# Management Issues In EDA

Ajit M. Prabhu
Deloitte & Touche
Management Consulting
1633 Broadway, 8th Floor
New York, NY 10019

**Abstract - The Electronic Design Automation (EDA) industry has grown to around $2.5 billion in a span of about 10 years. The dynamics of this industry impacts firms involved in any form of electronic design. Effective management of the design automation function leads to significant competitive advantage. Yet, there has been very little published study in this area. This paper presents a unified framework which captures the essence of the EDA industry . We identify and describe the main forces driving the industry. We present management issues viewed from both the end-users and the EDA companies' perspectives and we use two models to explain these issues. We also show that superior technology is not the only criteria for success in these markets. In this context, we present a socio-cognitive model which addresses the connection between the end-user's perceived need and the criticality of correctly timing the introduction of new products.**

## I. INTRODUCTION

Electronic Design Automation (EDA) was virtually non-existent 12-15 years ago. In this short period, it has grown to around $2.5 billion, exhibiting revenue growth rates of more than 60% annually upto 1988, between 20-30% till 1990 and double digit growth projected till 1995. Within the industry, however, the competitive landscape continuously changes and growth has flattened for the larger players (Cadence, Mentor). Their place on the growth curve has been taken by companies like Synopsys and Viewlogic.

The changing dynamics of this industry impacts firms involved in any form of electronic design. For these firms, effective management of the design automation function leads to significant competitive advantage. On the other hand, EDA is a cost and errors in vendor selection and/or internal development can become competitive disadvantages and negatively impact time-to-market or significantly affect a product's cost structure.

For the EDA provider, the market demand for continuous innovation implies constant competitive pressure. Managing rapid growth is itself a challenge. In addition, EDA requires management of critical issues, such as the constant metamorphosis of the underlying market from technology-pull (new technologies) to technology-push (incremental changes), the close link with the computing industry, and the progression into industry consolidation. Misplaced focus can very quickly lead to the demise of a company, as evidenced by one-time market leaders such as Daisy and Cadnetix, or loss of market share and/or reduced profits through increased costs (Mentor, Cadence).

The basic goal of this paper is to present a framework for EDA management. In any industry, there are providers and end-users (consumers) and for a framework to be useful, it must address both facets. This framework consists of two management models - one for the EDA vendor and the other for the EDA user. We also discuss some critical observations related to the cognitive aspects of new product development and market timing. While these are not the main focus of this paper, we believe that they are extremely important issues in technology management.

We begin by briefly discussing the structure of the industry. This section also deals with the importance of end-user perception, and the effects of lock-in and lock-out on the EDA user. We show that these phenomena occur even in high-technology markets driven by technology-pull. We then present our framework. Concluding comments center around the framework and it's application to the EDA industry. The socio-cognitive model for technology management is also discussed in this section.

We assume familiarity with the EDA market and the events which led to its current state.

## II. DYNAMICS OF THE EDA INDUSTRY

### *A. Industry Consolidation and the Dominant Design Paradigm Re-defined*

In any industry, there is an evolutionary process which defines its current structure. In a "classical management" sense, the industry evolves from the rapid growth of small companies to consolidation and onto relative stability. This stage is often a sort of dynamic equilibrium, where the dynamism either reflects the entry and exit of players, or some restructuring, which does not fundamentally affect the direction of the market. Evolution, however, implies a process of natural selection. While this may be true, in general, we show that EDA actually followed a *logical* progression, where there was some natural selection - but this was *not* the main evolutionary force.

Assume a simple design process consisting of Design Capture followed by Design Verification and then Physical Design, with iteration permitted between the steps. A discussion of system, or ASIC design methodology and the design process can be a subject of considerable debate. But for our purpose let us assume that this process is representative.

The promise of EDA has always been "***top-down***, ***technology independent***" system design which aims to focus the designers attention on the design capture and verification phase (creativity), while automating the implementation tasks. In other words designers decide *what* they want done and let the tools decide *how*. For ***top-down*** design to exist, however, there is a fundamental requirement - design abstraction to a higher domain. In the ASIC context it is the transition from the logic/functional design description to the behavioral domain. This requirement means design automation tools (EDA), to *unambiguously* capture the complete design intent without undue concern for physical implementation detail or implementation technology. Verification mechanisms to verify the design intent and automated, optimized design implementation in the technology of choice are *assumed* to be available. But, introduction of new abstraction mechanisms and the supporting design tools required new and unfamiliar thinking by the end-user, since it completely re-defines the current design process. This compounds the difficulty of new technology introduction, and in effect forces a ***bottom-up*** approach. This is why the EDA industry delivered using the ***bottom-up*** approach. The first successful automation tools were automatic layout and low-level gate/functional simulation. They did not require elaborate design abstraction mechanisms and were therefore more palatable to the design community.

The implications of this expectation of ***top-down*** design and the supporting tools are profound, both for EDA companies and their end-users. Basically, it is this promise which has driven the industry to its present structure and one which will continue to dominate the evolution for the next few years. For example, behavioral languages have been around since the early 1980's. They were the first step towards abstracting the design to a higher, technology independent domain. Until fairly recently however, their use was only to document the design. Even if the supporting tools, such as simulators and synthesis tools, did exist, they were not successful in the market, since the new conceptualization processes were not in place for the end-users. In effect, this amounted to a market failure. From a technology management perspective, this is a very significant observation, since it is directly connected to new product introduction.

Technological success is not simply defined by superior technology. Especially in the growth phase, it is the careful combination of leading-edge technology coupled with an embodiment of *perceived* reality. The perception is associated with delivery of a "solution" which addresses a *perceived* bottleneck irrespective of whether the bottleneck is a symptom or a cause of the real problem. Although the technology may exist to address the cause, the delivered solution will succeed only if it directly solves the *perceived*, most pressing need or it can be cloaked to do so until a ***dominant design***[13] emerges. The EDA industry is replete with companies which had superior technology but failed because of incorrectly timing product introduction. Of course there are companies which succeeded purely through technology-pull, i.e. through superior technology. But, in these cases, it is the *timing* of the introduction which is a key determinant of success. The concluding section utilizes a socio-cognitive model to explain this phenomena further. In this section, the *promise* of ***top-down*** design and the *delivery* of ***bottom-up*** design is used to illustrate this concept as well as explain the industry consolidation.

The infancy of the EDA industry, coupled with the requirement to redefine the design process, was a natural barrier for the introduction and acceptance of the design abstraction philosophy. As stated earlier, the main reason was the compounded effect of the introduction of new tools together with the necessity of a new form of conceptualization. The delivery of EDA vendors therefore took the path of least resistance - ***bottom-up*** design. Herein lies the key to understanding the "consolidation" of the EDA market. Design automation being in its infancy, had significant investment sunk in research and this resulted in a spate of new technology. The combination of technology-pull coupled with the easier *acceptance* of a ***bottom-up*** process led to the delivery of point solutions which addressed automation of specific steps in the *existing* design process. Successful companies were those which provided niche products (point solutions) *perceived* by the end-user as bottlenecks, in the design process. Hence, for example, early EDA vendors specialized in circuit simulation and automatic IC layout. The technology embedded in each of these niche products was complex and new, but they were introduced with relative ease because they addressed what was believed to be an immediate need. The EDA market was thus fragmented and consisted of small, highly specialized companies competing solely on technological superiority. End-users were caught up in assimilating technology and the market was clearly being pulled by the technology.

The ***dominant design*** paradigm [13] states that for a given technology various competing designs progress through the early stages of research and development (eras of ferment) and a single dominant design emerges as the winner. The ***dominant design*** is then incrementally improved (era of incremental change) by companies which choose to compete. In our context, ***dominant design*** is obtained when two competing products have similar end-user performance characteristics, *even if the underlying technology is completely different*. For example, from the early to mid 1980's, companies providing software logic simulation were developing new algorithms (products) with very rapid improvements in performance (CPU usage). End-users were quite willing to switch companies simply to gain this performance advantage. A threshold was reached after which the same level of performance improvements was not sustainable. This is the ***dominant design*** stage. It is the point when the technology stops pulling the market. Even though significant performance improvements continue to be provided till today, unless they are breakthroughs, they will not be sufficient to convince end-users to switch simulators. The major competitors however, do not necessarily have the same or even similar simulation algorithms (underlying technology), but simulation tools are definitely in the era of incremental change.

Now, if we accept that the basic expectation for EDA is ***top-down*** design, then once the point solutions reach the ***dominant desig***n phase, a "consolidation" phase must begin since the size of the market is finite. In other words, the merger of pieces (point solutions) to solve the real problem of ***top-down*** design must take place (a) in order for EDA companies to continue to grow and (b) evolve towards provision of system-level EDA solutions. The important point here, is that this movement is not vertical integration or consolidation in the "classical" management terminology, but rather a natural progression towards what the end-user has always been implicitly expecting.

In reality, this is exactly what has happened in EDA and will continue to happen. Mergers and acquisitions continued at the rate of one per month from 1987 to 1991. Market leaders such as Daisy and Mentor Graphics who failed to see the strategic value of mergers soon paid heavy penalties. Daisy merged with Cadnetix in an effort to dominate the market, but eventually went out of business partly because of their focus on using EDA software to sell proprietary hardware. Mentor Graphics, which also believed that the revenue lay in selling turnkey systems consisting of hardware together with their own EDA, began to lose market share. Mentor has returned to profitability by turning to a software-only approach to revenue generation and promising an integration tool which links point solutions to create an integrated system. Cadence Design Systems on the other hand, focussed only on EDA software and integration of the various point solutions it acquired. In four years it made strategic acquisitions to "round-out" its product line and in 1992 became the market leader with revenues of $500 million.

### *B. Lock-in and Lock-out*

The EDA industry and the industries constituting it's complimentary assets[12] are characterized by rapid changes in technology and fierce competition. These industry characteristics create frequent changes as companies leapfrog each other with technology and attempt to protect their interests through proprietary regimes. Lock-in and lock-out are important industry considerations for both the user and the vendors since switching costs for both can be very high.

An EDA vendor's switching costs are related to the size of its user base, the extent of its alliances, and the breadth and depth of its R&D capability. Ideally, an EDA vendor would like to jump on the bandwagon of every "hot" technology or platform but is constrained due to resource limitations. In addition, the vendor cannot abandon older embedded technologies and risk cutting off it's current user base. Thus, to offer users robust choices in the market EDA vendors create alliances with the companies which sell complimentary assets. The alliance has costs associated with it because software must be developed. But an incorrect choice can lock them into an unsuccessful company and lock them out of the new rising star which forms an alliance with a competitor.

End-Users on the other hand, get locked-in to an EDA vendor's tools because of switching costs related to how far down the learning curve the customer is and how much data has accumulated in the system. A considerable amount of time and effort is spent by an organization assimilating EDA products into its process. This assimilation includes the explicit training of users, tacit understanding of the system, application software built over the EDA framework, and interfaces to other systems. Also, as the system is used, an inventory of design data builds up. The cost of switching tools can be very high if this data must be migrated to accommodate another EDA vendor. The other component of the switching cost is the sudden obsolescence of the knowledge and expertise in the tool set and the cost of learning the new system.

The importance of complimentary assets is amplified by the "critical mass" nature of the business. Once an EDA vendor garners enough market share, and/or powerful customers, the complimentary product makers are forced to do business with them regardless of their technical merits. For example, it is in the best interest of an FPGA manufacturer to have its tools interface with as many EDA vendors as possible. The more users who have access to their technology through existing investment in EDA, the more FPGA devices they can sell. Yet, the FPGA company has limited resources and usually can not create robust interfaces to all EDA vendors. Thus, those EDA vendors with the greatest market share, or growing the fastest, will get the lions share of the support from the component supplier (FPGA vendor). Conversely, the EDA vendors are stuck in the same situation as they try to support the hottest complimentary products, which may or may not provide a consistent long-term revenue stream.

Due to the "critical mass" nature of the industry it might be expected that eventually the largest industry players will grow at the expense of smaller firms until one dominant firm emerges. Historically this has been true, but only to a point. The rapidity of change in EDA and complimentary technologies creates lock-in/lock-out and a large user base can become a liability, for the EDA company. A firm may be forced to expend a major portion of its resources supporting its alliances and its existing user base. In this situation, competing firms can develop alliances with "hot" new technologies or introduce their own new EDA technology more easily.

For example, by the late 80's Mentor had become the dominant EDA vendor and was clearly showing signs of having attained "critical mass" as complimentary asset vendors sought its user base. But, Mentor was committed to running only on the Apollo platform at a time when Apollo was rapidly losing market-share to Sun in the workstation race. Mentor's slowness in porting to Sun combined with often delayed releases of new technologies tempered its dominance and permitted Valid and Cadence to gain ground in the race.

## III. THE M1-M2-M3 MODEL-MANAGEMENT OF EDA FOR END-USERS

Steiner and Teixeira[11] use the M1-M2-M3 framework to assess technology and its impact on competitiveness in the banking industry. The fundamental concepts of this framework can also be used to gain insight into the EDA industry. Though this model is discussed with the end-user as the main beneficiary, EDA vendors must also understand it, as we demonstrate in this section.

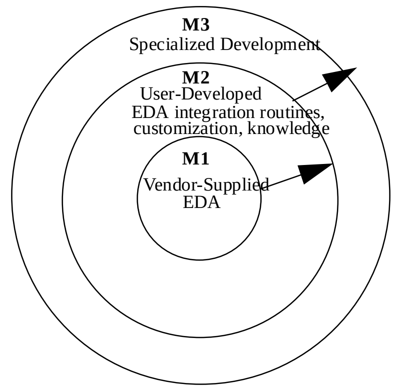


**Fig. 1** The M2-M2-M3 Model

### A. Definition

The M1-M2-M3 framework presents a hierarchy of technology ranked by the competitive advantage it offers a company (see Fig. 1.). In the banking context, M1 technology refers to the basic hardware and systems that a bank uses but which is available to everybody in the industry, such as check processing equipment. The commodity nature of this level of technology implies that no bank can gain a large, sustainable competitive advantage through its use. Yet, investment can not be foregone since it is a requirement to be a player in the game. While employing M1 technology will not offer competitive advantage, not employing it will most certainly present a disadvantage, perhaps a crippling one.

M2 technologies are the tools (generally software) and processes that bind together the basic automation of M1 into a system unique to a particular company. At this level there is an opportunity to gain a sustainable competitive advantage, since this technology is both tacit in nature and proprietary to each firm. Thus, its deployment can be both advantageous and protectable. M3 refers to the use of technology to change the strategic thrust of a company. At this level there is even more room to create competitive advantage.

The boundaries between M1, M2 and M3 are not static. Especially in a rapidly evolving, technology-based industry, such as EDA, these boundaries are dynamic. This movement is illustrated by the arrows in Fig. 1. An understanding of the technology evolution and its relationship to the evolution of the industry is critical to end-users. To illustrate the application of the M1-M2-M3 framework to EDA we use several examples to show the evolution of the M1/M2 line over the years.

### *B. Application of the M1-M2-M3 Model to the EDA Industry*

Arguably, the first EDA tool was SPICE, a public domain software package. Since the software was readily available, it was an M1 level technology. However, SPICE at that time, was largely only a simulation engine. The accuracy and usefulness of SPICE was a function of the creation of component models by the user and fairly detailed knowledge of the software. These models and the associated knowledge were therefore M2 technology since they were unique to each firm, protectable, and sustainable.

SPICE was available before the formal EDA industry developed, and it became widely used and accepted. Thus, SPICE has become a defacto ***dominant design*** in the industry for analog circuitry. Over the years, many companies have made incremental proprietary improvements to SPICE and sold them under various names such as PSPICE or HSPICE. Each product is generally designed to target a specific niche in the analog design market. SPICE's long legacy has kept it as the ***dominant design*** despite technical limitations. Only recently have other analog simulators, such as SABER challenged it's dominance. Another similar example is ESPRESSO.

Before the EDA industry emerged in the late 70's and early 80's, tools were only developed for internal use by large companies, such as IBM and AT&T. Other than SPICE, these tools were not commercially available and no M1 technology existed. The emergence of the EDA industry made EDA software available to all companies and an M1/M2 split developed.

In the early 80's, Calma was the premier IC layout tool on the market. This product was available to everyone except companies locked into another technology. Thus, the basic automation aspects that Calma provided were M1 technology. Organizations which did not possess Calma or a Calma-like engine were probably not competitive in the Semi-custom IC development area. This raises the issue of *competitive parity*. Firms which do not have the basic M1 infrastructure are at a competitive disadvantage. For EDA users, maintaining competitive parity requires an understanding of the structure of the EDA industry and the current position of the automation tools in their respective evolutionary paths. In this Calma example, *not* possessing a competitive IC layout tool places a company below par when compared to it's competitors.

While Calma provided the basic hardware and software to automate the task, it still required considerable programming and customization to work well in a user's process. This additional programming, operating skill, and integration into the user's environment were the M2 aspects of the technology.

The switch from turnkey systems (bundled hardware and software) to software only solutions is an illustration of the shift of the M1 level. From it's inception to the mid eighties, EDA software required more computing power than was available from general purpose hardware. EDA companies, therefore, developed their own turnkey systems and were able to use its performance as a competitive weapon. From the end-user's perspective, no dominant design had emerged and the hardware was *not* M1 technology.

By the mid eighties, however, the processing power of generic hardware was sufficient to support most EDA applications without special tuning. Combined with the growth of the workstation market, this meant that hardware had become a commodity in the user's view and it no longer provided a competitive advantage to either the user or the vendors. Thus, the turnkey approach to selling EDA systems changed to the software only approach of today. Once workstation hardware became a market commodity (M1 technology) customers became reluctant to pay a premium for it and most vendors realized it was not only becoming a low margin item but also that it fell outside their core competency [10]. Mentor's late realization that hardware had become an M1 technology and its continued reliance on selling turnkey solutions cost them dearly in lost revenue.

Today the M1 level has made serious inroads into much of the EDA software itself. Schematic capture is largely considered a commodity with little competitive advantage for either the user or vendor. Not only are the basic features of digital simulation perceived as standard technology but robust interfaces to schematic capture (and to a lessor degree other tools, such as, PCB) are commonplace and expected

As pointed out earlier, user's perceptions are an important factor in the dynamics of the EDA industry and in most cases M1 technology is defined by the user's perception of the technology rather than the actual technology. The technology is moving very rapidly and it is the user's perception of the current bottleneck(s) that defines the M1/M2 line. Once the user (or organization) has determined (correctly or incorrectly), that a technology is no longer a road to competitive outvoting it becomes M1 technology, even if no dominant design has emerged. For example, digital simulation has been around since the early 80's and a simulator's speed and accuracy were strong selling points and were used as marketing differentiators. Currently, simulators are considered as M1 technology. For competitive parity reasons, everyone has a good, fairly accurate simulator and there is at least the perception that the simulation engine itself will provide little advantage since the users attention has moved along to newer, sexier technologies. In reality, there are some major differences in the performance of simulators, but unless the differences can be shown to be very large, these points taken in isolation will not sell the simulator.

Thus, the M1-M2-M3 model is important to both EDA users and vendors. Users must understand M1 in order to create a basic competitive infrastructure. The investment in M1 is for "commodity" tools and it provides competitive parity. If it does not exist competitively, then the firm will be operating inefficiently. M2, on the other hand can, and should be used for competitive advantage.

For EDA vendors, a misunderstanding of the position of the M1/M2 will have several detrimental effects. First, tools belonging in M1 cannot command a premium price and mis-pricing can quickly lead to loss of market share. Second, the R&D effort must be constantly tuned to M2 and M3. Otherwise, resources will be wasted on incremental change which will seriously undermine the long-term competitive position of the company.

## **IV.** THE SPIRAL MODEL - MANAGEMENT OF EDA DEVELOPMENT

The Spiral Model has been proposed to address the management of the software life cycle[2]. It is basically a process model used to determine the order of the stages in software development and to establish the criteria for progressing from one stage to the next. We propose a modified Spiral Model for the management of EDA development. Our modifications change the model from management of the *software development process*, to management of the *technology functions* of the company.

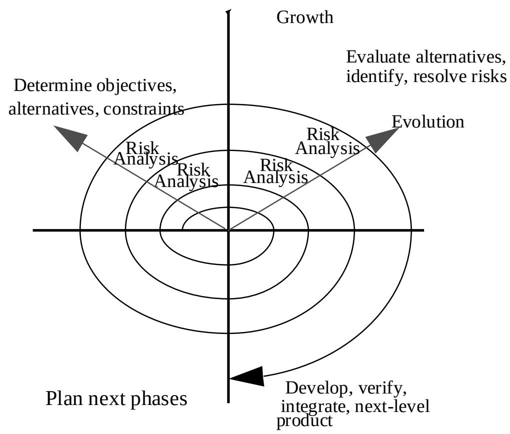


Fig. 2 - The Modified Spiral Model

In the original mode, the radial dimension represents the cumulative cost incurred in accomplishing the various steps in the software development process, such as prototyping, design, coding, testing and documentation. The angular dimension represents the progress made in completing each stage of the spiral. The spiral itself reflects the underlying assumption that each cycle involves the same sequence of steps. Each cycle is completed by a review involving the primary people or organizations concerned with the product.

EDA companies intending to grow or sustain their current market positions, must understand the reasons behind the shift from M1 to M2. More importantly, they must be able to identify the *direction* of the shift, in the short term as well as in the long term. In other words, strategic management must focus on correctly identifying the technological trend and allocate resources accordingly. This is crucial, since technology diffusion is very rapid, appropriately regimes [12] are weak, and as history has demonstrated, stragglers will not survive the purge which occurs every time the M1 line shifts. We emphasize that the dynamic nature of this industry means that unlike other industries, technology evolution does not take decades but relatively few years. It is therefore very important to be on the leading-edge in order to appropriate profits.

The Spiral Model, with some modifications, adequately captures the growth curve of EDA companies. Given that the consolidation is towards system suppliers, the ultimate market demand is for an *integrated* set of tools to address ***top-down*** design. Some point solutions still need to be built. Whether these are contracted out, acquired, or built internally, they must be integrated. Hence, the basic task, from an implementation point of view, consists not only of developing new tools, but building on developed software and integrating with tools in the M1 category.

An EDA company usually begins with development of a tool and then progresses towards an integrated system of tools. Progress, in this context, can be through the supply of additional products, acquisitions or mergers. In any event, the complexity of the software *always* grows. This growth is captured in the modified Spiral Model, (hereafter called the MSM). The radial dimension can be used to represent either the cumulative cost or the potential profits to the company. If cost is used, then the model can be used to track the cumulative development cost of the addition of each new project. The angular dimension measures the progress towards the final market demand, i.e. the technological evolution.

Fig. 2 shows the MSM. Each quadrant identifies tasks related to the relevant phase. These tasks are self-explanatory and will not be discussed here. An important part of the MSM is the risk-analysis. The risk-analysis can be done in two stages. It is well-known that the cost of project abandonment increases dramatically as the project proceeds towards completion. Hence, risk-analysis should be done in the first two stages, i.e. during determination of objectives and alternatives and in the evaluation of the alternatives. The option-pricing model [4] is the most suitable for this analysis. This model should be used as a guideline and not as the sole determinant of the decision, since the externalities are difficult to capture and are often inconceivable, as in any high-technology product. But, the advantage of the option pricing model is that it permits cost justification in stages, compared to traditional methods such as Net Present Value.

It should be clear why the largest market players are concentrating on providing frameworks for tool integration. In addition to recognizing that profits in EDA lie in software and not in turnkey systems, Cadence Design Systems was one of the earliest tool integrators. They developed and marketed expertise in this area before their competitors and forced the "open-systems" approach so that they could appropriate profits through external marketing agreements for tools they did not have. This practice has now become common, across the industry, with different flavors of OEM agreements.

To illustrate the application of the MSM, consider Synopsys, a relative newcomer to EDA. It began the spiral by providing superior technology for a market niche - logic synthesis. Synopsys quickly garnered a dominant market share, purely by technology pull. The next cycle of the spiral was a progression towards synthesis using a subset of VHDL, which is a higher level of abstraction. This strategy succeeded because simulators already existed for VHDL and the language had begun to gain acceptance as a modeling language. The progression was built with tight integration over existing modules. Synopsys continues to grow its market share by maintaining a *relatively* tight appropriability regime [12]. Employee turnover is relatively low and technology is not leaked through publications. In addition, most EDA vendors have OEM agreements with them. They are now entering the next phase of the spiral by building on the existing technology and providing test synthesis and simulation products.

## **V.** DISCUSSION AND CONCLUSION

We have discussed two models so far - the M1-M2-M3 and the Modified Spiral Model, which are applicable to the management of EDA for the end-user and the EDA company respectively.

Garud and Rappa[6] have developed a socio-cognitive model to describe technology evolution. It is beyond the scope of this paper to discuss their model, but we borrow some insights related to our notion of *perceived reality*. They describe the process of technology evolution as a three-way triangle consisting of Evaluation Routines, Artifacts and Beliefs. Researchers first build artifacts of what they believe can and should be done. Simultaneously a set of routines are developed to evaluate how well the artifact meets expectations. Two divergent outcomes can then follow: (1) If there are discrepancies between reality and the expectations these influence the beliefs which close the loop and the cycle of development continues (2) if artifacts do not exist or are not fully developed, the evaluation may produce unsatisfactory and erroneous results which could kill further development.

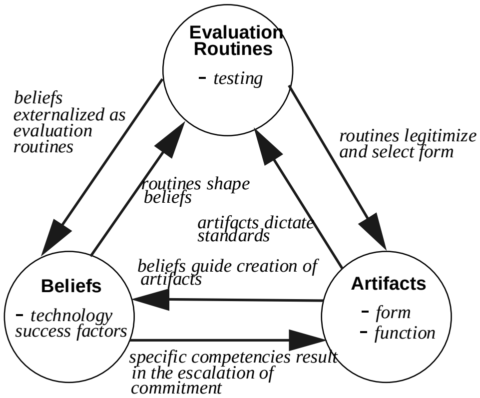


**Fig. 3** - Socio-Cognitive Model of Technology Evolution

In EDA, new technology requiring new forms of conceptualization, will continue to meet with initial market resistance. The reason for this is the evaluation is difficult. It should be mentioned that the evaluation routines do not refer simply to the *process* of evaluation. They embody an understanding of the underlying technology. This understanding is necessary in order to obtain an objective assessment of the capabilities and potential of the technology being evaluated.

The EDA user would like to use new technologies, but evaluation of these technologies requires a fair amount of investment, since the concepts underlying the technology are unfamiliar and complex. Since the evaluation routines do not exist they must work with the EDA company or research institutions such as universities to define the routines. This is resource consumptive and may also lead to biased results. Hence, the introduction of new technologies must be carefully timed or else they will be lost. In economic terms, this means failure to create a market and hence appropriate first or second-mover profits. As is well known, first-mover advantage is significant because of the price premium which can be obtained.

To illustrate this phenomena in EDA, we use synthesis. Synthesis existed as early as the mid-1980's, offered commercially by companies such as Silc Technologies and through internal offerings in large companies like AT&T and IBM. Silc did not succeed, however, since their offering focussed on input from the behavioral domain while users were still conceiving their designs at a lower level of abstraction (gate level). Evaluation of Silc technology required (1) abstraction of the design to an unfamiliar domain and (2) an understanding of the embedded technology so that the output could be deciphered. Synopsys on the other hand introduced gate-level logic synthesis. This level of abstraction was dominant. In addition, at the same time, there was a flurry of publications associated with gate-level logic synthesis methods at all the major conferences. Acceptance of this technology was therefore easier since the evaluation routines could be focussed on the output produced by the tool rather than on the concepts behind the tool. In AT&T's case, at this time, there were two sets of synthesis tools offered: logic-level and behavioral-level. The acceptance level of the logic-level tools was comparatively good, while the behavioral-level tools did not gain acceptance. A middle-of-the-road strategy which introduced pseudo-behavioral level models was subsequently introduced and this succeeded. These models were really logic-level descriptions (RTL level), described in a higher-level language.

The above discussion leads to the conclusion that the timing of the introduction is crucial. It is this factor which is different from other consumer driven markets. For high-technology products, markets are often inefficient because the evaluation mechanisms are undeveloped. This is the reason why organizations like the FDA take years to evaluate new technologies. The implications for technology companies, therefore, are serious because they must expend expensive resources to ensure that the market inefficiency does not obliterate their products.

## *A. Implications for EDA Users*

Competitive parity and competitive advantage are critical factors contributing to the success of any firm. We used the M1-M2-M3 model to illustrate these factors, in the EDA context. The fact that the boundaries between M1, M2 and M3 are not clearly defined can be used to gain early advantage. They can also be used to drive the EDA market. On the other hand, a misunderstanding of the M1/M2 line can become a significant disadvantage due to the lock-in phenomenon.

The most important element to consider for the user firm is the M2 level of technology. This level is the software or process that enables the automation hardware to function in the user's design flow. Because of its unique nature, it is protectable and provides a competitive advantage. In the rapidly changing EDA industry, the definition of M1 and M2 technology changes quickly. As the breadth of technology falling into the M1 category grows creation of M2 technology for advantage becomes a more challenging problem.

Until the late eighties, EDA products generally targeted the most easily automated design problems. These were mainly point solutions. The proliferation of these point solutions, usually from different vendors, became known as the "Islands of Automation" paradigm because without industry standards data could not be exchanged between the tools. The economic justification for purchasing the EDA software was fairly clear in this context. For example, if automated PCB routing could be done 50% faster compared to the manual mode, simple calculations could be used to show that the investment made economic sense, and it was a necessity to be competitive. In other words, the technology was largely M1. The M2 aspects were customized software or procedural changes that best integrated the new tools.Thus, for point solutions, the M2 aspects are generally within organizational units. Hence, they are clearly visible, easy to quantify, and easy to implement.

The "Islands of Automation" problem is being solved and currently the industry focus is on the overall design process with the eventual goal of true top-down design. Thus, the M2 level has moved beyond the organizational unit to the entire design process, frequently covering many functional units. A design organization can no longer achieve competitive advantage through automating individual functions. These are standard industry practice - M1 technology. Integration of units, or the automation of the whole process, is where competitive advantage lies.

This presents a whole new set of circumstances and a much more challenging set for both the user to take advantage of and the vendors to sell into. When solutions were point based, M2 technology was under the control and guidance of the group who bought the tools. As the M2 capabilities become intergroup dependent they require many intergroup changes including the redefinition of group functions. Justification becomes harder to quantify and political considerations create resistance to change. Decisions that affect the entire process get pushed up the management hierarchy to people less familiar with EDA technology. The end result is that as the M2 level decisions are pushed up they are harder and riskier to get correctly resolved. But, precisely because of these difficulties, the rewards to the firm, in terms of a sizable competitive advantage, will be correspondingly greater. On the other hand, bad decisions will soon lead to fragmented EDA within the organization and the competitive level will be limited to M1. As shown earlier in this paper, as well as in [11], competitive parity must exist for the firm to remain profitable. Hence, resource consumption focussed only on M1 will eventually cause the firm to be non-competitive.

### *B. Implications for EDA Companies*

Since the market is no longer is driven by technology pull, but by technology-push, integration of tools to provide a system solution (top-down design) continues. However, software integration is a complex and risky task. EDA users are therefore demanding that the EDA companies take on the task of providing an integration mechanism. The user will then have the opportunity to choose the best-in-class tools from various vendors. This is the equivalent of the “open-systems” approach and is called “plug-and-play” in the EDA context. In our M1-M2-M3 model, this means the widening of the M1 circle.

It is well known that Sun Microsystem’s open-system approach was a major factor in its growth. More importantly, it is the *externalities* derived from this approach from which Sun Microsystems has benefited. Similarly, an EDA company which can provide a true seamless integration tool will be able to benefit from the externalities derived by the tool becoming a de-facto industry standard. One example is the services it can provide to help end-users integrate their EDA tools of choice. On one hand, this may seem counterintuitive to the EDA company, since it opens the door to competitor’s tools. But, users have their own integration methods anyway (their M2) today and are therefore looking to move this function to the EDA vendor so that it will be supported, robust and potentially lower their operating costs.

The ***top-down*** design philosophy will also come to pass, albeit slower than expected. The reasons for the reduced pace of introduction are as discussed in the socio-cognitive model. These reasons must be considered when future strategy is being defined. Niche products can and will still be introduced in the interim. The larger EDA companies should aim for economies of scope, providing well-integrated, system-level solutions. But they must also be prepared to unbundle tools and integrate them with other vendors solutions. This service will be an increasing source of revenue, since it ties in with the market demand for an integration framework.

So far, the appropriability regimes have been fairly weak. This trend is changing, and EDA companies are not publishing their proprietary algorithms which directly contribute to their competitive advantage. This protection is necessary and is applicable to the tools which are close to the dominant design phase. For emerging technologies, this is a more difficult problem since the evaluation routines are undeveloped. Therefore, the benefits of publication must be carefully weighed against the cost of unknown evaluation routines. It should be noted that (a) the knowledge is tacit and controlled by key developers (b) the technology diffuses very rapidly. Protection mechanisms must revolve around these two aspects.

EDA vendors must also be sensitive to the user’s switching costs. This issue arises when new tools or technologies are introduced and also plays an important role in the marketing of existing products. It is an important issue during the consolidation phase, when acquired tools or new tools are being integrated with existing ones. Horizontal integration to eliminate competition has a price in this industry - support for the embedded user-base must continue. This can be a high cost item and bears the risk of sending an incorrect signal to the industry. The cost of software integration is high (in this context, integration is the merging of acquired technology). Coupled with the switching cost, the overall cost of merger or acquisition to the company can be large enough to consume precious development resources. Undoubtedly this will effect the long-term competitiveness of the company, as evidenced by the Cadence-Valid Logic merger.

The Modified Spiral Model is a useful tool to use for strategic development and evolutionary decision-making. The financial evaluation of horizontal or vertical integration, or new product development should use the option-pricing model instead of more traditional approaches such as the Net Present Value method.

In conclusion, we have identified key factors to be considered by participants in the EDA industry. Our approach is to provide a unified view, from both the provider and the end-user’s perspective. We use three models to discuss the issues we raise and conclude with implications which tie our issues and explanations into a framework which can be extrapolated to other rapidly evolving technologies.

ACKNOWLEDGEMENTS

Professor Raghu Garud, Stern School of Business, New York University helped in many ways. Jay Posklensky, NYU, also participated initially. The author would also like to thank Hao Nham and Jean Dussault at AT&T Bell Laboratories, Murray Hill, NJ, and Don Curtis, Deloitte & Touche, for their support and encouragement.

REFERENCES

[1.]Abernathy, W., Clark, K., "Innovation: Mapping the Winds of Creative Destruction", Research Policy, 14, 1985.
[2.] Boehm, B. W., "A Spiral Model of Software Development and Enhancement", IEEE Computer, May 1988
[3.]Copeland, T., Koller, T., Merin, J., "Using the Option Pricing Model to Value Flexibility", Valuation: Measuring and Managing the Value of Companies, John Wiley & Sons, 1990.
[4.]Dos Santos, Brian, “Justifying Investments in New Information Technologies”, Journal of Management Information Systems, Spring 1991
[5.]Garud, R., Rappa, M., “A Socio-Cognitive Model of Technology Evolution: The Case of Cochlear Implants”, Submitted to Organization Science, Nov 1992.
[6.]Garud, R., Nayyar, P., Schapira, Z., "Technological Oversights and Foresights", New York University Working Paper.
[7.]W. F., "Managing R&D As a Strategic Option", Research in Technology Management, May-June 1985.
[8.] Porter, M. E., "The technological Dimensions of Competitive Strategy", In R. Rosenbloom (Ed), Research on Technological Innovation, Management and Policy, Vol 1, JAI Press, 1983
[9.]Porter, M. E., “The Competitive Advantage of Firms in Global Industries”, The Competitive Advantage of Nations, Free Press, 1990.
[10.]Prahalad, C. K., Hamel, G., “The Core Competence of the Corporation”, Harvard Business Review, May/June 1990.
[11.]Steiner, T., Teixeira, D., “Technology in Banking”, Dow Jones Irwin, 1990
[12.]Teece, D. J., "Profiting from Technological Innovation: Implications for Integration, Collaboration, Licensing and Public Policy", The Competitive Challenge: Strategies for Industrial Innovation and Renewal, Cambridge Mass: Baligner Press, 1987.
[13.] Utterback, J., Abernathy, W., "A Dynamic Model of Process and Product Innovation", OMEGA, Vol 3, No. 6., 1975