



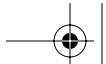
# 3 The User Perspective



## *In this chapter...*

- Introduction
- Four Key EDA User Decisions
- How to Buy EDA Tools—Five Key Issues
- Standards Efforts—Who, What, and Why
- Personnel—The Key to EDA Support
- University Connections
- Summary
- Quick Quiz





## INTRODUCTION.....

This chapter gives the EDA tool users' perspective on the tools and the industry. Who are the users? They are the electronic systems, semiconductor, or ASIC development engineers who design ICs. They are also the engineering design and EDA administration and support managers.

New EDA tool customers need to be aware of the significant costs and risks involved. We will consider tools, tool integration, computer choices, and staff.

Since the IC technology shrinks so rapidly, EDA tools are continually updated or replaced to handle more complexity and new design issues. Tools are normally rushed to market with increased capacity for more complex chips, speedups, and new capabilities.

Users need to expect some frustration in evaluating tools, tool performance, vendor support, in-house support, and documentation. As one might expect, users often find problems which the tool designers never imagined. Everyone is learning at the same time.

Nora's supervisor Sanjay has sent her to Hugo, an EDA tool user manager at Fabless Design, Inc. Nora will be working on his account for Sandbox. Let's listen in as Hugo shares his experience with Nora.

*Nora:* Hugo, thanks for taking the time to explain what EDA users need. What kind of problems do you deal with?

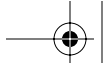
## FOUR KEY EDA USER DECISIONS .....

*Hugo:* Nora, we first need to clarify some issues before we talk about tools. An IC design manager or the EDA manager has to deal with four main questions. (I act in both roles here, by the way, as a design manager and as an EDA manager.) Those questions are:

1. What is the design organization (who and where)?
2. What kind of network capability will you need?
3. What are the security requirements you must meet?
4. What kind of computer systems do you need?

I listed them in this order because each depends somewhat on the previous answers. The tools used are affected by the answers to these questions. Let's take them one by one.





## Organization

1. First, look at the organization. Will one or two people or a group do the design work? Will they all be at the same location or in different remote places? For example, we have a design group in Ireland.

Running design jobs remotely is common on private networks or on the public Internet. This has been an area of tool development for many years. (Some EDA companies provide network support utilities.) *Distributed design* allows you to partition or share the work between design groups separated by time or geography.

Will you do the design work at different sites on the same “campus” (metro)? Perhaps you will do it in different time zones or at different geographic sites (same or different countries)?

**Hugo:** (chuckling) I remember trying to set up a video-conference time common to Japan, the US, and the UK. Someone was always inconvenienced.

### Did You Know?

**Coordinating design groups working around a country or around the world is often very difficult. Just scheduling a time for a video conference or a conference call is complicated. For instance:**

**At 9 A.M. in San Jose, people are just starting work.**

**It is 5 P.M. in London; people are ready to leave work.**

**It is 2 A.M. the next day in Tokyo!**

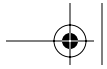
However, the massive size of many EDA files is a deterrent to the rapid transfer between remote sites. Since EDA files contain proprietary chip designs, protecting them is a major concern, particularly over the Internet. For example, we have multi-gigabyte files that can take hours to send or even longer, if there is a network failure and we have to resend them.

File sharing also requires identical or compatible EDA tools at the local and the remote site. So you also need consistent design, change control, and backup procedures.

However, the biggest challenge is the cultural and personnel issues, even within the same company. For example, I once had a design group in Japan working with one in the US. Just following the same checking procedures and agreeing on deadlines was a real problem.

Will the various users doing separate parts of the design be using different software or computer systems? (They may have to be the same, and the same revision.)





If you can answer these questions, then you can determine the sort of computer network you need.

## Computer Network

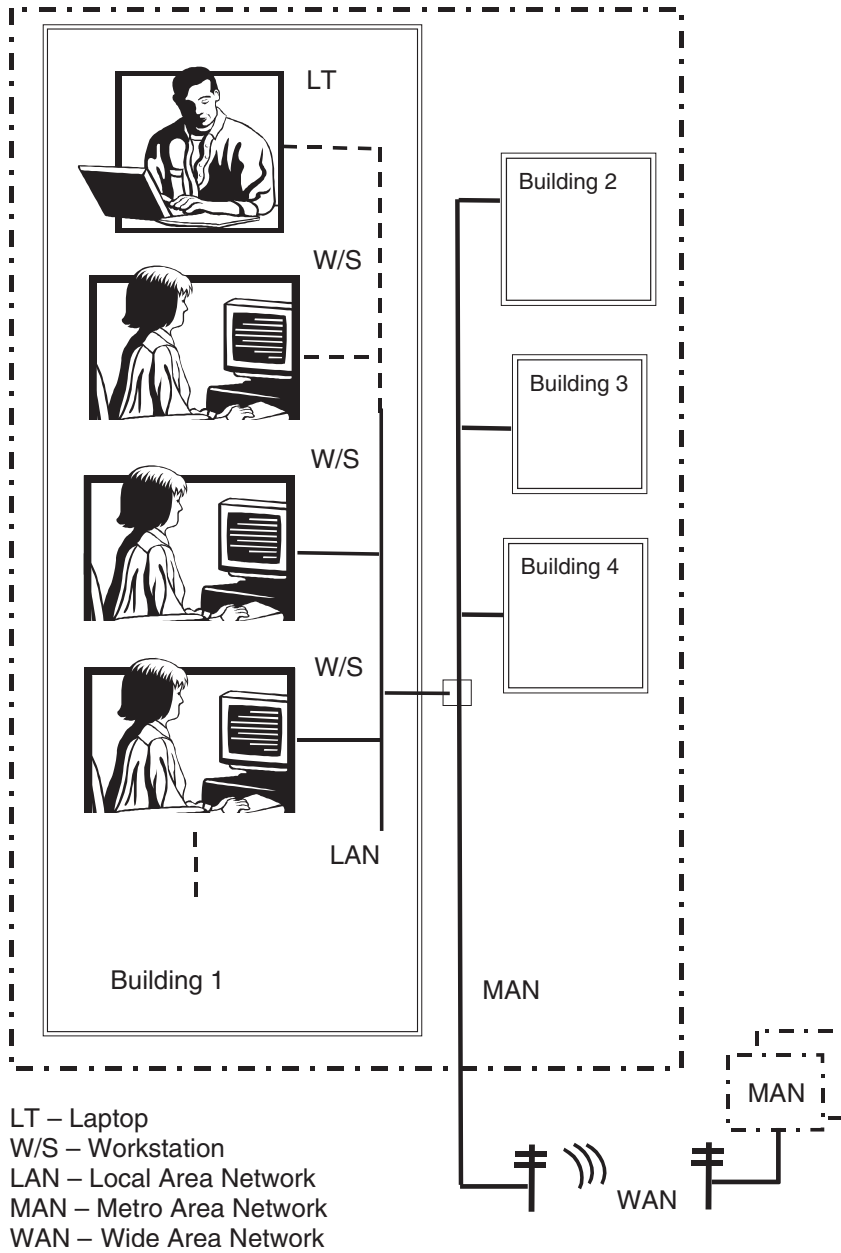
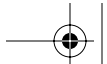
2. Second, look at the structure of the computer network. What does the organization already have, and what does the engineering staff need? Most design engineers like to run or control the tools from a workstation at their desk. (This is important for interactive work. For simulation runs, the tool will run faster on a shared *server* computer or a group of shared computers (*server farm*) with massive amounts of memory.)

If there is a design group, you may want to share a local engineering server. Is a company-wide shared server on-site? Are you required to interface with it? Do you need a *local area network*? Is there one in place, and is it fast enough?

Is there a geographically remote server, involving high- (or not so high-) speed communications lines? Is a connection to the Internet required for remote communications? (Some countries do not allow their residents open access to the Internet...)

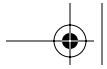
If you can define the computer network you have or need, then you can address the security questions. Let me draw you a quick sketch of computer networks. (See Figure 3.1.)





**Figure 3.1**  
Computer Networks





Note that a *local area network (LAN)* links all the workstations in one building. A *metro area network (MAN)* can link several nearby buildings (as on a campus). The LAN or MAN can connect to a *wide area network (WAN)* to reach worldwide. The Internet is a public WAN. Some companies have private WANs.

## Security Requirements

3. Third, what security requirements are required and are available? The crown jewels of the company will be on your computers. Your engineers need to work with unencrypted files. Industrial theft of *intellectual property (IP)* is quite common, unfortunately. It is not even a crime in some countries. (We mentioned the Cadence/Avant! lawsuit previously. IP includes design ideas about chips, tools, or anything.) Company attitudes towards security vary from clueless to paranoid. Some security examples in use are:

*Firewalls*—these restrict who and what programs have access to and from the network.

Dial-in *crypto tokens*—these have a user password and electronic token code that changes every few minutes. The codes must match the codes at the host computer.

Frequent equipment location checks—location and content checkups of desk machines. Check-in and check-out of laptops.

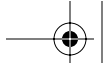
### Did You Know?

**Laptops require extra security. Travelers can easily lose them or have them stolen. The very portability that makes them so handy also means that they can vanish in a second.**

*Password control*—this should be used on project files. The files should also be encrypted when they are not in actual design use. This is a real problem if a tool vendor needs your proprietary design files to resolve a tool bug. There needs to be *non-disclosure* terms in the tool contract (they promise not to reveal any sensitive user/customer information). The vendor also needs to have close control of who can access your design files at his shop. The vendor must ensure that the design files are not put on a networked machine with open access to non-authorized people.

The security environment decisions may affect the computer environment choices, which is the next issue.





**Did You Know?**

**Whatever procedures, encryption, firewalls, etc., are used to deter external access, the largest source (80%) of IP theft is your internal people!**

## Computer Systems

4. Fourth, what computer environment will be needed? All the previous questions affect this decision. On what computers do the current or expected EDA tools need to run (PCs, workstations, servers, or a group of servers)? Most large companies use farms of dedicated multiprocessor servers with huge amounts of memory and disk storage.

On which operating system will the EDA tools run? Options include UNIX (most common), Linux (increasing), Microsoft NT (a few), or Windows (very few). There are different supported versions of each of these. The version differences may be minor, but may require compatible versions of EDA tools which share data.

A few small EDA tools can run on Personal Computers (PCs). However, EDA users usually need high-resolution graphics, large screens, and fast processors. Most EDA tools require as much computer memory and disk space as you can get. They need long dedicated (uninterrupted) run times. As you know, there is great benefit from getting a product to market early. So, the designers should have access to the fastest possible computers.

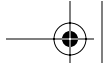
**Did You Know?**

**Some vendors offer accelerators (both hardware and software) that make the tools run faster.**

Changing the computer environment is always a risky experience, with unforeseen delays. So, the managers need a flexible migration plan (and fallback plan!) to include upgrades, company growth (or reduction), and remote sites.

**Nora:** There certainly are many concerns to take into account. I was thinking it was just a computer decision. In marketing, I just use a laptop. What do your designers use?





## Engineering / Non-engineering Goals

**Hugo:** You've touched a sensitive area. There is often a conflict between the company *Information Technology (IT)* and the engineering departments. The IT department seeks shared use of company central computers to lower costs, while the engineering department usually needs its own dedicated machines.

Conflicts arise over computer purchases (e.g., central vs. department servers, workstations vs. personal computers). Operating systems (NT vs. UNIX vs. Linux, etc.), and company-wide standards are also issues. The biggest argument may be over ownership / administration (i.e., company IT vs. local engineering).

Minimizing costs and support staff is the IT department's (in-house or contract) normal goal. Standardizing maintenance and centralizing all business applications are two ways to achieve this. High utilization of their computers (90% or more) is another.

IT business applications typically use small files (a few megabytes) and have short (seconds or minutes) transaction-oriented run times. User response time is not a major issue. User downtime and lockouts for maintenance are normal side effects. So are disk storage limitations. However, these are only annoyances for most of the organization.

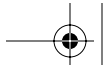
In contrast, EDA tools operate with gigabyte files and long run times (hours and days!). Engineers make a small design change and re-run a whole suite of programs. Thus, long run times are the norm, not the exception. The engineering users need all the computing power and disk speed they can get. Moreover, a hangup or disruption is very costly to their project.

Most large workstations require administration and support (setting up and managing file storage, backup, network connections, etc.). These are usually network administration jobs, not IC designer tasks. So EDA users need IT support staff. They also need EDA support staff to manage the expensive tool licenses, upgrades, and usage.

EDA users need to **reduce** computer loading (utilization) to gain speed for their compute-hungry EDA tools. To meet design deadlines, engineering management must control computer priority and access to the tools and files.

Companies often address these conflicts with separate local computers, administration, and support for engineering. The rest of the company





uses a shared IT server and administration. In companies with multiple projects or engineering departments, the same issues occur among the engineering groups. The engineering managers must share the high-powered engineering server farm. (I have seen these issues in several large companies.)

*Nora:* That solution must be difficult to sell.

*Hugo:* Yes, but it is essential if the organization wants to get the revenue benefit from the EDA tools.

**Did You Know?**

**EDA workstations should always be fully equipped. The cost of adding memory, disk space, and high-speed network links is usually negligible compared to the costs of engineering time and TTM.**

*Nora:* How do you go about buying EDA tools?

## HOW TO BUY EDA TOOLS—FIVE KEY ISSUES .....

*Hugo:* First, you have to know the IC architecture that will be used—custom, standard cell, FPGA, etc. That narrows down the tool choices. Then there are five key issues to consider when buying tools. They are cost/performance, training/support, make or buy, compatibility, and transition. Let's look at them one at a time.

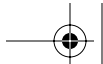
### Cost/Performance

*Hugo:* How will the tool perform compared to similar tools? Tool demonstrations at the vendor's site, at our site, or at a conference are worthwhile. They can answer a lot of questions and raise some more. Of course, vendor demonstrations always show off the tool well (no surprise). They usually are done using small examples or to show just a few features.

Demonstrations also help to show up features from one vendor that are missing from the other. It's also good to ask each vendor why they are better than the competition.

Then you have to decide which features are most important to you. There are often several tools for a specific problem area, such as power estimation. Therefore, comparing the tools can take a lot of homework and engineering time.





Following up references is another way to gain insight on the tool performance. User groups are independent reference sources. For example, on the *Synopsys User Group (SNUG)* website, users describe tool problems and solutions. These include both Synopsys tools and others as well.

Unfortunately, some tool license agreements restrict dissemination of benchmark results. Vendors claim that results taken out of context do not provide a valid measurement. Therefore, users need to get comparative information informally.

What is the total cost for the tool? (Prices have varied from free university tools to over \$750,000.) Tools may be bought outright, or licensed in several different ways. To compare different vendor tools you need to look at the total costs and the ROI to the user company.

What are the tool *licensing* options? These may be on a per engineering “seat” or per use basis or a floating license (any “seat” can use the tool). They also vary on a per month, quarter, year, or perpetual basis, and may be site-specific, company-wide, or worldwide. Table 3.1 shows some licensing options.

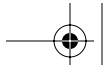
**Table 3.1** Licensing Options

FACTORS	POTENTIAL OPTIONS
LOCATION	Department, site, multisite, division, city, country, worldwide
WHO	Named person, fixed seat (machine), floating seat, # of seats, employee, consultant, subsidiary, customer, etc.
DURATION	Free trial period, month, year, perpetual, subscription (quarterly)
PORTABILITY	Fixed seat, registered persons, dynamic person assignment, anywhere, anytime
CHARGE BY	Seat, minute, hour, monthly, yearly, usage, tool run, project, location, etc.

**Hugo:** Different vendors offer different variations—and new options emerge all the time. The table gives some examples of the variables possible. The costs may vary widely. For instance, a perpetual license may cost four to five times more than a subscription license.

A design manager needs to have as much flexibility as possible. Workstations fail or become obsolete, people come and go in projects and companies. Design managers need to be able to assign a temporary license to anyone they choose, as required. This may be





anywhere, on any project, for whatever time they need. EDA licensing agreements can be a substantial burden for design managers.

The user should get a free trial period and automatic warnings **before** a license expires. The design manager needs easy license renewal, without either user or vendor bureaucratic delays. There have been cases where a license expired at a critical time without warning and took days to renew.

Some vendors charge for the right to use the tool, whether or not you are actually using it. It is obviously preferable to be charged only when the tool is actually being used.

The total cost involves the initial cost, the technical support cost, bug fixes, and upgrades. Yearly support cost often runs 15% or more of the initial cost. (All these things are usually negotiable...) In addition, there is your own in-house support cost to install upgrades and check bug fixes.

What is the cost/performance lifetime ROI? Will this tool create a better ratio of revenue improvement than a similar tool?

**Nora:** I see you need to take into account many cost factors.

**Hugo:** Yes, these are expensive tools, and there are a lot of them. If they make basic cost/performance sense, then we look at the support issues.

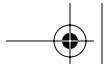
**Nora:** What kind of support issues?

## Training and Support

**Hugo:** EDA tools are complex, with many options and parameters to set or enter. There is a huge amount of data to be entered in specific (often proprietary) data formats. *Graphical User Interfaces (GUIs)* help somewhat, but they differ from tool to tool.

Therefore, engineers need training on how to use the EDA tools. There is a substantial learning curve to become skilled in using each tool. Furthermore, each (frequent) upgrade of the tool may require a refresher course.





### **Did You Know?**

**EDA tools involve a constant learning process. Users often have to learn to use new tools or new features. Training time is a significant burden (losing a designer for a week or so) for any size company. Training can be quite expensive. Nevertheless, companies cannot afford to fall behind, and they need to train multiple people on each tool, to avoid dependency upon a single person.**

Therefore, an important question is: what training is available? Are both initial and follow-on training included? Is emergency (24-hour hotline telephone and/or on-line) service provided? How good is the documentation? Program documentation is often minimal, particularly early in the life of an EDA tool. The vendor is focused on getting the tool up and working, so the documentation usually comes later.

How long will the vendor company be around to support the tool? Is there a backup provision to get the source code and documentation if the vendor support ends? EDA companies may fail or be acquired or merged. Their tools may be discontinued after such an event. Users then have to provide their own support or cope with a new tool and learning curve.

Will you get the tool source code (so you can modify it yourself if need be)? Perhaps you will get the tool object code (cannot modify it), or license the tool. (You normally don't want to touch source code if you are getting support and licenses from the vendor. This becomes more of an issue with an unsupported or poorly supported tool.)

Is there a schedule for updates and releases from the vendor? How and when will change control be implemented in your shop? Who will do it—an engineer or an administrator? When will it be updated—at regular intervals or at specific points in the design?

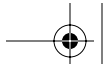
**Nora:** It sounds like a long-term relationship is necessary with the tool vendors. Do you buy all your EDA tools from vendors?

## **Make or Buy**

**Hugo:** Yes, we use only vendor tools. It's usually the easiest and fastest way to go. Most companies use vendor tools. However, some companies develop their own tools when they are not available from a vendor.

“Make or buy” is another important EDA user decision. If done in-house, the company must commit to long-term support. Just as with





vendor tools, internal customers want more support and features for the in-house tools.

We also buy tools to help with essential tasks that no one likes to do. These include design documentation, user manuals, and change notices.

**Nora:** Is there any problem with compatibility between in-house and vendor tools?

## Compatibility

**Hugo:** Yes, there is a big issue of compatibility between vendors tools and between vendor and in-house tools. How compatible is the tool with our existing suite of tools? (Who will make the interface translators if there is a compatibility issue?)

Will there be *backward compatibility*? (Will upgrades of the tool be able to run old versions of the design files?) This can be key if revisions to a product design are made (usually the case). Some managers resist installing tool upgrades to avoid this kind of problem.

On the other hand, vendors will not fix bugs on two-year-old versions (no surprise). And managers cannot easily move people from one project to another if each group uses a different tool version. So companies must upgrade versions frequently, and all users together! This is often a major complication.

**Nora:** I thought that was a problem only when they upgrade the operating system. You mentioned earlier that the tools have to match the operating system version.

**Hugo:** These are issues with both tools and operating systems. In addition, improvements in the tool may often be incompatible with prior design work. Developers tend to expect you to use the new tools only on new designs. You must keep old tool versions available until you are certain they are no longer needed.

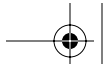
**Nora:** The transition to using a new tool or upgrade sounds tricky.

## Transition

**Hugo:** Well, you need to be sure that an engineering group/project is willing to be the “first user” for a new version of a tool or operating system.

I mentioned earlier the update schedules from the vendors. What is their timetable for bug fixes and/or upgrades for a new tool? (A new



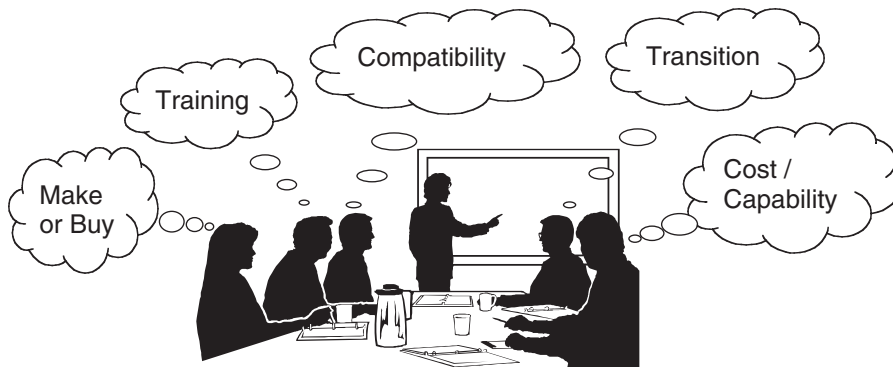


EDA product will need more productization to get it usable for a real design.) When will it be ready for your first user and will the vendor give extra support?

Is there a Trial Period and Escape clause (i.e., if you try it and don't like it, can you get a refund)? The vendor is unlikely to negotiate a low price with such a clause. But you need an agreed-upon *acceptance test* as a criteria for acceptance or rejection.

The internal staff also serves as user gurus for the design engineers for all the EDA. If new tools are bought, will you need more in-house staff to support them? More staff can be a budget issue unless the EDA tools are shown to directly support revenue-generating projects.

**Hugo:** So those are the five essential issues in buying EDA tools. I hope you are not overwhelmed. See the little picture I have over here that summarizes the negotiation process. Everyone has his or her own focus area, but we have to cover all the concerns. (See Figure 3.2.)



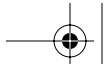
**Figure 3.2**  
Buying EDA Tools

**Nora:** That's a good summary. There sure are many factors to negotiate. Do the marketing people from the tool vendors assist you in that?

**Hugo:** Certainly, up to a certain point. Even with reference users, we always have to try the tools ourselves. We have to ensure they are really compatible with our existing tools and design needs.

**Nora:** Yes, someone else mentioned the user's tool integration issue. What is the story on that?





## STANDARDS EFFORTS—WHO, WHAT, AND WHY ...

### Design Flow Integration

**Hugo:** The problem is this. The IC design engineers seek to assemble the best-in-class tools for their design system. This means picking one tool from vendor A, another tool from vendor B, and so on.

Most EDA tools read in one or more *design files* (file of design data). The tool or the designer may generate additional data and then write out an enhanced version of the design file(s).

Each tool arranges the file data in a specific way (*format*) for fast operation. The different tool functions are generally sequential, with the data passing from one tool to the next. (However, recent tool suites are trying to do more things concurrently.)

Most EDA tools are developed independently of each other. Design groups use various scripting or programming languages (such as Scheme, SKIL, or Perl) to stitch the tools together into a *design flow*. A design flow is the sequence of specific design tools used.

These scripts translate the data from one tool format to another, call up the correct files, and initiate the EDA tool operation. The scripts provide the glue that simplifies the tool design flow for the users. They also reduce the amount of tedious, error-prone manual work in running and using EDA tools.

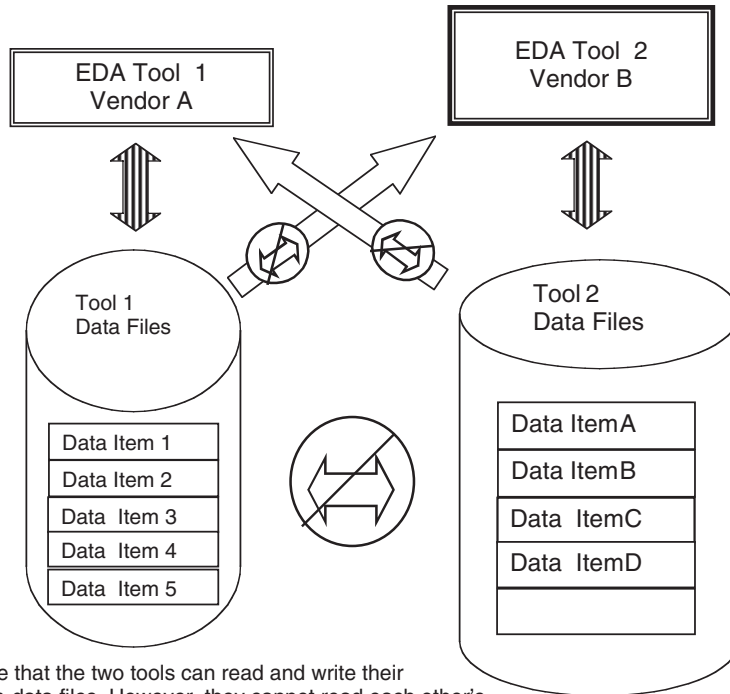
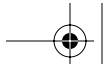
Tool integration scripts would be easier to write if tools communicated in a standard way. However, most tools were developed independently with little concern for a standard *interface* or file format. (There is no standard scripting language, either, by the way.)

An analogy would be different paper filing systems. Suppose I keep a file of newspaper articles organized in three folders by date, author, and newspaper. Suppose you keep a file of articles all in one folder, arranged by date, subject, author, and article length.

I wouldn't know how to find something in your files, and you wouldn't know how to find something in mine. However, some of my articles would be of interest to you and vice versa.

With data files, the *data representation* (numbering, units, location, coding) may also be dissimilar. Here, I will sketch you a picture of what I mean. (See Figure 3.3.)





Note that the two tools can read and write their *own* data files. However, they cannot read each other's data files. Thus, data cannot be exchanged between the two tools or their files without a translator.

**Figure 3.3**  
Lack of Tool Interface Standards

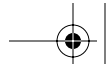
Tool 1 data files may have data items such as:

- Name: Gate31
- Netdelay: 0.3
- Location: G78
- Block #: 300
- Chip #: ACD&

Tool 2 data files may have similar and dissimilar data items such as:

- G78
- Gates—31, 33, 36
- Dly—300 psec
- Revision—88





The number and order of items and the specific data format in each file are different, and each file may have data which the other does not.

So, as you can see, even though the data items contain common information, EDA Tool 1 cannot read the data file of Tool 2. A utility program or script is required to translate the data.

**Nora:** Are there no standards?

## EDA Tool Interface Standards

**Hugo:** Yes and no. There are some. Standards help ease the task of assembling a workable EDA *design flow*.

Tool users or tool vendors, or both, usually create standards. There are always issues of vested interests, politics, resources, and competing standards groups. Other issues include testing conformance to the standard, vendor and user adoption, and so forth. Standards take years to create and may be obsolete by the time they are done!

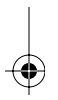
Some standards are created after the industry fights over competing approaches. Then they settle on a standard approach, although it might not always be the best technical solution.

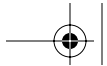
Other interfaces become *de facto* standards simply by broad acceptance and use. An example of a *de facto* standard is the *GDSII* (*Graphic Design Station II*) format. It is used for the graphical layout data of IC masks. (However, there is an effort to replace it with an improved and more compact standard format.)

An example of a planned EDA file standards group is the *Electronic Design Interchange Format (EDIF)*. This is a human-readable file format developed to enable cooperating semiconductor manufacturers to exchange data. Although not intended to be an EDA tool interface, it has been widely used for that.

Vendors who offer a suite of interoperable tools also provide interfaces (often proprietary) between their tools. They may support an interface to **read** data in from other vendors' tools. This would bring the design data (and the customer) into **their** tool suite.

However, it is **not** usually in their interest to **write** data from their tools to a competitor's tool. They might lose the customer. Therefore, although many vendors claim EDIF compatibility, it has been mostly **read-in** only. (The company Engineering DataXpress has long provided translators and other services based on the EDIF standards.)





**Nora:** So there are “semi-standards”?

**Hugo:** I guess you could call them that. It’s a little like railroad tracks. Historically, different railroads used different types of tracks. Each railroad “standardized” on their type of track. However, a train could not move from one company’s track to another! Passengers had to switch trains.

In the 1980s, the industry really tried to create a universal interface.

## Frameworks

The ASIC Council is a small (seven or eight members) consortium of the largest ASIC semiconductor manufacturers. In the 1980s, the council sponsored the Silicon Integration Initiative, Inc. (SI2). It worked on a universal *framework* into which users could easily plug in different vendor tools.

However, the result was multiple vendor proprietary frameworks, instead of a single framework standard.

**Nora:** So standards have not always succeeded. Is there at least a common database for all the EDA data views?

## Design Database Standards

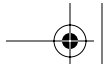
**Hugo:** A good question. The answer is—not yet, but that may be coming. Usually each tool’s files have an internal arrangement of data items (*data structure*) optimized for that tool. The data structure affects how fast the tool works. It also affects how easily different tools can (or cannot!) exchange data.

Some vendors have a proprietary database to which all **their** tools can talk (*interface*). As more multivendor tools need to work together, a **common** design database becomes more important.

An open, universal database would be a great improvement for users. There are significant technical and business implications, making this a difficult standardization goal. Several existing databases or access methods are being considered in a couple of standards committees.

**Nora:** Are there several standards groups?





## Standards Groups

**Hugo:** Yes, many of them. With new ones forming all the time, some compete with each other. As one industry pundit said, “The best thing about standards is that there are so many of them!” I’ll mention just a few groups:

- The Institute of Electronic and Electrical Engineers (IEEE). Primarily for hardware designers, it has developed hundreds of standards.
- The Joint Electron Device Engineering Council (JEDEC), a division of the Electronics Industry Association (EIA). It has electronic product standards of all kinds (48 different committees).
- The Special Interest Group on Design Automation (SIGDA) of the Association for Computing Machinery (ACM). It is for EDA professionals. (In spite of the name, this is a software group, not hardware.)
- The Virtual Socket Interface Alliance (VSIA). For product, IC, and EDA vendors, it creates standards for system-on-chip *Virtual Components* (blocks of intellectual property).
- Accellera—an EDA and designers’ group. It is trying to standardize a formal verification language and a system design language. Accellera manages two prior standards groups, Open Verilog International and VHDL International.
- Open-Access Coalition (OAC). Led by Silicon Integration Initiative (SI2), this user-backed group has established an industry-wide data model and application programming interface (API) which potentially can access any database.

**Nora:** Do your people work on any standards committees?

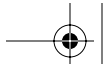
**Hugo:** Sometimes, but it takes a lot of (unpaid) time. We get involved when we need a standard to have a salable product. Sometimes we sit in just to ensure that a competitor doesn’t dominate the standard to fit its product!

**Nora:** Can you tell me more about your EDA staff?

## PERSONNEL—THE KEY TO EDA SUPPORT .....

**Hugo:** That’s a good area to discuss. An essential element in the success of a design group is the EDA support staff. It administers the EDA design system with its tool licenses, libraries, computer, upgrades, and backup support.





In any company, there is a significant learning curve for new people. There is a need for job satisfaction and incentive for experienced people. The company needs to cross-train people. If one person leaves, that should not remove the only knowledge about a particular area or tool.

Most EDA staff members want to work with the newest tools—to develop, test, or use them. However, a large part of the workload is the support and maintenance of the existing tools. EDA managers try to balance this research-and-development (R&D) vs. maintenance dilemma in several ways.

One approach has junior people learning the tools and problems by handling most of the support. They may move on to R&D after they learn about the existing tools. However, experienced staff also needs to be available to train them in the support work.

A second approach rotates the staff through both roles—so everyone does some R&D and some support for a time. This must be done carefully to avoid losing continuity in either area.

Another approach has every R&D person also supporting one or more tools. That way they have a role in, and appreciation for, both areas. They also learn how to make their development work easier to maintain.

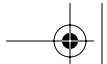
*Nora:* That's quite a balancing act. Where do you get your staff?

## UNIVERSITY CONNECTIONS.....

*Hugo:* From other companies, of course, and from colleges and universities. Many universities have industrial liaison programs to present their research projects to industry. Often EDA companies fund the research and hire graduate students as interns.

We hire quite a few new college graduates—usually former interns. We maintain contacts with several leading universities which are very active in EDA research. Most of these have strong industry relationships as well.





**Did You Know?**

**Many EDA companies (and large ASIC and FPGA vendors) provide low-cost licenses and training to universities. Vendors want the students to become familiar with their product. The students are more likely to use it at work later if the product is familiar. And their companies like the shorter learning time.**

These universities are involved in EDA research, and their professors consult to the industry. Many EDA companies have technical advisory boards that meet regularly. They review the research products and technical direction of the company. Their input is critical to the company for future direction and to avoid noncompetitive products.

Internship benefits the student in learning how the real world works. It also provides low-cost expertise for the company. Many student interns return to work full-time and can contribute immediately with little learning delays. I can give you a list of some universities with which we work.

The list includes:

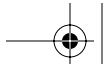
- Carnegie Mellon University
- Massachusetts Institute of Technology
- Princeton University
- Stanford University
- University of California at Berkeley
- University of California at Los Angeles
- University of California at San Diego
- University of Illinois at Urbana
- University of Texas at Austin

And many others.

(Appendix E has websites and other information on these universities.)

**Nora:** Thanks, Hugo. Now I have a better appreciation of our EDA customers' problems.





## SUMMARY .....

There are four important new EDA user questions:

- What is the design organization?
- What kind of network capability is needed?
- What are the security requirements of the computer network?
- What computer systems are needed?

There are several considerations in buying EDA tools.

One is cost versus performance. Another consideration is the choice of licensing options. Training and support are critical factors, as is compatibility with the user's existing tools. Finally, there are concerns about ensuring a smooth transition in using the new tool.

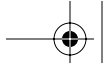
Integrating tools into a design flow is not a simple task. Many EDA tools and data files are not compatible with each other. There are many EDA standards groups trying to improve this situation.

EDA staff comes from universities, electronic product and IC manufacturers, and design houses. Universities have close financial and technical relationships with industry.

## QUICK QUIZ .....

1. Which one is **not** a key **initial** decision for a new EDA user company?
  - a. Design organization (who and where)?
  - b. Network capability needed?
  - c. Time-to-Market?
  - d. Security requirements?
  - e. Computer system needs?
2. Which is not an important factor in buying an EDA tool?
  - a. Runs under Windows
  - b. Offered by major vendor
  - c. Cost/performance
  - d. Have staff already trained on the tool





## Quick Quiz

61

3. Most vendor tools are compatible with other vendor tools.
  - a. True
  - b. False
4. EDA tools training is usually not required.
  - a. True
  - b. False
5. EDA tools usually run quickly and use little memory.
  - a. True
  - b. False
6. There are many types of EDA tool licensing options.
  - a. True
  - b. False
7. Which of the following is an important standards group?
  - a. IEEE
  - b. Accellera
  - c. VSIA
  - d. All of the above
8. Design Flow refers to
  - a. Fast or slow design speed
  - b. Style of design
  - c. Sequence of EDA tools used
  - d. Flowchart design
9. Much university EDA research is supported by:
  - a. IC manufacturers
  - b. EDA companies
  - c. Government agencies
  - d. All of the above

Answers: 1-c; 2-a; 3-b; 4-b; 5-b; 6-a; 7-d; 8-c; 9-d.

