval, revision, or any other business process.
- **Data Administration:** The content of PDM metadata allows access privileges (security) to be managed, authorship to be recorded, and multiple versions of a single datum (revision control) to be tracked. Furthermore, all the data concerning a product can be transferred, backed up, and archived as a single block.

Many vendors offer PDM solutions. The Object Modeling Group developed a *PDM Enablers* specification [4] to promote interoperability between different PDM systems. For more information on PDM see CIM, 98.

2.2.2 Virtual Design Engineering and Manufacturing

For decades, weapon systems have been modeled with CAD software. More recently, emerging technologies have supported other related processes such as engineering, prototyping, and manufacturing. Off-the-shelf CAE software can model and test the mechanical, thermal, and structural properties of a product before its first prototype is even built. Similarly, CAM software is used to develop molds, stamping tools, weaving patterns, and machining paths from CAD models.

The advent of virtual reality (VR) has spawned the development of virtual prototyping applications that enable a product to be assembled, inspected, and tested in a

virtual world. Some aspects of designs can be tested for maintainability and human factors while they are still on the drawing board. More generally, interactive 3D visualization is used at every step of a design cycle to view single parts, animate assemblies, visualize scientific data, and create marketing and technical documentation.

Current CAD models are *parametric*. Along with their geometric description, they include dimensioning constraints that relate different parts. For example, the diameter of a shaft and the bore it runs through can be constrained such that the bore is updated when the diameter changes. These parametric models allow encoding some of the design intent into the CAD data.

Knowledge-based engineering (KBE) goes further in this direction by capturing design expertise. With a KBE approach, a design bureau can record the lessons learned from past projects and reuse them in the future. For example, a KBE system can help choose an energy source or manufacturing process. A KBE application is mostly an expert system.

It is now possible to go “virtual” though most of a product’s design. Software vendors sell integrated computer-based solutions that support virtual design and manufacturing [5]. Nevertheless, “hands on” virtual prototyping applications, such as technical order validation, remain experimental [6].

2.2.3 Concurrent Engineering

Traditional product design is a chain of sequential development stages going from conceptual design to a finished product. Each stage deals with a specific aspect of the product, such as engineering, manufacturing, prototyping, testing, and servicing. When a design fails to satisfy the constraints of a given stage, it is sent back to one of the earlier stages for redesign. This process can be very costly, because most design defects are discovered during the later stages when changes are more expensive to correct.

Concurrent engineering [7] attempts to reduce development costs by accomplishing each development stage in parallel. This method is particularly challenging because it relies on the collaboration of specialized teams. Aside from the organizational difficulties, concurrent engineering relies on collaborative design environments to share models and ideas.

These environments are built upon available CAD/CAM/CAE software integrated within a PDM system. In particular, the PDM system allows the versioning and review processes to be shared across design teams. Along with these core applications, simpler and “lighter” CAD model visualization tools are being used to communicate the design’s shape and function to a wider audience within and outside a company. These tools allow models to be annotated to simplify review processes. Some are geared toward collaborative design sessions, allowing remote users to share the same virtual space where they can manipulate, modify, and annotate a 3D assembly in turn while communicating through voice or video links [8].

TO validation fits within a concurrent engineering process as a co-design activity. Concurrent engineering can help authors identify maintainability flaws early in the design process and prevent cost overruns and delays later during physical prototyping.

2.2.4 Human Models

Computer-graphic human form models (referred to as *human models*) have been available for 25 years. Significant developments occurred during the last decade, as computer power and three-dimensional graphics improvements have led to interactive models with sufficient biomechanical accuracy to allow their use as ergonomic evaluation surrogates. These models allow figures with anthropometric variations based on a sample population and represent body shape with more or less smooth polygonal surfaces and adjustable joints. The more capable human models provide mechanisms to control the actions of the model, for example, through a walking algorithm, inverse kinematics limb reach, and automatic satisfaction of balance and other postural constraints. Additional improvements include analytical reports on strength, visibility, reach zones, comfort zones, and lifting hazards.

As human models mature, they appear to be departing from stand-alone systems and assuming a more integrated role in the design engineering process. This requires that they interface with CAD models and the design process at increasingly early stages of the product life cycle. Where human models were used primarily by human factors engineers, they are now used by engineers throughout the design process. This shifting of responsibility should affect both the engineering process and the need for human modeling software. Engineers

will be able to easily perform cursory human use analysis as part of form, fit and function analyses.

Features important to manipulation and maintenance (as well as manufacturing) should find their way into the Product Data Models (PDM) to be shared across the engineering enterprise. We have decried the lack of what we called *maintenance features* that would allow a human model to understand a device in terms such as handles, connections, contents, and even function. As designers use human models to evaluate their designs, we hope they will note these features, sites, parts, and contents in the PDM databases so that such information can be used elsewhere by the design team. Besides helping the human factors analyst, such annotations will clearly help the TO author. Annotations that relate part features to CAD features are now inserted manually by TO staff and used by the TO author to create callouts in the graphic images that accompany and amplify the TO steps and text. It seems inefficient to ask the TO authoring staff to insert information that is already known to the design engineers. Although it is not within the research scope of this project, we hope that PDM systems will emerge to reinforce good labeling habits. The alternative – directly automating the determination of maintenance features from the CAD data – is a fascinating research project but appears unlikely to be economically justifiable in actual practice.

The second effect of this shift in responsibility is related to the design of human modeling software. Early human models required that each joint be posed manually, sometimes through a tediously-created and non-intuitive data file. Interactive systems ameliorated some of these problems, but not enough: only the development of robust and flexible inverse kinematic algorithms made human models usable. As the human models are integrated into enterprise-wide CAD systems, users will want better software tool integration and easier-to-use interfaces. A proven approach to the former is through a software library and application program interface (API) that allows another system (such as a host CAD package) to access and control the human model. The user interfaces for the human model, in turn, are expected to resemble those of the host software. The best example of this situation is the *Jack Toolkit*, from Engineering Animation, Inc. The toolkit interface lives in the host software, which encourages user interfaces that are as simple and straightforward as possible. We expect that in the future, other human models will have to adopt this software approach

to live across multiple systems. The alternative is to wed the human model with the CAD system, but this becomes difficult to extend outside the CAD vendor's environment.

In general, human models can potentially aid the TO validation process. Since the primary role of the TO author is to create instructions for real human maintainers, such instructions should be first tested on synthetic maintainers within the given CAD environment. Accordingly, a level of control as close as possible to the actual instruction level will greatly aid the TO creation and validation process. In particular, a user interface that supports the expression of task instructions in natural language (as found in TOs) will reduce the need for the TO author to be both animator and programmer.