Carnegie Mellon University
Research Showcase

Department of Electrical and Computer Engineering

Carnegie Institute of Technology

1-1-1994

Benchmarking an interdisciplinary concurrent design methodology for electronic mechanical systems

Smailagic
Carnegie Mellon University

Carnegie Mellon University.Engineering Design Research Center.

Follow this and additional works at: http://repository.cmu.edu/ece

Recommended Citation
Smailagic and Carnegie Mellon University.Engineering Design Research Center., "Benchmarking an interdisciplinary concurrent design methodology for electronic mechanical systems" (1994). *Department of Electrical and Computer Engineering.* Paper 182.
http://repository.cmu.edu/ece/182

This Technical Report is brought to you for free and open access by the Carnegie Institute of Technology at Research Showcase. It has been accepted for inclusion in Department of Electrical and Computer Engineering by an authorized administrator of Research Showcase. For more information, please contact research-showcase@andrew.cmu.edu.

NOTICE WARNING CONCERNING COPYRIGHT RESTRICTIONS:
The copyright law of the United States (title 17, U.S. Code) governs the making of photocopies or other reproductions of copyrighted material. Any copying of this document without permission of its author may be prohibited by law.

Benchmarking an Interdisciplinary Concurrent Design Methodology for Electronic/Mechanical Systems

A. Smailagic, D.P. Siewiorek, D. Anderson,
C. Kasabach, T. Martin, J. Stivoric

EDRC 05-89-94

Benchmarking An Interdisciplinary Concurrent Design Methodology for Electronic/Mechanical Systems

Asim Smailagic
Daniel R Siewiorek
Drew Anderson
Chris Kasabach
Tom Martin
John Stivoric

Engineering Design Research Center
Carnegie Mellon University
Pittsburgh PA 15213

Abstract*

The paper describes the evolution of an Interdisciplinary Concurrent Design Methodology (ICDM) and the metrics used to compare four generations of wearable computer artifacts produced by the methodology at each stage of ICDM's growth. The product cycle is defined, its phases, and the design information representation for each phase. Six generic axes of design activity are defined, and the concept of benchmarking a complete design methodology using these axes is introduced. In addition an approach for measuring design complexity is proposed. When applied to the four generations ot the CMU wearable computers, the ICDM has demonstrated two orders ot magnitude increase in design and efficiency.

* **This work has been supported by the Engineering Design Research Center, a NSF Engineering Research Center, and Advanced Research Project Agency.**

1 Introduction

Products of the future will be technologically more sophisticated, highly customized, produced in small batches, and brought to market quickly. These products will require an approach which integrates the application, artifacts, the CAD environment, and physical prototyping. A comprehensive program to integrate these four activities has been developed.

Since 1980 technology has been devoted to shrinking the size and weight of personal computers without substantially changing the way users interact with their computing environment. Conventional input/output devices place an ultimate limit on the size and weight of personal computers. Size is limited by the conventional typewriter-like keyboard whose dimensions have not changed substantially for over one hundred years. Both size and weight are limited by displays the size of notebook paper intended to be viewed from several feet away. Since the size of the display places a lower bound on the personal computer's energy consumption, weight is dictated primarily by the weight of the energy storage devices such as batteries.

Wearable computers deal in information rather than programs, becoming tools in the user's environment much like pencils or reference books. The wearable computer provides automatic, portable access to information. The information can be automatically accumulated by the system as the user interacts with and modifies the environment, thereby eliminating the costly and error-prone process of information acquisition.

Since wearable computers represent a new paradigm in computing, there is no consensus on the mechanical/software human computer interface or the capabilities of the electronics. Thus iterative design and user evaluation made possible by a rapid design/prototyping methodology is essential for quick definition of this new class of computers. This paper describes a rapid design/prototyping methodology that has produced four generations of wearable computers in 40 months.

The four generations of wearable computers produced by the Engineering Design Research Center (EDRC) are products of an evolving design methodology for coordinating the concurrent design activities of teams of students in electrical and computer engineering, industrial design, mechanical engineering, and computer science. Each computer could be considered a benchmark of the methodology. This paper will describe the evolution of the design methodology and the metrics used to compare the artifacts produced by the methodology at each stage of methodology's growth.

2 Design Methodology

The EDRC has defined a product cycle which ranges from perceived customer need to initial product design through detailed design to design of manufacturing, distribution, repair processes and eventually to product disposal, as depicted in Figure I. In the product cycle there are several entities to be designed. In addition to the artifact, the design/manufacturing/repair organizations as well as the design process at each step in the product life cycle need to be designed. Indeed, even tools which facilitate the entities in the product cycle have to be designed. At each step the objectives from the previous step are formulated into goals, constraints, and specifications for the

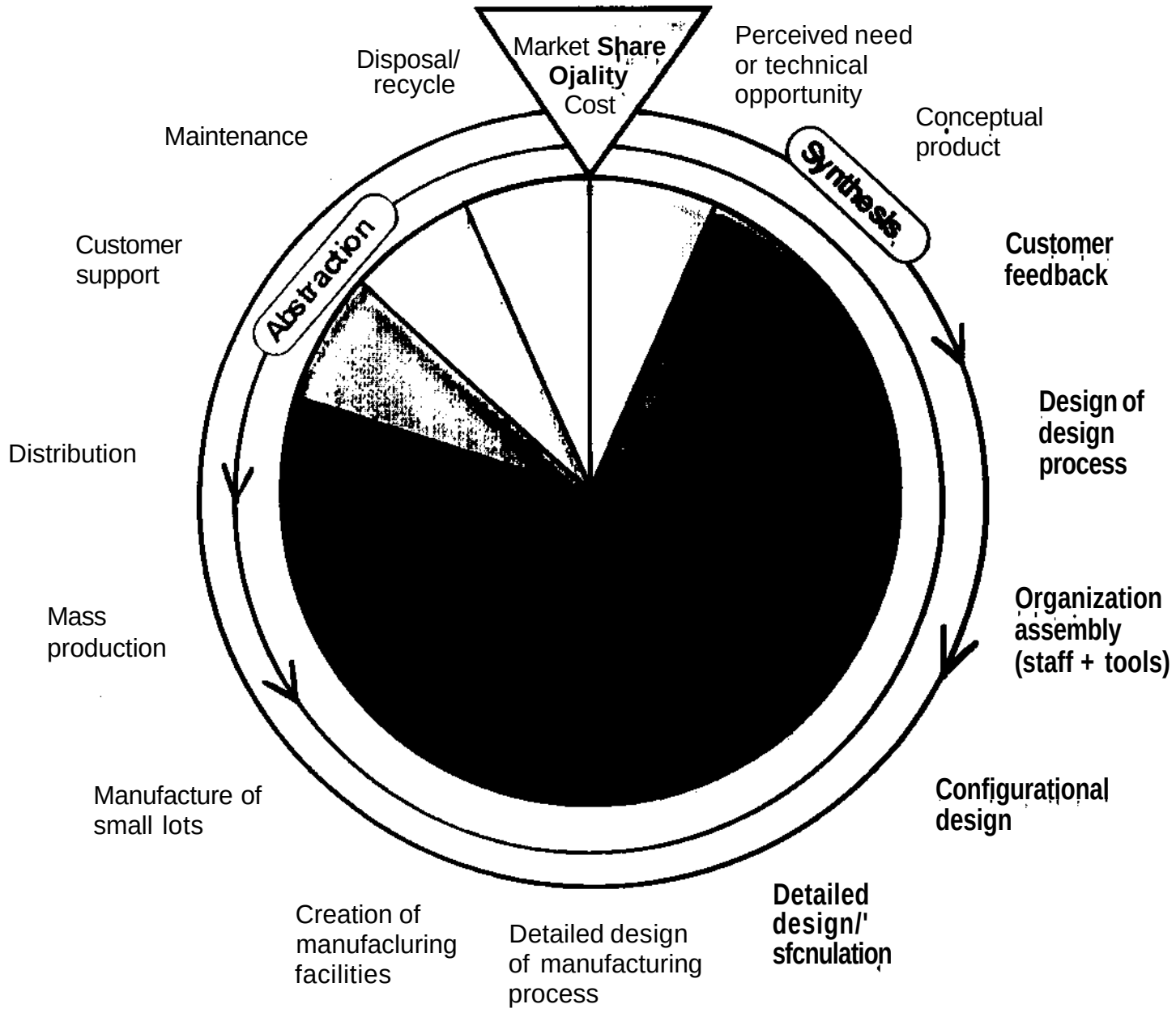


Figure 1. The Product Cycle

current step. Alternative designs are generated, analyzed and evaluated. If the design does not meet its specification, other alternatives are analyzed and evaluated. Successful designs represent opportunities for deriving rules to be applied to future design activities. To date most attention has been focused on the detailed design of the artifact as indicated by the density of shading in Figure 1. Methodologies and tools need to be developed to support concerns further upstream (e.g. conceptual product) and further downstream (e.g. manufacturing and disposal).

Throughout the design of the four generations of wearable computers, an interdisciplinary concurrent design methodology (1CDM) has evolved [3]. The goal of the design methodology is to allow as much concurrency as possible in the design process. Concurrency is sought in both time and resources. Time is divided into phases. Activities within a phase proceed in parallel, but are synchronized at phase boundaries. Resources consist of personnel, hardware platforms, and communications. Personnel resources are dynamically allocated to groups which focus on specific problems. Hardware development platforms include workstations for initial design, personal computers for development, and the final target system. Communications allow design groups and individuals to communicate between the synchronization points.

Table 1 depicts the evolution of the ICDM through four generations of wearable computers. The first column in Table 1 represents the steps in the product cycle from conception through manufacturing that deal with the artifact. As the methodology evolved through successive generations of more capable wearable systems, more subphases were added. The four original phases of the product cycle, the subphases that have been added to refine the original phases, and the design information representations will now be defined.

2.1 Conceptual Product.

During the conceptualization stage, the multidisciplinary design team establishes a common vision of the end product. This vision provides a consistent set of design goals for all disciplines to maintain throughout the product development cycle. Without a common understanding or vision between design groups and their members, each would be forced to rely on their own set of assumptions and criteria based on only a single view of the product.

As experience was gained with deployment of wearable computers, the key role of users became apparent. Thus the methodology evolved toward user centered design and human computer interaction experts were added to the design team.

- **Technology Survey.**

Since electronic technology changes so rapidly, what may have been infeasible only six months ago may now be possible. The initial subphase consisted of a technology survey in which alternative technologies are identified for each subsystem and evaluated in isolation from other subsystems. Alternative technologies are compared using a Model Feature Matrix based on information gathered mostly from product literature. The results of the Model Feature Matrices are used to identify primary and back up alternatives for each of the subsystems. Occasionally the feasibility of a technology decision is unknown but is pivotal to the entire design concept. For example, the replacement of the CGA adapter card in VuMan I by direct processor control on

VuMan 2 was critical to the goal of size reduction so a small wirewrap circuit was constructed to demonstrate feasibility. While the potential electronic technology is being identified, various mechanical forms are conceived and demonstrated physically (with styrofoam) or virtually (with a solid CAD model).

- **Problem Definition.**

The goal of the problem definition subphase is to define the problem which is being solved, perform requirements analysis, and evaluate user needs. A variety of brain-storming techniques is employed to develop a product design definition including attributes such as functionality, cost, performance, technology acquisition, and fabrication techniques. This subphase is critical for successful customer acceptance and was employed during the design of VuMan MA which is to be delivered as a vehicle maintenance assistant.

2.2 Configurational Design.

During the first two generations, the systems were derived directly from the feasible technology. Subsequently the product configuration became a critical phase.

- System **architecture** specification.

This subphase attempts to integrate the results of the technology survey phase to produce the first concept of the total system. Interactions and interfaces between subsystems are identified and inconsistencies between subsystem alternatives detected. Interactions between subsystems **are** summarized in a Design Dependency Matrix. These dependencies identified communication points for subsequent phases of the project. The primary technology alternative for each subsystem is refined to eliminate inconsistencies. Based on the refined set of technology alternatives, an architecture for the total system is specified.

- Subsystem specification.

Both firm and probable design decisions are identified and specifications are produced for each subsystem based on these design decisions. Interfaces to each subsystem are specified completely so that other subsystem designers can continue into the detailed design phase. Subsystem specifications are integrated into a complete Product Design Specification. This phase represents the last exploratory phase of the design process.

2.3 Detailed design.

The detailed design phase is traditionally defined by a well developed methodology and a rich set of CAD tools. A detailed design *ot* each subsystem is performed, with particular attention to maintaining the interface specifications as defined in the Product Design Specification. The technology selected for each subsystem is acquired and analyzed in terms of functionality and performance. Analysis of the technology may necessitate changes in the subsystem specifications. These changes are communicated to the relevant groups. Regular design reviews are held with

The Product Cycle	jVuMan 1	VuMan 2	Navigator	VuMan MA
Conceptual Product	• Technology Survey - mechanical form	• Technology Survey - technology assessment - feasibility study (including wire-wrap prototyping) - mechanical form	•Technology Survey •technology assessment - mechanical form - product feature matrix	• problem Definition - requirements analysis, user needs evaluation - product design definition • Technology Survey -technology assessment - mechanical form - product feature matrix
Configurational Design			• System Architecture Specification - dependency matrix • Subsystem Specification - product design specification	• System Architecture Specification - dependency matrix • Subsystem Specification • product design specification
Detailed Design	• Detailed Design - place and route - thermal analysis	• Detailed Design - place and route - protoboard providing geometric information • thermal analysis	•Detailed Design • place and route - protoboard providing geometric information -thermal analysis -taskdependency graph	• Detailed Design - place and route - mechanical/electronic/software mock-ups - thermal analysis - task dependency graph -user feedback
Manufacturing	• Implementation - printed circuit boards - vacuum formed housing • System Assembly (30 minutes/unit)	• Implementation - printed circuit boards - rapid prototyping of housing • System Integration (2 minutes/unit)	•Implementation -printedcircuit boards - pressure formed housing - evolving list of open issues • System Integration •Integration tree	• Implementation - printed circuit boards - rapid prototyping of housing - evolving list of open issues • System Integration - integration tree - test of the working system • Design Methodology Evaluation

Table 1. Product Cycle Steps and Corresponding Desjgp Methodology Phases

group liaisons, to ensure that interface specifications are not violated. Task Dependency Graphs are used to identify individual design decisions that affect more than one group. Thus design changes can be rapidly propagated. Initially traditional electronic (place and route) and mechanical (thermal analysis) CAD tools were employed. As the housing shape became more complex, mock-ups of the physical dimensions of the electronic board helped to reduce risk that the final housing and electronic board would not match. A mechanical mock-up of the proposed final form for VuMan MA was provided with a prototype electronic interface between the mock-up and the application software on the VuMan 2 hardware. The mock-up was used to gather early user feedback on the mechanical and software user interface.

2.4 Manufacturing.

The manufacturing phase uses a combination of electronic and mechanical job-shops and on-campus rapid prototyping facilities.

- **Implementation.**

After detailed design is completed, each group implements their subsystem using the acquired technology. As implementation progresses, the subsystems are demonstrated at various stages of development. A check list of open issues and action items is used to highlight remaining design dependencies and schedule their resolution.

- **System Integration.**

Each subsystem is individually tested and then integrated into the final system. An Integration Tree is used to sequence the merging of subsystems along the designed interfaces.

- **Methodology Evaluation.**

As a linai phase, the design methodology is quantitatively and qualitatively evaluated and modifications suggested.

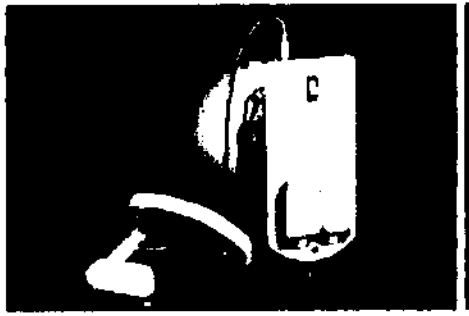
3 Four Generations of the CMU Wearable Computers

Each computer will be brieily described in order to provide a context for the complexity metrics introduced in the next section. Table 2 shows the characteristics and attributes of the CMU wearable computers.

VuMan I [I], conceived in 1991, allows a user to maneuver through the blueprints of a house using three buttons tor input, much like the mouse of a desktop computer. Output is provided on a commercially available head-mounted display, the Private Eye [4], which gives the illusion of viewing a personal computer screen from about five feet. The VuMan 1 's electronics includes an 8 MHz 80188 processor, 0.5 MB ot ROM, and a Private Eye controller board.

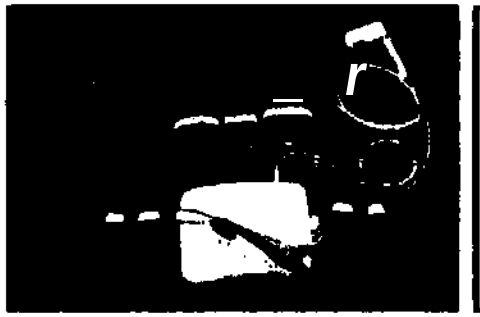
The original application of VuMan 2 [2], built in 1992, was to allow an user to navigate the

Characteristics and Attributes of EDRC Mobile Computers

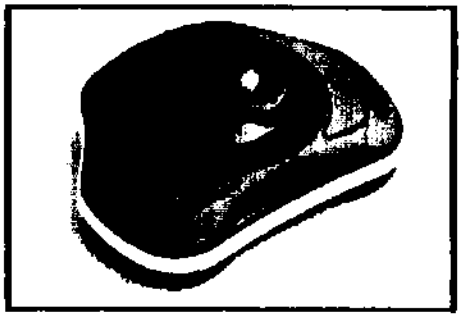


vū · man 1

vū·man 2



Navigator



vū · man 3

Artifact	vō·man 1	vOman 2	Navigator	v6man3
Delivery Date	Aug 91	Dec 92	June 93	Od94
Number oi units	30	7	3	20
Embedded/GP	embedded	embedded	general purpose	embedded
Design Style	semi-custom	fully-custom	design by composition	fully-custom
# oi custom boards/chip count	1/24	1/5	3/15	2/8
# oi off-the-shelf boards	1	0	5	0
lines oi code	1800	4700	38000	12000
Processor	80188 -8MHz	80CI88-13 MHz	80386-25 MHz	80386EX · 20 MHz
RAM/Nonvolatile storage	8KB · 512KB	512KB- 1MB	16MB - 85MB	2MB -40MB
Input	3-bullon	3-button	speech - mouse	rotary multi-position switch
Display resolution	720x280	720x280	720x280	720x280
Dimensions (inches)	2.5x5.5x12	1.5x4x4.4	3x8x10	2x4x5.8
Power (W)	38	1.1	7.5	2
Weight (lbs)	33	0.5	9	1.75

Carnegie Mellon campus. It has a database of buildings, departments, and people, so that a user unfamiliar with the campus can find the location of an appointment, get information on a faculty member such as a phone number or office number, or locate a specific building on the campus. Like VuMan 1, VuMan 2 uses the Private Eye for output and three button mouse for input. However, VuMan 2 is not dedicated to a single application. New applications are loaded via a Flash memory card. A second application developed for VuMan 2 is an electronic maintenance manual for a vehicle. An user can scan through manual pages and then access the corresponding diagram. Table 2 illustrates that VuMan 2, which has over twice the functionality of VuMan 1, has a factor of four reduction in volume, weight, and power consumption. The savings were a result of replacing the Private Eye controller board with a single programmable logic device, replacing the glue logic with a Programmable Logic Device (PLD), and replacing EPROMs with a Flash memory card.

Navigator 1 [3], built in 1993, is a general purpose computer consisting of off-the-shelf boards: an 386SX processor card, Private Eye interface, VGA controller, A/D card, GPS (global position sensing) card, modem, disk, and mouse controller. The custom boards in Navigator include the GPS interface, on-set of speech detection, and power control. The initial application was a campus navigation tool, similar to VuMan 2. Navigator can use speech as input, allowing completely hands-free operation. The Navigator's speech recognition system is speaker-independent [5], has a 200 word vocabulary, and currently runs at about eight times real time. A mouse is also available, in case that speaking is undesirable. Navigator runs the Mach operating system [6], allowing applications to be developed on a Unix workstation and then transferred to the Navigator platform. Software developers can use the standard Unix environment, such as X Windows [7] and Shell scripts, in their applications. Another difference in comparison to the VuMan embedded computers is that the Navigator architecture is modular, so that the hardware can be re-configured based upon the application. The Navigator modularity is composed of global position sensing and telecommunications (via a cellular phone and modem) subsystems. A study of the Navigator produced a set of techniques that reduced the power consumption of the off-the-shelf boards by 50% [8].

VuMan Maintenance Assistant (MA) incorporates new electronics, application software, and housing design to withstand shock, temperature, water, and dirt. The electronics includes new capabilities for input and power control. VuMan MA uses a combination of a rotary dial and a single selection ring that provides easy scrolling through menu options. The software application consists of a 600 item vehicle inspection check list. A sublist of possible status options for each inspection item allows the user to specify areas of the vehicle requiring repair. The software for the VuMan MA is based on the VuMan Hypertext Language (VHTL), using a forms based hypertext paradigm that provides quick access to maintenance manuals. The VHTL considerable simplifies the task of creating document systems that integrate forms, references (hyperlinks), images and complex control structures, such as nested menus. A link is provided between VuMan MA and a Logistical Maintenance Computer (LMC) so that results from vehicle inspection check lists can be uploaded for scheduling and planning. In addition to a Flash memory card, another PCMCIA device can be supported in a modular fashion, such as a radio. VuMan MA will be deployed for (ield trials in order to gather a detailed customer feedback.

The next section measures the impact of the evolving methodology on the design of wearable computers.

4 Benchmarking Design Methodologies

Design methodologies, much like new software and hardware tools, must be debugged and benchmarked, in order to compare and improve methodologies. Unlike an isolated tool, however, a methodology involves many variables, including the people making up the design team, the artifact being designed, and the tools used in the design. Isolating variables is difficult if **not** impossible. The large number of variables, along with the fact that a non-trivial design usually requires several months to execute, makes benchmarking a methodology costly, often involving thousands to tens of thousands of person-hours for each benchmark data point. Advances in technology and the improvements in the design methodologies make each design process, **and** hence the benchmark of its methodology, unique. The problem, then, is how to compare **these** unique design experiences. Metrics must be established that are appropriate to each design, **and that** capture those qualities of a design process which differentiate it from other, similar processes.

The EDRC has focused on four domains of artifacts: buildings, automotive subsystems, **chemical** processes and computers. Six generic axes of design activity have been defined:

- The number of designers
- The number of CAD tools used during the design, in each of the design domains
- Total effort to complete the design, in person-months
- The number of artifacts manufactured
- Whether the design is routine or innovative
- Complexity of the design

We will adopt these axes as metrics for the benchmarking of the methodology. All the metrics, except complexity, are straightforward to measure.

4.1 Design Complexity

Reducing all the aspects of a heterogeneous electronic/mechanical design to a single quantity is difficult. A metric of complexity of a wearable computer should factor in the complexity of the software, the electronics, and the mechanical design. The metric should reflect the complexity of both the functionality and the implementation of the functionality, and in the case of different implementations of the same functionality, should favor the implementation which is more technologically advanced. For example, if the electronics factor of complexity were simply the chip count, then an SSI circuit implementation of an adder would seem to be more complex than the same adder implemented with a single programmable logic device (PLD). Experience would lead to the opposite conclusion, so chip count by itself is not an appropriate metric.

The proposed metric is a product of the number of unique technologies in a discipline and a discipline specific measure of complexity. A standard unit of complexity for mechanical design is the number *of* features (i.e. rectangular solids, holes, cut-outs, etc.) and software design is the number of lines of code.

The measure of complexity chosen was

$$\textit{Mechanical complexity} \times \textit{Number of lines of code} \times \textit{Electronic complexity}$$

Mechanical complexity will be defined as equal to the product of the number of mechanical technologies and the number of features. The number of mechanical technologies is the number of distinct manufacturing techniques (such as assembly, material cutting, and molding) used in the production of the system. The number of lines of code is the amount of new code written for the system, measured in thousands.

The most involved portion of the metric is the last term. The electronic complexity will be defined as equal to the sum of the squares of the pin counts for all chips and boards in the design. Rent's Rule gives a relation between the number of gates and pins of an integrated circuit:

$$Pins = a \times Gates^{b}$$

Empirical results have shown that $a < 10$ and b is roughly 0.5 for VLSI circuits [10]. Since the relative value of complexity is more important than the absolute value when comparing systems, a is assumed to be 1 and b 0.5. An estimate of the gate count for a circuit implementing a given function can then be found by raising the pin count of each chip to the l/b power, and summing over all the chips. As was shown earlier, a simple chip count would lead to incorrect conclusions when comparing SSI and PLD implementations of a function. In addition, it was necessary to compare designs which employed only custom printed circuit boards with designs which used both off-the-shelf and custom printed circuit boards. While a standard board or VLSI chip does not have to be designed, it still needs to be understood by the designer so that it can be integrated with the rest of the design. Rent's Rule allows these dissimilar design styles to be compared, while capturing some of the effort expended in understanding how to design with off-the-shelf components.

The metric gives only a first order approximation of complexity, but the systems being compared are spread sufficiently in the design space that a first order approximation is all that is necessary. Had the designs been less distinct from one another, the metric might have needed to employ more complicated variables, a more involved combination of variables, or both. For instance, the metric does not distinguish between a line of assembly language code and a line of code in a high level language, such as C. Nevertheless, the metric chosen is useful for comparing the artifacts and the methodologies, as will be shown shortly. Table 3 gives the complexity of the four generations of wearable computers.

Generation	Mechanical Complexity (Tech. x Features)	Lines of code (Thousands)	Electronic Complexity	Total Design Complexity
VuMan 1	2x30	1.5	19,868	$10^{6.25}$
VuMan 2	3x182	4.5	20,784	$10^{7.71}$
Navigator	2x223	38	23,220	$10^{8.59}$
VuMan MA	4x385	12	59,628	$10^{9.04}$

Table 3. Complexity of Wearable Computers

Generation	No. of Designers	No. of CAD Tools	Design Complexity	Effort (p-m)	No. Fabricated	Design Activity
VuMan 1	**4**	**16**	$10^{6.25}$	**12**	**30**	**Innovative**
VuMan 2	**6**	**16**	$10^{7.71}$	**6**	**6**	**Innovative**
Navigator	**21**	7	$10^{8.59}$	**28**	**3**	**Innovative**
VuMan MA	**16**	7	$10^{9.04}$	**19**	**20**	**Innovative**

Table 4. Six Axes of Wearable **Computer Design Methodologies**

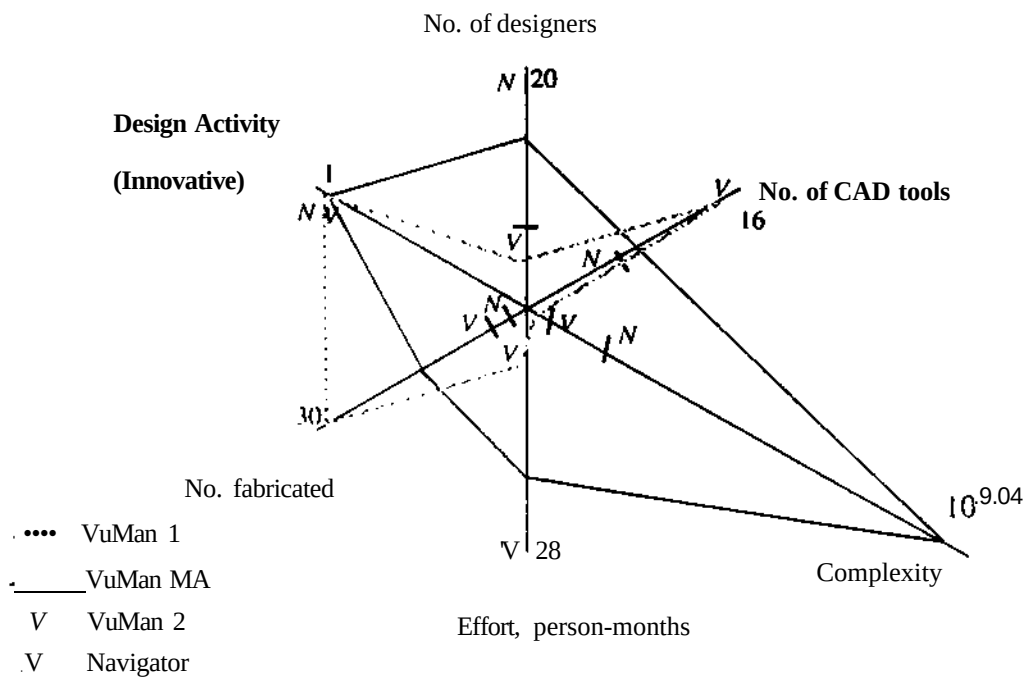


Figure 2. Comparison of Wearable Computer Design Methodologies

Table 4 shows the design methodology data for each generation, and Figure 2 shows the data plotted along the six axes *oi* design activity. The data for the first generation, VuMan 1, and the most recent generation, VuMan MA. are connected with lines to give the reader an impression of the change in shape as the artifacts have evolved. The data for the intermediate generations, VuMan 2 and Navigator, arc marked with a *V* and an *N.* respectively. Each generation tends to cover more area than us predecessors. The exception to this is along the axes of "Number of fabricated" and "Number ot CAD tools", where VuMan 1 has the largest values. A large number *o* the VuMan 1 units were produced because they were easier and cheaper to build than the later generations. In the case *oi* number *oi* CAD tools, VuMan I and VuMan 2 used more standard fabrication techniques and hence conventional electronic CAD tools could be employed. As the

designs began using more advanced technology, these conventional tools were no longer applicable. However as the design methodology stabilizes, the phases represent opportunities for new CAD tools.

4.2 Design Methodology Efficiency

For the purposes of this paper, efficiency of the design methodology is defined to be its complexity divided by the person-months of effort. One of the goals in developing the methodology has been to increase the efficiency of the design methodology, i.e. to use less effort to complete designs of similar complexity, or the same effort to complete designs of greater complexity. Figure 3 shows the efficiency of the design methodology of each of the wearable computers. It is evident that the design methodology efficiency increased with each generation, with the design methodology efficiency **of the last generation being two orders of magnitude greater than the efficiency of the first. Whether or not the design methodology efficiency will continue to increase for future artifacts is unforeseeable. One possibility is that the plot of efficiency versus time will be a series of plateaus. The design methodology efficiency might rise rapidly with the introduction of a new tool or technology, and then level off until the next innovation.**

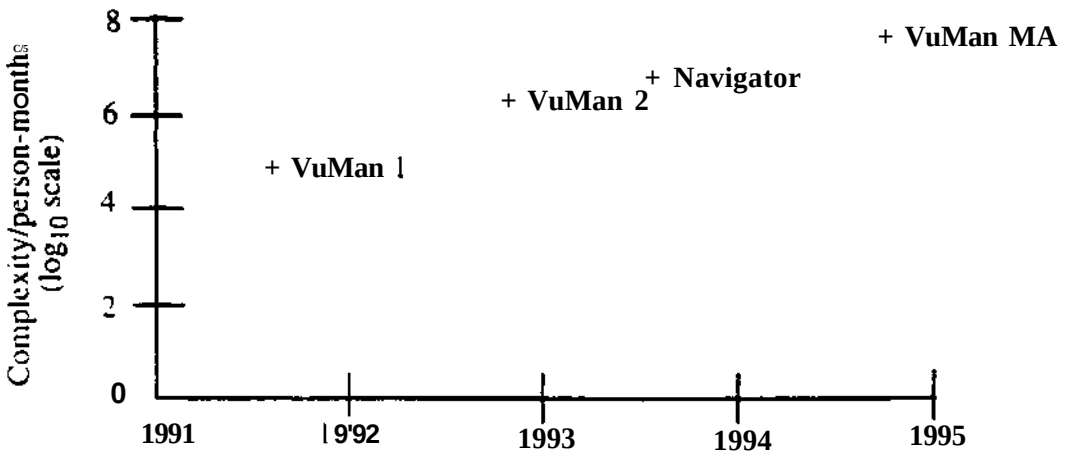


Figure 3. Design **efficiency for four artifacts**

Figure 4 shows the design methodology efficiency **of each wearable computer** design **versus the** number of designers. Although the small **number of data points limit the** conclusions **that can be** drawn from the figure, one can postulate **about the shape** of **the curve. For** example, one **would** expect that as the number of designers increases, **the** design **methodology** efficiency **would** initially increase as designers are allowed to focus on more manageable portions of the design (initial portion of Figure 4), and then as the **number increases further, the design methodology** efficiency decreases due to communication overhead **and** difficulty in **partitioning** the design (later portion of Figure 4) [9].

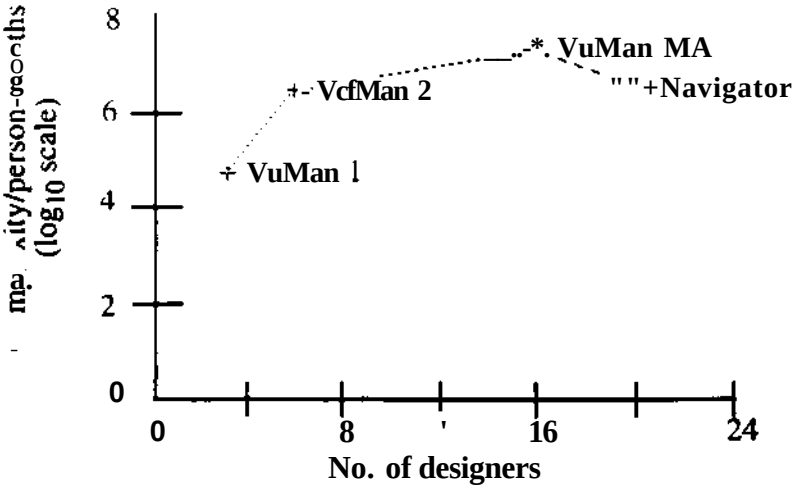


Figure 4. Efficiency vs. number of designers for four artifacts

The effect of the methodology on the artifact cannot be separated from the effect of the artifact on the methodology. All of the designs have involved multidisciplinary teams working concurrently on a wearable computer. Each team consisted of students from electrical engineering, computer science, and industrial design. If several types of artifacts had been designed, or if the teams had been composed of other disciplines, the **methodology might have evolved differently. Future work could include holding the methodology constant while applying it to artifacts other than wearable computers,** or **applying** several versions of **the methodology to artifacts of similar complexity.**

5 Conclusions

A multiphase design methodology for interdisciplinary design and implementation of **wearable** computer systems has been developed and design information representation defined **for each** phase. These phases represent opportunities **for new** generation CAD tools.

The concept of benchmarking complete design methodologies was introduced as well as a **proposal** for measuring design complexity. Once an acceptable complexity measure is defined, derivative metrics such as design methodology efficiency and design density (design complexity **per unit** weight or volume, assuming smaller physical designs are more difficult to achieve) can be explored. We hope that this paper will stimulate discussions on design methodology benchmarking and complexity metrics.

6 Bibliography

[1] Akella, J., Dutoit, A. and Siewiorek, D. P., *"Concurrent Engineering: A Prototyping Case Study,"* Proceedings of the 3rd IEEE International Workshop on Rapid System Prototyping Research Triangle Park, N. Carolina, June 1992.

[2] Smailagic, A., Siewiorek, D. P., "A *Case Study in Embedded Systems Design: The VuMan 2 Wearable Computer"*, IEEE Design and Test of Computers, Vol. 10, No. 3, pp. 56-67, September 1993.

[3] Siewiorek, D.P, Smailagic, A., Lee, J.C.Y., Tabatabai, A.R.A., *"Interdisciplinary Concurrent Design Methodology as Applied to the Navigator Wearable Computer System,"* Journal of Computer and Software Engineering, Vol. 2, No. 3, pp. 259-292, 1994.

[4] Becker, A., *"High Resolution Virtual Displays," Proc.* SPIE, Vol. 1664, Society of Photooptical Instrumentation Engineers, Bellingham, Wash., 1992.

[5] Li, K.F., Hon, H.W., Hwang, M.J., Reddy, R. *"The Sphinx Speech Recognition System,"* Proceeding of the IEEE ICASSP, Glasgow, UK, May 1989.

[6] Rashid, R. et al. *"Mach: A System Software Kernel,"* COMPCON Spring '89, San Francisco, CA, March 1989.

[7] MIT X Consortium, *X Window System, Version 11, Release* 5, MIT Laboratory of Computer Science, Cambridge, MA, 1990.

[8] Martin, T., *"Evaluation and Reduction of Power Consumption in the Navigator Wearable Computer,"* Masters Thesis, Report 1-18-94, EDRC, Carnegie Mellon University, July 1994.

[9] Brooks, F., *"The Mythical Man-Month: Essays on Software Engineering,"* Addison-Wesley Publishing Company, Reading, MA 1975.

[10] Preas, B., Lorenzetti, M., *"Physical Design Automation of VLSI Systems,"* Benjamin/ Cummings Publishing Company, Menlo Park, CA 1988.