Chapter

16

Working in an Electronic Environment

Paul Matthews
Ultrak
Lewisville, TX

Paul Matthews has been practicing mechanical design engineering for the past 12 years. In his 10 years of experience with Texas Instruments, he was part of the design team for the F-117 Stealth Fighter infrared night sight and a major author of the Mechanical Product Development Process for the Defense System and Electronics Group. At TI, he gained a high proficiency at 3-D solid modeling using ProENGINEER and developed several standard best practices for modeling and data management. For the past two years he has been employed as a design mechanical engineer and division director at Ultrak, specializing in the design of larger volume commercial and professional security-related CCTV products.

16.1 Introduction

One question I've dealt with as a mechanical engineer is: "Why generate so many paper drawings and documents to get a product built?" A simple answer to this question is to provide a manufacturer information on how to make the product parts and assemblies. However, a more important and often forgotten reason is to *make a profit for the company that pays me.*

I get paid to design and build a product to sell. In today's environment, if I can't accomplish this faster than my competition, I might as well not do it at all. If I'm really paid to produce a product faster and better than my competition, will I have the time to generate 2-dimensional (2-D) paper documentation to capture the 3-dimensional (3-D) design information and notes referred to in the previous chapters? Will I ever consistently generate a drawing that everyone in the product life cycle interprets the same way? And will this drawing provide the information necessary to build the component? Even if I did, does a manufacturer use this information in a way that helps an improved product move faster to market?

The main reason for writing this chapter is to give you ideas for *capturing and sharing design information to manufacture products with minimal paper movement*. The ideas presented here are not limited to drawing dimensions and tolerances, but include all information associated with the product development process and the data formats used to better support today's rapid product development and production.

16.2 Paperless/Electronic Environment

16.2.1 Definition

I've been in several situations where design programs advertise hours saved by going to a paperless design and manufacturing environment. When asked how they do it, the responses usually indicate that drawings are transferred to the manufacturing facility by modem, e-mail, or LAN-based communications. After the drawings are downloaded, the manufacturing engineers print the files and pass the paper to the next person in the process. This saves numerous hours compared with the hand delivery of the same paper drawing. Yet this does not reflect the true meaning of "Electronic/Paperless Environment" that I want to discuss here. There's more to this environment than the speed in which electronic data can be transferred from point to point.

An electronic environment process has two distinct functions:

- To capture the design and manufacture information in a data format best suited to the person making the decisions for the particular process step.
- To share and reuse the captured information in concurrent engineering for later steps in the process.

For many of the designs done in industry today, this data format is a computer-aided engineering (CAE) database; a 3-D computer aided design (CAD) database, and various other formats for supporting notes. By putting less emphasis on paper documentation and more emphasis on a well-documented concurrent design/manufacture data capture and share process, the cycle time, cost, and quality of new designs is improved.

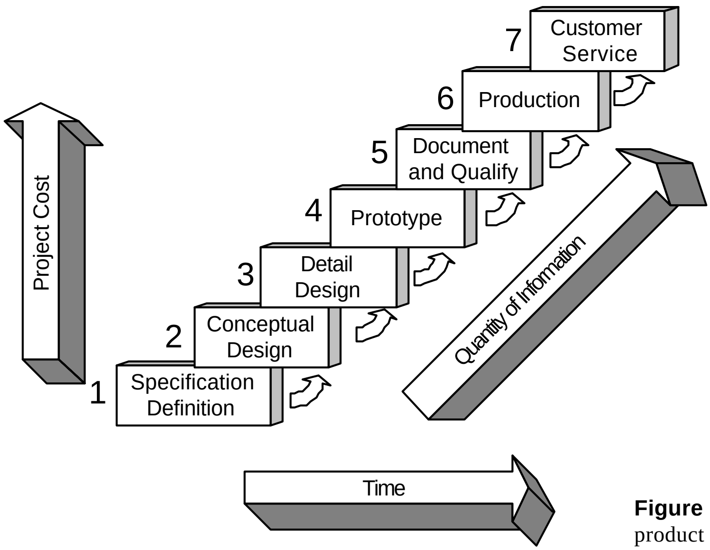

Figure 16-1 Information flow in the product development process

A typical product development process is shown in Fig. 16-1. During the product development process, the quantity of information increases rapidly and each prior process block's information supports the process block above it. The majority of this information is in several types of computer formats and each separate block in the process represents not only a process step, but possibly a different person, department and even company completing the task. It is critical to the process that this information is captured and seamlessly shared from block to block. As seen in the figure, the bigger the information overlap on the blocks, the shorter the time and inherently the increased strength of the product design process.

16.3 Development Information Tools

What we all want to do is make the product development process better. To make the process better, we need to capture and share design and manufacturing information in the most efficient way possible. The most efficient way, for some companies, is to use paper and pencil and many manila folders to navigate information through the development process. For the majority of the competing companies in the marketplace, the computer is used to help guide the information flow.

This section describes several techniques to help the product team with design and manufacturing information in electronic forms.

16.3.1 Product Development Automation Strategy

Electronic automation is a simple concept for most companies today. The best automation is generated from a simple idea put together with other ideas to form a completed tool. It starts with something known and builds on solutions until the requirements are met.

What generates a good automation solution?

- **Product Process Requirements Knowledge**

The product process must be defined. Often companies build automation and then figure out how the process needs to flow to use the automation that was constructed. Inherently, this forces the automation and process to iterate until a common compromise on both automation and process is met. Clearly, successful companies know what information is needed during the product life cycle and what the process needs to be to support the capture and flow of the information. The automation of the information flow becomes very well defined and simple to implement.

- **Automation Experience**

Solid experience is critical. To know when something worked before (or didn't work!) enables automation designers to think ahead and not waste time pursuing paths that will dead end later. A new technology is always alluring to automation designers, but may not be the best solution to the problem. Experience, with not only the latest and greatest technologies, but also the tried and true technologies, will usually generate the best solutions.

- **Process Tool Proficiency**

Tools are meant to help someone complete a task. When a person who generates automation is proficient in the process tool that the automation is designed for, the automation is stronger. The proficient tool user enhances the features in the process tool and does not construct the automation to force the desired outcome. A simple example is a person writing a Visual Basic script to add up a column of numbers in a spreadsheet program. Obviously, the spreadsheet program has built-in functions to do this task and a script would be foolish.

- **Imagination**

Without the ability to solve a problem in many different ways, automation designers can get easily stuck. There is always a way to complete the desired task. If you don't think of the best way to do it, your competitor will. Don't underestimate the importance of this point. Most often, the simple obvious choice is the right choice. In those situations, when the obvious choice does not produce the desired outcome, the automation designer needs to think outside the confines of previous solutions. Here is an example of a problem and a solution.

Process step: During this particular product development process step, a design team member is responsible for providing a marketing team member with a photorender of the new product for marketing literature, such as an advertisement for new company products.

Problem: The new product's 3-D solid model is so complex and has so many features, the photorender software used to automate this process step will not run to completion on the current computer system.

Solution: The automation designer develops the parameters associated with this size of the solid model and flags solid models this size or larger as candidates for Stereolithography and paint. After the scaled model is built and painted, a real picture can be taken.

In this example, the automation designer has the ability to think outside his expertise for a solution to the problem. A more powerful computer helps (by the way, you can never have enough!), but for this particular company, it was not cost justified for the number of products that fell into this category.

- **Automation Flexibility**

No product development process will remain fixed long enough to develop a full set of automation support. Automation that is built to endure modification in the process is very costly and almost impossible. The process must be able to change with the company's growth and expectations. When the process changes, the automation must be updated to support the change without major rework.

- **Support**

Like any tool, automation requires maintenance and repair. Support personnel are required to keep the tool current with the process and also with changing technologies. Automation that is left alone will slowly wilt like a plant without water. The difference is that the plant will show signs of fatigue, where the tool will just stop growing with the process. The first sign of trouble is when the product competitors beat you to market with better designs.

- **Luck**

Luck is a relative word. Anyone who claims they can control product development team expectations, keep key employees from leaving the company, and prevent lightning strikes to the main computer, has had incredible luck in their career. I prefer to anticipate bad luck (even expect it) and always be ready to regroup and attack.

The above concepts together create good process automation. Keep in mind, automation is not the most important point here. The main effort with any automation is to support the process that needs the automation. A tool never dictates what a process should be.

16.3.2 Master Model Theory

As computer software becomes more advanced, it enables the design team to capture more information into a single database. This single database is referred to as the master model. The information captured in this database appears in many forms. Some are listed in Table 16-1.

The master model is the controlling design database, capturing all relevant design data in one central location. The key to the master model concept is to generate the design and manufacturing process based around a focused design data set and use this master set to generate all supporting documents. Once captured, other engineering and manufacturing disciplines reference this information in formats best

Table 16-1 Information captured in a database

Information Type	***Description***
Graphical Data	The nominal geometrical representation of the design.
Graphical Data Attributes	Geometry attributes such as line colors, widths, and visibility.
Dimensional Attributes	Dimension and tolerance attributes associated with the geometry. Dimensional attributes provide the scale of the geometry.
Design Notes	Notes and design calculations used in the product process that may be needed for future revisions of the product.
Parameter Data	Information such as cost, part name, designer name, part number, material, and design revision are a few examples. The number of fields of parameter data can be quite large and provide excellent process automation opportunities.
Software-Generated Parameters	Calculations done by the software using designer parameters and attributes as inputs: mass properties, number of parts in an assembly, and measurement calculations are several possibilities.
Manufacturing Process Data	Manufacturing specifications needed to complete the fabrication of the design. Material finish, packaging/shipping requirements, surface roughness, special tool requirements, and regulatory conformance requirements are examples.

suited for what they need during any particular process step. When the master model is updated, supporting information is updated concurrently, with little interpretation. This update process can be very efficient if automated.

Fig. 16-2 shows a simple example of when the engineer decides to add a screw to an assembly. The most logical place for this to take place is in the CAD model, where he parametrically adds the screw model into the CAD database. The database is considered the master model in this case. Other documents are linked to this master model, and because of this, are directly updated with the new information. The principal point here is that all the other product design disciplines know to look at the master model for

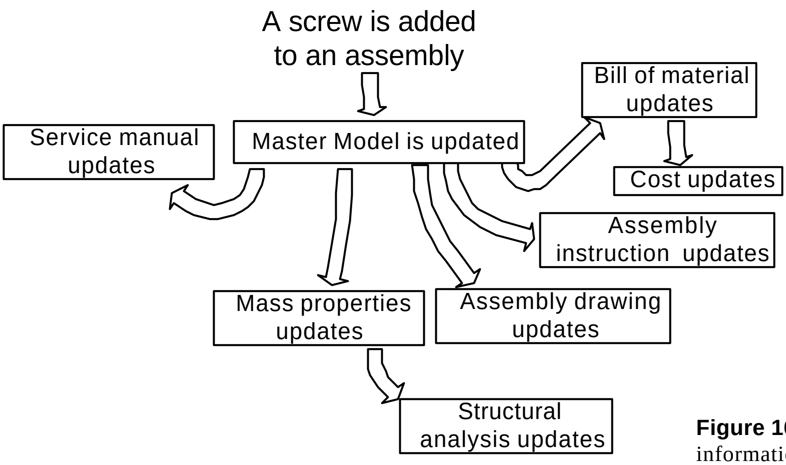

Figure 16-2 Master model process information

changing information. Once again, if this process is automated, very little effort is needed for this change to be cleanly incorporated across the product design group.

There are many examples of how the master model can be used in the product design process.

- Computer Aided Process Planning (CAPP) software for the manufacturing process uses the master model as the seed for generating detailed work-flow estimates and numerical-controlled (NC) code for machining.
- Purchasing may use the master model source as a guide for ordering purchased hardware for the assembly.
- The structural analysis of a part may automatically be recalculated for updated geometry. A document may be autogenerated showing inspection dimensions that fall below a certain process capability of a machining center.
- The tolerance analysis may be directly linked to the solid model CAD database, so that when the tolerance is changed in the model, the analysis is automatically updated.

Theoretically, information is captured one time in a single database file by one software program used by all disciplines of the product development process. In reality, this is unfortunately not the case. A printed circuit board assembly (PCBA) design is a good example. A PCBA will have a mechanical database to specify packaging constraints constructed in one CAD software, electrical schematic data to define the circuit in another CAD software, a circuit board layout for the etch runs, bill of materials in a third software, and possibly simulation data in a fourth. There are also numerous soldering specifications, material specifications, component data sheets and any other referenced document. All of these together capture the design intent for the product. One of the most important pieces to the success of the product process is to know the master model or master data set, and let this single data set control the design automation and reference.

The following is an example of a very common occurrence that illustrates the importance of the master model:

I used ProENGINEER™ solid modeling software to create the design database. It was common practice to take the 3-D solid ProENGINEER™ files and convert them (using a DXF conversion standard) to 2-D AutoCAD® files to generate the drawings. These drawings were taken to the shop where 3-D Computer Vision (CADDS4X) databases were generated to create the NC program. Remember the design database (master model) was ProENGINEER™.

Here are the problems:

- The design was interpreted five times, with each conversion moving farther away from the designer's thought.

  Designer thought → 3-D CAD→2-D Drawing→3-D CAM→ NC Program→Inspection
- When making changes, the change was updated and interpreted in at least four different databases. If the parts were measured with a coordinate measuring machine (CMM), this adds another interpretation.
- Each step in the process may have a different owner, department, or in some cases company involved to complete the process step.

This simple idea can provide a powerful tool for automation and a strong product process information set. Concentrate on the fundamental purpose behind the master model: *Focus all product team members to a common data set.* When the product team can quickly and easily find the needed information in a convenient format, the development process will flow smoothly.

16.3.3 Template Design

The most powerful technique for product development is the ability to quickly reuse information from past experience. In my opinion, 80% of all product design work has been done before, and when a company can capture this history and standardize it to boost new products, the company is successful.

Templates can be generated for everything. A template consists of known information that is formatted in such a way to enable the person using it to supply only minimal bits of new information. The template is complete when all the missing variables are supplied. This concept is critical in the product design process. It not only aids in the capture and format of information, but it tells the user when they are done and can go on to the next task. In the electronic environment, templates are linked to provide easy access and update to the master model.

Template strategy is important. As with any product development tool, the tool or template must directly support specific tasks in the process. Not only does the template need to support the process, it needs to properly link and reuse the information with other templates or tools in the process. Common variable attribute names should be generated and used to ensure the compatibility and consistency between the tools. The following list shows a basic procedure for generation of templates.

1. Define and document the complete product development process.
2. Determine the flow objects needed to complete the process. Flow objects are considered the bits of information passed from one process step to the next, the inputs or deliverables of a particular process step. Think of flow objects as the baton passed to the next runner in a relay race.
3. Generate the list of variable names or parameters needed to efficiently define the flow objects' information.
4. Group the parameters using timing requirements or functional disciplines. As an example, cost, size, and weight goals need to be known at the beginning of product design. Usually, marketing determines these constraints based on customer demands or expectations. The designer uses these goals as requirements during the design of the product and, during the design process, updates the parameters. This group of parameters (cost, size, and weight) begins with a marketing function and flows to the designer for ownership and update.
5. Capture the parameters or attributes in a template format best suited for the person making the decision. Once the parameters are captured, reformatting for reuse into other templates later in the process should not be a problem. The goal is to have the person who makes the decision enter the information only once.
6. Test the process templates. Remember my comment about luck earlier in the chapter. The product development process will change as fast as you generate these templates. Don't focus on designing the perfect process or the perfect set of parameters. Design the process, templates, and all other tools to be flexible to change. The idea is to improve the design process using a consistent means of capturing and communicating information, not to overly constrict or require data that has no positive effect on the design process.

Define→ Determine→ Generate→ Group→ Capture→ Test

16.3.3.1 Template Part and Assembly Databases

There are many feature-based CAD tools on the market today. A feature-based tool allows the user to build the geometry and design requirements by parametrically adding up small mathematical features into the final, sometimes complex database. When using these types of tools, the user does not have to start modeling the design from the first feature. This is not always obvious. However, many times parts and assembly databases have common information based on the classification of the model. By capturing these common elements and putting them in data models, you define templates.

A template part or assembly can be used to capture common information or modeling technique into a *starting database* to jump-start the model. These model databases are declared standard and are used as the base elements of a design. Since these elements are predefined, automation can be easily written to retrieve information needed.

Templates should not be confused with library components. The templates are starting points of a new design, where a library component is a complete configured data set that is not changed during the product development.

Common elements for a template database were shown in Table 16-1. Table 16-2 adds more detailed descriptions and suggestions for these elements.

Table 16-2 Examples of templates

Information Type	***Template Examples***
Graphical Data	Common starting geometry such as a cylinder for a lathe part or a rectangular chunk for a hog-out
Graphical Data Attributes	Defined entity colors and feature or drawing layers. Standard views such as front, back, right, left, top, bottom, and isometric
Dimensional Attributes	Standard dimensional scheme or modeling practice. Defined datum planes for the associated geometry. Standard units such as inch or millimeter. Material values such as density.
Engineering Design Notes	Engineer's name, employee number, computer name, and design location. References to other designs with similar characteristics.
Variable Attribute Data	Part cost, part name, part number, material description, design revision, drawing number, part title/description, revision level, current mass properties, vendor number, and customer number are a few examples. File attributes such as size of database, database location, and last modified date.
Software-Generated Parameters	Mathematical relationships in the database. Formatted mass property reports. Equations that may calculate estimated cost based on parameter information supplied during the design process.
Manufacturing Process Data	Standard material finishes and specifications. Reference to a standard tool list or feature list used for geometry generation. Tolerance limits for process capability calculation. Common raw material or stock parts.

16.3.3.2 Template Features

Similar to template parts and assemblies, common features can be generated and put into libraries to be shared by all. Often there are common feature groups that can be inserted into the model as a set. A common example would be two pinholes for location of a part to a mating part. The holes can have the correct tolerancing and dimension and also reference the correct pins to use in the assembly. Library features can have built-in knowledge parameters to pass on information such as cost of machining operations, process capabilities, NC machine code, tooling list, and design guidelines for using the particular feature. With this information available to the designer, the designer has the immediate ability to know the impact of using the feature before the feature is designed into the product. The designer also does not have to spend any extra time locating information that could easily be supplied as a parameter or attribute.

16.3.3.3 Templates for Analyses

It is very unlikely a designer will do an analysis new to the industry. I must have 30 spreadsheets that I've generated or acquired that perform specific design-related activities ranging from tolerance analysis to trade-off analysis of cost and scheduling of a new product. Once again, a company's success is dependent on the ability to use its resources to generate these common templates and build them into standards. Once standardized, electronic information can be shared between product team members for efficient design and manufacture of products.

16.3.3.4 Templates for Documentation

One of the most common uses for a template is a drawing. As seen in Chapter 4, drawings are made up of various elements put together to define a particular product. For commercial products, there is a limited number of manufacturing processes, materials, and drafting rules to generate product documentation. It is very possible to generate complete documentation directly from a master model with little or no user input. Current Internet and Intranet technologies can generate these pieces of documentation in the background without any designer effort.

Other common document templates used by other product development team members are shown in Table 16-3.

Table 16-3 Common document templates

Engineering Change Notices (Requests, Proposals, etc.)	Assembly Work Instructions
Material Requests	NC Machine Programming
Purchase Requisitions	Service Manuals
Marketing Information	Quality Control
Manufacturing Instructions	Budgets, Schedules

16.3.4 Component Libraries

Component libraries are very powerful resources for the product design team. Not only can the library provide a CAD model; it can include all necessary data associated with the respective library component. All parameters and attributes should be set to reflect all needed information about the component. With this data captured in the component, it is available throughout the development of the product.

When capturing components for libraries, keep in mind the following:

- Geometry must reflect the component as accurately as possible, but not provide so much detail that the application software is overloaded. As an example, an actual helical thread on a solid model of a screw is most likely too detailed.

- Geometry should be modeled at the mean of the manufacturing process. This is usually the center of the tolerance zone. To illustrate: A bearing which may be specified at .437 +.000/-.014 should be modeled at a process mean dimension of .430 ± .007.
- The attribute data must be correct and under configuration control so as not to be inadvertently changed.
- Library components should be controlled from a central distribution area for ease of update and configuration.
- Library components should be verified with any application software revision.

16.3.5 Information Verification

Information is easily entered *incorrectly*. Companies are increasing their dependence on the information captured in complex Master Models to support concurrent product development and manufacturing. The current problem with this dependence is the possible lack of control and verification of this information. Questionable user proficiency in the tools, growing product development processes, and constant change in personnel complicate the standardization, completeness, and integrity of the design data. In turn, the cost and quality of the developed products suffer.

Mechanical solid modeling tools are very powerful. Along with the strength and capability of the tool comes the complexity of the tool use. In my 10+ years of mechanical design using ProENGINEER™, I have seen many models that have grown into complex webs of features. One of the main issues is that the person modeling the design may not recognize the problem. Often, these designs were released for production without any verification to corporate modeling standards. After several weeks, when the design needed to be updated, the complex model was virtually destroyed in the process of update.

All product development data should go through an automated verification process prior to process step acceptance. This information can be used to determine schedule milestones, resource requirements, and verification of clean information flow to the next product development team member.

The following shows a few common examples of corporate standards to verify and document in a solid model to keep consistency in the quality of the databases:

- Adherence to corporate modeling standards
  - 3 Model was started with a common template.
  - 3 Corporate standard-defined features are used.
  - 3 External references to other geometry are controlled.
  - 3 Tolerances are correctly attached to features.
  - 3 Parameter information follows corporate standards.
  - 3 Model name convention follows data management standards.
- Model Completeness
  - 3 Number and type of features reflect completeness of design.
  - 3 Material has been defined.
  - 3 Complexity of model.
  - 3 Proportion of sketch dimensions per feature measures model complexity.
  - 3 Number of parent/children features measures model dependence complexity.
  - 3 Number of mathematical relations in the model shows design-captured information.

  - Family tables or grouping information displays family parts.
  - Regeneration or rebuild time helps determine computer hardware requirements.
  - References to other data forms show relationships to other information.
  - Total database file size helps determine archival requirements.
  - Proportion of physical size of model versus physical volume gives insight into fabrication costs.
- Integrity of model database
  - A regeneration error list helps determine problems in the model.
  - Dimension values less than .01% of the model size can help determine questionable design.
  - Suppressed or hidden features list can determine modeling mistakes.

16.4 Product Information Management

The management and control of the product data is the key to a successful electronic environment.

A paper document is a fairly easy item to keep in revision control and requires very little knowledge to handle. A database, on the other hand, requires knowledge of the database format, knowledge of the software used to extract the required data, and hardware to support the electronic media. Many lawsuits have forced society into legal document frenzy. Okay, maybe I exaggerate a little. But no doubt, having a fully dimensioned, fully toleranced, printed drawing, makes any fabrication shop a little happier. The manufacturer wants to point to a piece of paper and say, “That’s what I built.” The drawing, then, acts as the common interface, the legal binding document, between the designer and the fabricator. There are several main elements to consider about product information management:

- The product team will NOT use an information management tool that inhibits the development process.
- The developing product must be defined well enough to fabricate and verify.
- Product data must be in a format that is supported throughout the life of the product.

There are several ways to manage the configuration of the product documentation. Each of these methods should be used to ensure the electronic data is under configuration control. The Master Model Theory really comes into play in this task. To have only one place to update and control information is much safer than several different places.

16.4.1 Configuration Management Techniques

Configuration and control of information is big business for many companies. There are hundreds of software developers selling their information management products. Each of these tools is designed to support a data management process, as suggested in section 16.3.1. To select the correct tool for the development team, choose the tool that supports the team’s process.

Remember that the best tool for a job is the easiest and simplest to use to get the job done. This may result in no automation tool at all. If the product team understands the importance of data management, less formal control is needed and the data is instinctively controlled. On the other hand, if the team does not understand the importance, the process and associated tools need to be strict and authoritative to assure data is not inadvertently damaged.

16.4.2 Data Management Components

There are a few simple components to a data management philosophy. Fig. 16-3 shows the hierarchy and descriptions of these components.

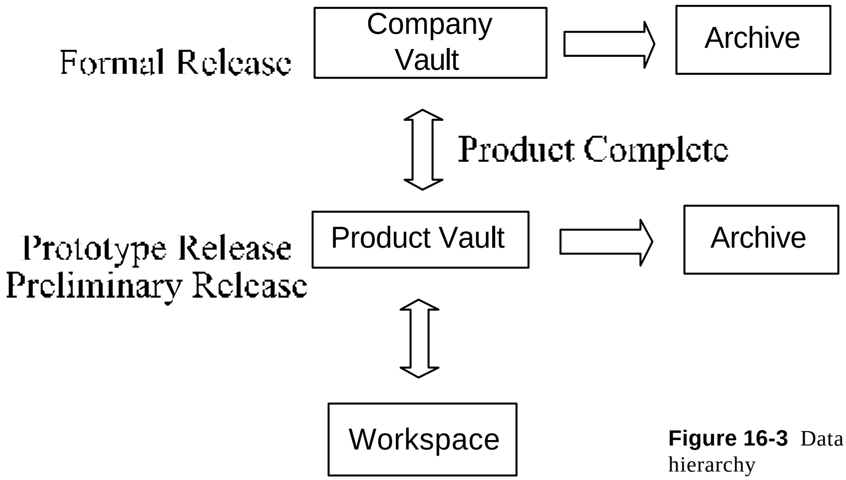

Figure 16-3 Data management hierarchy

16.4.2.1 Workspace

The workspace is the area where the daily development efforts take place. This is where the work is moved for update and the addition of information. Think of this area as the desk where the work is being done. The contributing development team member has full control over the data. They are responsible for all changes to the data and are also responsible for putting the data back at certain levels of completion. There is only one version of the data kept at this level and it is normally not archived or controlled.

16.4.2.2 Product Vault

A product vault is a place where the data is kept and controlled for the product. Multiple revisions may be captured and managed to ensure the product data is current and available to the complete product team. At this level, the data is archived for safety. Release levels may be set to ensure particular revisions, such as the release for a prototype part, are kept, When a particular part of the data is considered complete, it can be put into a preliminary release status to make sure it does not change while it waits for promotion into the company vault. This level may be thought of as a special locked office in the product area where everyone puts their information at the end of the day.

16.4.2.3 Company Vault

Formal release procedures are in place to submit data to the company vault. This level gives the entire company access to the information. Strict change management is in place. This level is archived at the company level to ensure the product data set is not lost or corrupted. The company vault is a crucial component because product development teams may not remain intact after the product is released.

16.4.3 Document Administrator

In any orchestra, there is a conductor. For data management this conductor is the document administrator. The focused effort of this product team member is to manage the data. This is not just a policing effort, or a sign-off block on a print, but a detailed understanding of data that emphasizes wrapping up the data in a consistent package. Verifying the file formats, modeling and documentation standards, release levels and where the data is stored are all responsibilities of the document administrator. This is a perfect application for the information mentioned in section 16.3.5.

16.4.4 File Cabinet Control

One of the simplest, lowest cost and most effective approaches to data management is the concept of file cabinet control. In a paper world, this would equate to (as the title suggests) a file cabinet. Each drawer on the file cabinet can be locked and unlocked by different people on the development team. Each paper folder in the cabinet drawer may represent a different revision of the product. In the computer world, this translates to folder permissions, computer access, and database filenames. Directory levels are set up to match with appropriate permission levels. This method may become cumbersome with larger product teams and higher administration efforts, but is very effective for small and medium product development efforts.

16.4.5 Software Automation

Product Data Management (PDM) software is available in many different levels to support the processes mentioned. The cost and level of detail on these packages range from low, such as a simple program used to copy the data to a different area, to very high, such as a total data management system that supports an entire company worldwide. Remember that no automation at all may be the best solution for the development team. Rely on the product development process to help pick the appropriate automation.

16.5 Information Storage and Transfer

The capture of product information is important, but without the storage and distribution of the information, the process comes to a halt. This section describes some of the most common information storage and distribution methods available. The world is changing fast in this area, and new methods and techniques appear every day. Don't limit the product team by what method has been used in the past and don't forget to support the development process with the methods you choose.

16.5.1 Internet

The World Wide Web (WWW) has grown enormously in the last few years. Many companies have both an Internet (outside the company security) and an Intranet (inside the company security).

The company Internet (outside) usually supports information distribution for the customer of the products. This allows very easy access and distribution of product specification information, troubleshooting tips, costing and sales-related information, software upgrades and patches, and many other customer-related service elements.

The company Intranet (inside) supports information and distribution of information for internal company use. Phone lists, human resources procedures and policies, technical data, and product specific development efforts are just a few examples. The Intranet is internal to the security of the company. Usually a firewall device inhibits outside hacking and provides the necessary security.

Both the Internet and Intranet are powerful with today's electronic information. When generating these systems keep several points in mind.

- Keep the focus of the system on the support of the process.
- Make sure there are support resources after the initial posting of information.
- Advertise where the information is located.
- Allow the structure and organization of the system to change with the process.
- Don't be scared to try new system technology.

16.5.2 Electronic Mail

E-mail has become one of the most used (and abused) forms of information transfer and distribution. Unlike an Intranet, information is pushed to the recipient but not able to be pulled when needed. This electronic communication is incredibly fast and convenient by allowing files to be attached with text and sent around the world in a matter of minutes.

There are several points about the use of e-mail.

- The e-mail you send can be intercepted and read by someone who really wants to get the data.
- E-mail is convenient, quick, and powerful. I sometimes find myself reading 10 to 20 e-mails daily addressed to "GROUP EVERYONE" sharing how someone in a different group may be leaving an hour early from work. Be aware of the groups you are sending the mail to and make sure the data is relevant to that group.
- The data you are sending may not necessarily be archived or kept. e-mail is like a paper letter that may get filed or thrown away.

16.5.3 File Transfer Protocol

Most transfers of files on the Intranet are transferred via FTP. Once connected to the Internet, this protocol allows not only getting data (use the command GET), but also putting data (use the command PUT). There are many software applications that support FTP and make it look and feel like a standard Windows-type program. If an application of this type is not available, a generic FTP program comes with Windows 95 and Windows NT; you guessed it, it's called FTP.

To GET or PUT a file using FTP follow these steps:

1. Logon to Internet
2. At a command prompt type: FTP HOST COMPUTER. The HOST COMPUTER is the FTP server with which you want to communicate.
3. Provide the appropriate login and password. For many servers you can use ANONYMOUS for the user and your e-mail address for the password.
4. Type BINARY. This sets the transfer mode to a binary protocol which will correctly transfer most files.
5. Type STATUS. This gives you status of the transfer.
6. Use GET to get a file from the server, PUT to put something onto the server.
7. EXIT logs off the server. QUIT leaves the FTP program.

16.5.4 Media Transfer

Transferring over the Internet is the fastest way to transfer data around the world. There are many times when a vendor or supplier does not have access to the Internet and another media needs to be used to transfer the information. Here are several media types commonly supported.

- CDROM. Writable CDROMs (WORM - write once read many) are very convenient for media data transfer up to about 650 megabytes. Almost every computer has a CDROM and can read the data. CDROMs are excellent because the data sent won't be accidentally erased or changed. There is a permanent record of what information was sent. Although CDROMs are common, there are different formats for the data. It is necessary to know which CD format is most versatile.
- Tape. There are many tape archive formats available ranging from 400 megabytes to more than 4 gigabytes. Although the tape can hold a lot of data, the data retrieval is cumbersome and slow.
- Floppy Disk. The 3.5 inch floppy is supported everywhere. It will hold up to 1.4 megabytes, is small, and very cost effective.

These different media are all useful, but the most powerful tool used during transfer, both electronic and by shipping media, is the ability to compress the data. There are different data compression algorithms and tools, but the most common are Zip utilities by PKWARE®. It is not uncommon to compress ASCII data formats by 80% as well as adding security encryption at the same time.

16.6 Manufacturing Guidelines

This book is titled as a dimensioning and tolerancing handbook. The chapter so far has delivered suggestions associated with electronic data; how to use it, control it, –and automate it. This section is devoted to providing some guidelines and best practices associated with the mechanical engineering development process, specifically the transfer of information to manufacturing for fabrication.

16.6.1 Manufacturing Trust

The most important aspect of working with a manufacturer and electronic data is *trust*. The customer must trust that the vendor will do their best and the vendor must trust that when they do their best, the customer will be satisfied. More often than not, a manufacturer will require a detailed drawing for inspection of the finished part. They do not necessarily need the drawing, but need the legal document to cover themselves if things do not go as planned. In the following sections, trust is a major element. Some of the new prototyping and manufacturing processes are higher risk to get a better delivery schedule or cost. The higher risk processes are more likely to have problems, and when the problems come up, the manufacturer needs to know he is part of the product team.

Another point to make in this section concerns the inspection methods used by the manufacturer. Although there will be some inspection to stabilize a production process, the movement of manufacturing is to verify processes. What this means is that the tolerances are not inspected if they fall within the manufacturing process capability. Only the tolerances outside the manufacturing process capability are verified and therefore only those tolerances and dimensions need to be relayed to the inspector.

16.6.2 Dimensionless Prints

A common compromise to no printed documentation is a dimensionless print. Basically, views are put onto a drawing format with dimensions and tolerances outside the process capability shown. Specific notes and processes are also captured on the print to allow easy access on the shop floor. This lets the database control the programming and majority of the features, yet allows paper control of inspection, notes, and processes. This also provides a printed document that can be used for better communication between the shop and change control.

CAD/CAM feature-based modeling software is able to capture tolerances associated with feature dimensions. Prior to passing a manufacturing database to NC programming, all dimension tolerances should be set to the mean of the manufacturing process, which is usually the center of the tolerance zone. This will force the geometry to regenerate at its nominal size and therefore the NC program will be written at the mean of the manufacturing process.

There are several standard pieces of information needed on a dimensionless print. These are usually called out in notes or in the title block of the drawing.

- Material. Specify the manufacturing material.
- Finish Processes. Specify processes such as heat treatment and surface finish.
- Manufacturing Process. Specify either the actual manufacturing process (possibly the machining center) or the general tolerance that drives the manufacturing process. A sample note may read, “All features in true profile of .030 relative to datums A (primary), B (secondary), and C (tertiary).”
- Marking Requirement. Specify any particular marking done on the part after finish.
- Design Model. Specify the 3-D model to be used for the geometry. Make sure to include enough information to clearly specify the exact model.

16.6.2.1 Sheetmetal

Many of today’s commercial parts are designed and fabricated using sheetmetal or sheetmetal techniques to deliver the product in a fast, cost-effective manner. One reason sheetmetal has such success is the relatively limited number of machine operations that can be done on it in a production environment.

Sheetmetal comes to the manufacturer as a sheet, as the name suggests, and from there it is cut, punched, formed, and bent. Cutting, punching, and forming are all operations thought of as 2-D operations. The sheet is horizontal and some type of tool strikes the metal, usually at 90 degrees. After the 2-D operations are complete, the flat pattern is bent to the desired shape. More bending processes add more complexity, and make the parts more difficult to manufacture. After bending the material, the process is complete after the finish process and hardware is added.

Table 16-4 Information provided for sheetmetal process

Information Type	***Description***
Provided Documentation	Dimensionless print showing installed hardware
Provided Database	3-D wireframe IGES/DXF format 2-D views of all features IGES/DXF format Unfolded flat pattern with bend lines and bend allowances are shown in IGES/DXF format. Be aware that each manufacture will probably use a different bend allowance, so make sure the one you used is defined for reference.
Prototype Methods	Laser-cut metal flat patterns, cardboard, paper, and scissors
Tooling Needed	Nonstandard punches or forms
Automation Methods	Standard library templates of known punches and process capabilities

16.6.2.2 Injection Molded Plastic

Plastic parts are the most prevalent parts in today's commercial products. After initial tool production and design, plastic injection molded parts are very cost effective and part tolerances can be controlled consistently. In the past, injection-mold tools limited this manufacturing technique to parts with very high production numbers. Techniques are available to use the injection molding process on lower quantity part counts, with drastically reduced tooling costs.

Table 16-5 Information provided for injection molding process

Information Type	***Description***
Provided Documentation	Dimensionless print
Provided Database	3-D solid model native format (preferred) 3-D STL format 3-D IGES surfaced file
Prototype Methods	Stereolithography parts RTV silicone molds generated from SLA patterns Foam and glue
Tooling Needed	High cost production steel or aluminum tooling
Automation Methods	Mold flow-analysis programs

16.6.2.3 Hog-Out Parts

Parts manufactured from chunks of raw material that are cut away into the desired shape are often called hog-outs. Mills, lathes, saws, drills, and many other machines have been designed to cut away material from a piece of raw stock. This type of manufacturing is sometimes time-consuming and often inefficient if the final part does not closely resemble the raw material. The major benefit is that the end item product may not require any tooling or up-front expenditure. This not only saves in up-front cost, but also in lead-time to produce the first samples or prototypes. The process capability of a hog-out can be very good.

Table 16-6 Information provided for hog-out process

Information Type	***Description***
Provided Documentation	Dimensionless print
Provided Database	3-D solid model native format (preferred) 3-D STL format 3-D IGES surfaced file
Prototype Methods	Stereolithography parts RTV silicone molds generated from SLA patterns Foam and glue Fast turnaround time of Investment Cast prototypes is possible using a Stereolithography QUICKCAST part as the casting pattern. Limited quantity prototypes from steel, aluminum, and assorted other metals can be fabricated at relatively low cost
Tooling Needed	Tooling required dependent on casting process
Automation Methods	Standard library templates of known process capabilities

16.6.2.4 Castings

Castings are an excellent way to produce metallic parts with minimal secondary machining. By casting the near net shape with machine stock on secondary machined surfaces, the time for machining is greatly reduced. The cutting machine needs to only clean up the features whose tolerance is greater than the casting process.

Table 16-7 Information provided for casting process

Information Type	*Description*
Provided Documentation	Dimensionless print
Provided Database	3-D solid model native format (preferred) 3-D IGES surfaced file
Prototype Methods	Stereolithography parts RTV silicone molds generated from SLA patterns Foam and glue
Tooling Needed	Very little special tooling needed
Automation Methods	Standard library templates of known process capabilities

16.6.2.5 Rapid Prototypes

There are many different prototyping processes for mechanical parts. The most versatile and affordable is the Stereolithography (SLA) process. This process can generate an epoxy resin pattern directly off the solid model usually in a matter of days and can also be used to generate molds for rapid tooling for multiple parts.

The methodology for creating a SLA is simple and the hardware for the growing of the prototypes is becoming more affordable. A simple description of the process follows.

Step 1. A solid computer database is sliced up into cross sections.

Step 2. Starting at the base of a model on a platform, a laser sweeps out the cross section on a pool of resin. When the laser strikes the resin it solidifies.

Step 3. The platform is lowered very little and another cross section is swept.

Step 4. The process continues until the part has been grown.

Step 5. The part is removed from the vat of resin and chemically cleaned.

Step 6. The prototype is sanded to remove any ridges.

There are a few things to keep in mind when using the SLA process for models, patterns, and tooling.

- The process capability of the machines is fairly good, (+/- .005) but the parts may dimensionally move over time. Keeping the parts cool will help. Transporting the prototypes in your trunk in the middle of summer is not a good idea. I know this lesson first hand.
- There is usually handwork needed to clean up the model. The quality of this personal touch will vary with manufacturer.
- Some epoxy resin prototype material becomes brittle with age. Care must be taken not to crack the models during handling.
- For rapid tooling, account for any shrink in the molding material in the solid model of the pattern.

Table 16-8 Information provided for prototyping process

Information Type	***Description***
Provided Documentation	Dimensionless print
Provided Database	3-D STL format
Prototype Methods	N/A
Tooling Needed	N/A
Automation Methods	N/A

16.7 Database Format Standards

The information generated about a product during its design, manufacture, use, maintenance, and disposal is used for many purposes during its life cycle. The use may involve many computer systems, including some that may be located in different organizations. To support such uses, organizations need to represent their product information in a common computer-readable form that is required to remain complete and consistent when exchanged among different computer systems.

There are many different types of electronic databases used in today's product development process. This sometimes causes a barrier to sharing information efficiently. When configuring templates, CAD data sharing and any other product development tool, be aware of the data formats used.

16.7.1 Native Database

A native database is considered the database generated by the computer program used by the person inputting the information. For Microsoft Word, the file has an extension .DOC and it is the default format in which the software saves the file. When a Master model uses its native database type, it is most powerful due to absence of anything lost during a conversion to another format. That is why it is critical to pick product development tools that support common database file types.

One of the problems with native database formats is the lack of control from software revision to revision. The data format will usually change with the revision of the software, making backward database compatibility an issue. A native format is also generally saved in a proprietary binary file, making it difficult to extract data file information from outside the native software. Most all common formats (IGES, DXF, STEP) save the data in a clearly documented ASCII file, allowing the data in the file to be used by any third-party software.

16.7.2 2-D Formats

These formats are supported by most popular software when needing to import or export 2-D wireframe graphics.

16.7.2.1 Data eXchange Format (DXF)

Data eXchange Format (DXF) is the external format for AutoCAD®. It is a text-based representation of a 2-D drawing database. A DXF file can contain 2-D geometry, dimensions, drawing cosmetics, and entity layers. The DXF format is usually stable between different releases of AutoCAD®, although items are added to the specification as new entities are added to AutoCAD®. Most all vector software, both CAD software and Microsoft Office products, strongly support the DXF format. Whenever a drawing or line drawings need to be converted to a vector format for another application, a DXF file is most likely to satisfy everyone involved.

16.7.2.2 Hewlett-Packard Graphics Language (HPGL)

The Hewlett-Packard Graphics Language (HPGL) was developed over a number of years by the Hewlett-Packard Corporation for use in their line of plotters. HPGL has become a standard for plotter formats and is supported by almost all plotter manufacturers as a standard emulation. Most CAD systems have the capability of outputting the 2-D format, but very few have the ability to input the format. The HPGL format can be read by some Microsoft office products and seems to be a clean way to import 2-D geometry into programs such as Word.

16.8 3-D Formats

3-D wireframe/surfacing and solid modeling software support these 3-D conversion formats.

16.8.1 Initial Graphics Exchange Specification (IGES)

Initial Graphics Exchange Specification (IGES) data format is considered a neutral file scheme for CAD data. Most vector graphic programs can convert to and from this neutral file. The IGES standard supports not only vector information, but also 3-D b-spline surfaces. Using a neutral file format decreases the total number of translators needed and provides a file format transferable to virtually any 2-D or 3-D CAD platform. Appendix A shows a listing of popular IGES entities.

16.8.2 STandard for the Exchange of Product (STEP)

STEP is the ISO STandard for the Exchange of Product data. The STEP format is evolving to cover the whole Product Life Cycle for data sharing, storage, and exchange. This format supports wireframe, surfaced and solid geometry. Current CAD/CAE exchange standards like IGES, DXF, SET, and VDAFS will be replaced by STEP, as well as allow for complete descriptions in electronic form of all data related to product manufacture. STEP is open and extensible and will meet design and manufacturing needs well into the next century.

The STEP format is defined in publications produced by the US Product Data Association (US PRO) IGES/PDES Organization. The complete set of specifications for STEP is referred to as ISO 10303. This is an international standard.

The STEP format is organized as a series of documents with each part published separately. Application Protocols (APs) that reference generic parts of ISO 10303 are produced to meet specific data exchange needs required for a particular application. AP203, Configuration Controlled Three-Dimensional Designs of Mechanical Parts and Assemblies, is an International Standard (IS) version.

Products supporting STEP can implement this interface using different levels of data transfer. Each level provides various mechanisms to store, accept, and pass product definition data between heterogeneous systems in a consistent and standardized way.

16.8.3 Virtual Reality Modeling Language (VRML)

Similar to HTML, Virtual Reality Modeling Language (VRML) has emerged as a standard database structure for viewing solid shaded geometry on the Internet. The user can see the shaded geometry, and navigate around and through the shaded geometry. As with HTML, current releases of solid modeling CAD software support this standard.

16.8.4 STereoLithography (STL)

STereoLithography interface format (STL) was generated by 3-D Systems, the designers of Stereolithography Apparatus (SLA), to provide an unambiguous description of a solid part that could be interpreted by the SLA's software. The STL file is a "tessellated surface file" in which geometry is described by triangle shapes laid onto the geometry's surface. Associated with each triangle is a surface normal that is pointed away from the body of the part. This format could be described as being similar to a finite analysis model. When creating an STL file, care must be taken to generate the file with sufficient density so that the facets do not affect the quality of the part built by the SLA. The SLA file holds geometry information only and is used only in the interpretation of the part.

STL files represent the surfaces of a solid model as groups of small polygons. The system writes these polygons to an ASCII text or binary file. Fig. 16-4 shows the file format for an STL file.

```
solid Part1
  facet normal 0.000000e+000 0.000000e+000 1.000000e+000
    outer loop
      vertex 1.875540e-001 2.619040e-001 4.146040e-001
      vertex 1.875540e-001 2.319040e-001 4.146040e-001
      vertex 2.175540e-001 2.619040e-001 4.146040e-001
    endloop
  endfacet
endsolid
```

Figure 16-4 File format for one triangle in an STL file

16.9 General Information Formats

The formats in this section are not specifically designed to support CAD information. These formats are best suited for document templates, product database interrogations, and general distribution of text and pictures.

16.9.1 Hypertext Markup Language (HTML)

HyperText Markup Language (HTML) operates as a database designed for the World Wide Web. HTML code is a basic text file with formatting codes imbedded into the text. These formatting codes are read by specific client software and acted upon to format the text. Most everyone has had experience with HTML and its capabilities. What makes HTML very useful is the power of not being machine specific. Many documents and pictures can be linked on different machines, in different offices, even in different countries, and still appear as if they are all in one place. This virtual Master Model follows the general rules of the Master Model Theory, yet allows multiple areas for the data to be stored.

Current releases of several CAD programs are supporting the product development process as follows:

- Showing the product design on the web as it matures
- Allowing the simple capture of design information
- Having other support groups "look in" without interrupting the design flow

16.9.2 Portable Document Format (PDF)

Portable Document Format (PDF) is an electronic distribution format for documents. The PDF format is good because it keeps the document you are distributing in a format that looks almost exactly like the original. For distributing corporate standards, this format is nice because it can be configured to allow or disallow modifications and printing, as well as other security features. PDF files are compact, cross platform and can be viewed by anyone with a free Adobe® Acrobat Reader. This format and accompanying browser supports zooming in on text as well as page-specific indexing and printing.

16.10 Graphics Formats

These formats are used to support color graphics needed for silkscreen artwork, labels, and other graphic-intensive design activities. The formats may also be used to capture photographic information.

16.10.1 Encapsulated PostScript (EPS)

EPS stands for Encapsulated PostScript. PostScript was originally designed only for sending to a printer, but PostScript's ability to scale and translate makes it possible to embed pieces of PostScript and place them where you want on the page. These pieces of the file are usually EPS files. The file format is ASCII-text based, and can be edited with knowledge of the format.

Encapsulated PostScript files are supported by many graphics programs and also supported across different computing platforms. This format keeps the font references associated with the graphics. When transferring this file format to other programs, it is important to make sure they support the necessary fonts. The format also keeps the references to text and line objects. This allows editing of the objects by other supporting graphics programs.

This is a common file format when transferring graphic artwork for decals and labels to a vendor.

16.10.2 Joint Photographic Experts Group (JPEG)

The Joint Photographic Experts Group (JPEG) format is a standardized image compression mechanism used for digital photographic compression. The Joint Photographic Experts Group was the original committee that wrote the standard.

JPEG is designed for compressing either full-color or gray-scale images of natural, real-world scenes. It works well on photographs, naturalistic artwork, and similar material, but not so well on lettering, simple cartoons, or line drawings. When saving the JPEG file, the compression parameters can be adjusted to achieve the desired finished quality.

This is a common binary format for World Wide Web distribution and most web browsers support the viewing of the file. I use this format very often when I e-mail digital photographs of components to show my overseas vendors.

16.10.3 Tagged Image File Format (TIFF)

TIFF is a tag-based binary image file format that is designed to promote the interchange of digital image data. It is a standard for desktop images and is supported by all major imaging hardware and software developers. This nonproprietary industry standard for data communication has been implemented by most desktop publishing applications.

The format does not save any object information such as fonts or lines. It is strictly graphics data. This allows transfer to any other software with minimal risk of graphic data compatibility. This is a very common format for sending graphic data to vendors for the generation of labels and decals.

16.11 Conclusion

Some of the many techniques for electronic automation, information management, and manufacturing guidelines are presented in this chapter. This small sample has given you more tools to use in successful product development. The chapter also provides two main points to keep in mind in future projects:

Engineering and manufacturing data are critical components in the development process and need to be strategically planned. Computers and electronic data can offer huge possibilities for rapid development, but process success relies on understanding not only *what* can be done but also *why* it is done.

The age of the paper document is not gone yet, but successful corporations in the coming years will rely completely on *capturing and sharing design information to manufacture products with minimal paper movement*.

16.12 Appendix A IGES Entities

IGES Color Codes

IGES Code	Color
8	White
5	Yellow
2,6	Red
4,7	Blue

IGES Entity

Type	Name	Form
100	Circular Arc	
106	Copius Data	11-Polylines 31-Section 40-Witness Line 63-Simple Closed Planar Curve
108	Clipping Planes	
110	Line	
116	Point	
124	Transformation Matrix	
202	Angular Dimension	
206	Diameter Dimension	
210	General Label	
212	General Note	
214	Leader (Arrow)	
216	Linear Dimension	
218	Ordinate Dimension	
222	Radius Dimension	
228	General Symbol	
230	Sectioned Area	
304	Line Font Definition	
314	Color Definition	
404	Drawing	
406	Property Entity	15-Name 16-Drawing Size 17-Drawing Units
410	View Entities	