1994

Creating an advanced collaborative open resource network

Susan Finger
Carnegie Mellon University

Carnegie Mellon University.Engineering Design Research Center.

Creating an Advanced Collaborative Open Resource Network

S. Finger, S. Konda, F. Prinz, D. Siewiorek,
E. Subrahmanian, M. Tenenbaum, J. Weber,
M. Cutkosky, L. Leifer, R. Bajcsy,
V. Koivenum, W. Birmingham

EDRC 05-80-94

Submitted to the Sixth International ASME Conference on
Design Theory and Methodology
Minneapolis, MN
September 11-14, 1994

Creating an Advanced Collaborative Open Resource Network

Susan Finger, Suresh Konda, Friedrich B. Prinz,
Daniel P. Siewiorek, Eswaran Subrahmanian
Engineering Design Research Center
Carnegie Mellon University
Pittsburgh PA 15213

Marty Tenenbaum
Enterprise Integration Technology
Palo Alto, CA 94301

Mark Cutkosky, Larry Leifer
Mechanical Engineering
Stanford University
Stanford, CA 94305

Ruzena Bajcsy
Computer & Information Science Department
University of Pennsylvania
Philadelphia, PA 19104

William Birmingham
Electrical Engineering and Computer Science Department
University of Michigan
Ann Arbor, MI 48109

Abstract

As part of of the ARPA MADE initiative, we are beginning to develop ACORN, an Advanced Collaborative Open Resource Network, ACORN will provide the infrastructure to create an electronic community which will be able to design and sell engineered products in competitive markets as well as conduct research and development by collaborating through a network. Creating such a community is a task of national proportions and cannot be accomplished by our group alone since the target community encompasses the entire country. We have an unprecedented opportunity to create and experiment with an electronic community which can serve as the model for a larger national community. In this paper, we outline the architecture for an information infrastructure to create and sustain such a community.

This paper is not a standard research paper; it is being published to invite the members of the community to participate in the evolution of the ideas expressed here and to encourage the shared development of the infrastructure necessary to create this network.

L Introduction

This paper describes an open network to provide resources to the design and manufacturing community. As part of the ARPA MADE initiative, we have been funded to begin developing the necessary infrastructure. The ACORN infrastructure must leverage off existing facilities such as the Internet and the research laboratories that have been developed within the community. Whenever possible, we will use tools and services that already exist commercially and in universities. Initially, the ACORN infrastructure will be created as a test bed for designing electro-mechanical devices.

A crucial part of creating the ACORN infrastructure is the development of the community that will use and expand the network. Such an electronic community can increase in scale only if the community shares its resources to build continually on the work of each participant, not only for the creation of engineered products but also for the infrastructure itself. The evolution of the infrastructure requires that as software modules arc created they become products and parts in themselves and be used to bootstrap and sustain the infrastructure. Thus the effort expended in the creation of the ACORN infrastructure is expected to produce results that are three fold:

1. a prototype electronic design and manufacturing community
2. a basis for the National Information Infrastructure and
3. a national repository in which engineering designs (including software) can be stored and retrieved for reuse.

The creation of such an electronic community in the market place requires a critical mass of participants representing the entire potential user community: industry, government, and universities. In the absence of such a critical mass, the test bed will not grow beyond the universities because the necessary diversity of offerings will be absent. In short, an economically viable electronic community can exist only if each participant perceives advantages to belonging to the community which outweigh the costs as seen from the parochial perspective of purely competitive advantage. The proposed work must therefore draw on a microcosm of the larger community. The lessons learned in its creation and sustenance will serve as the basis for the creation and maintenance of the community which, in turn, can serve as the basis for similar communities focused other interests.

The community we envisage can be composed only of voluntary participants. Creating an environment which facilitates such voluntary participation requires considerably more than just generating the necessary hardware and software elements. It requires the careful orchestration of evolving practices, conventions, and standards. These issues must be addressed by one or more bodies that concentrate on the issues that surround any voluntary entity: the basis for entry, the conventions and standards that arc required, the processes for establishing conventions and transforming them into standards, and the systems to ensure the safety, security, and confidentiality of individual interests without jeopardizing the interests of the community.

Our community is not monolithic; it is composed of business firms, universities, research establishments, the government, consultants, etc. Each group also has users with widely different technical skills, requirements, facilities, and needs. As such, ACORN must be flexible enough to accommodate the needs and constraints of each potential user. This flexibility requires that the entire effort be based on a compositional approach to systems building where each participant brings their services to both building and creating new applications and work-relationships. This demands that the application services should not be layered; layering would hinder the community's dynamic and ad-hoc composition of working design environments (combination of interaction modes, types of information exchanged etc.)

In order to understand the requirements that will drive the architecture of the system, it is important to recognize that ACORN is predicated on organizational and institutional solutions as well as software and hardware solutions. If all aspects are not addressed, ACORN will fail to achieve fully its intended goal of creating flexible design and manufacturing shops based on electronic commerce. Consequently, this technical rationale treats hardware, software, the architecture, the derived procedures, and organizational process all as products of proposed effort.

2. Design Scenario

To show the potential benefits of ACORN, we present three scenarios that involve solving the same design problem: first with a geographically co-located design team, then with a geographically distributed team without the ACORN infrastructure, and finally with a geographically distributed team with the ACORN infrastructure. The first two scenarios are examples of certain types of design situations as they occur today. The third is an example of how design might be done once the ACORN infrastructure is in place. Reference is made within the scenarios to software that is being developed by the members of the ACORN team. Although we do not describe the software in this paper, the functions of the software should be clear within the context of the scenario.

The first scenario is based on a class design project at CMU. The other two scenarios are variations on the first scenario.

2.1. Co-located Design Team

A team of six students from four disciplines is assigned the task of designing and building VuMan 2, a wearable computer with a heads-up display. The project is organized to achieve concurrency in time and resources. The students work together in a single room, so the class operates like an industrial skunk works. The students have access to the expertise and computational tools that are available on campus in much the same way that an industrial team would have access to in-house expertise and tools. Similarly, the students have been given a budget and can use local job shops for tasks such as PCB manufacture.

The goal of the VuMan 2 project is to create a prototype of a half pound, belt-mounted computer with a heads-up display. The students must design and build the hardware, software, and mechanical systems in four months. The design is based on a prior, heavier version with less functionality than the specifications for VuMan 2. Figure 1 illustrates the primary tasks and the interactions between the tasks as they evolve over time.

Together, the team lays out the expected tasks: create functional specifications for the hardware, software, and mechanical design, assess similar designs and available technologies, synthesize the electronics, design the software and user interface, and design the housing. Because the design team is co-located in close quarters, the members of the team can interact continuously. The need for interaction among subgroups representing various disciplines varies depending on the state of the design. In the beginning, discussions are frequent and intense. As the specifications for each subsystems becomes firmer, the subgroups tend to work among themselves. Each time the systems are integrated, interactions across groups again becomes frequent and intense.

Due to time pressure, the students often do not document or record their design decisions. This can lead to problems for absent or new members of the team who will need to be filled in on all the changes and decisions that have been made. Moreover, if design problems arise later in the process, not keeping track of prior negotiations and compromises for how and why certain design threads fit together may make it very difficult to fix problems without causing the threads to unravel. This is especially true when a design is sewn together by groups or team members from different disciplines.

Each group spends two or three weeks reading trade magazines and calling vendors to assess the price, capabilities, robustness, and compatibilities of available technologies. As the design progresses, the co-location of the design team allows some (but by no means all) potential problems to be resolved before they became inextricably embedded in a subsystem. For example, when a member of the industrial design team sees a 4" high backplane that the electronic designers are planning on using, he objects that

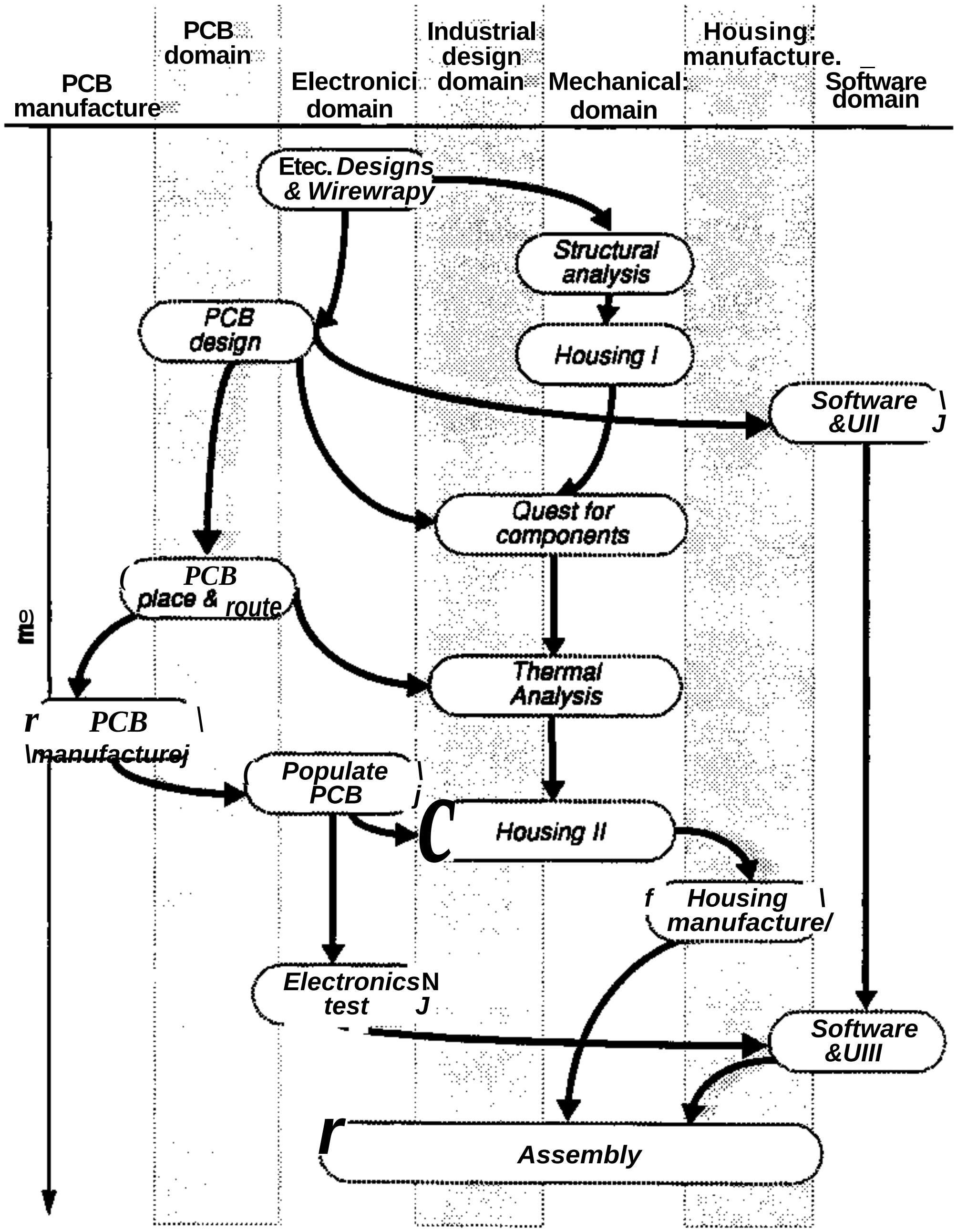

Figure 1: Design Tasks for VuMan 2: Co-located Team

such a thick belt-mounted module would interfere with the wearer's normal movements. The electronic design team explores alternate technologies and decides they can modify the design to use a flexible ribbon cable instead. The final belt-mounted module is less than two inches high.

Both the electronic and mechanical designers have problems with parts acquisition. For example, the electronic design team needs a dozen CUI stack 7-pin connectors, but the minimum order size is two hundred. They also decide to use a newly-available Flash EPROM Memory Card. They procure three of these cards to use in the initial prototypes, but then discover that, due to problems with a third party vendor, they will have to wait six months for delivery of any new cards. They have committed to deliver 10 prototypes of VuMan 2 to their sponsor within four months of the start date of the project. They again search the trade magazines and call vendors searching for an alternate technology or vendor, but arc unable to locate anything suitable. This is an important problem for skunk works projects — their connections to and negotiations with outside resources is often limited. The project team needs to be able to pull in outside resources and vendors to act as members of the team.

2.2. **Geographically-Distributed Design Team without ACORN**

An electronic design class is assigned the task of designing and building VuMan 2. The electronic design team creates the functional specifications for the electronics, synthesizes the design, and generates a PCB layout using the tools provided to the class on the local platform. The class consists only of electrical engineering students, so all other aspects of the design must be completed using outside job shops. Since the job shops arc geographically distributed, their interactions arc limited to telephones, fax machines and express mail. Having completed negotiations on the phone, they use a fax machine to send drawings of the rough shape of the PCB to an industrial design shop which has done similar work for the university before. Unfortunately there is limited communication between these two groups after this exchange. Starting from the rough shape, the industrial design team generates a shape for the housing and searches for the mechanical components such as switches, cables, LEDs, and connectors. They spend about two weeks acquiring parts. Once they have a preliminary design for the housing, they check its thermal performance since this was one of the primary concerns of the electronic design team. They do not have access to any thermal analysis programs, so they subcontract to a professor who uses the drawings to create a three-dimensional finite element model. Based on assumptions about the components on the PCB, he returns a report to the industrial design shop with some suggestions for changes. They modify the housing to improve the thermal performance. In the mean time, the design team searches for the electronic components to populate the PCB. They send the board to a subcontractor for manufacture. Within two weeks, the PCB is returned and tested. It is sent to a software shop which will write the embedded operating system and the user interface.

A few days later, the industrial design shop delivers a prototype of the housing which they have manufactured using a local machining job shop. Because of the height of the backplane, the PCB will not fit in the housing. In addition, the design team now begins to suspect problems with assembly. The housing for VuMan 1 required over a dozen screws for assembly and nine discrete wires that had to be cut, stripped, and soldered. The assembly of the new housing design is as complicated as the one for VuMan 1. The design team had neglected to include ease of assembly with the specifications that were sent to the industrial design shop.

The final PCB and the assembly requirements are sent to the industrial design shop. The head of the industrial design shop calls to say that they feel that the height of the new housing will make the product unacceptable to the marketplace. But so much time and investment has gone into the current design, the design team decides to proceed. They tell the industrial designers to do the best they can within the constraints they have been given.

Figure 2 illustrates how the tasks are assigned to the geographically dispersed participants. Because of the difficulty of communication, the boundaries between tasks are more sharply defined and the links between the tasks arc fewer than in the case of co-located design.

2.3. Geographically-Distributed Design Team using ACORN

Suppose ACORN is already in place and functioning. A marketing engineer at a large company has realized that there may be a market for a light-weight, voice-operated, wearable computer with a heads-up display. The company does not have the resources to devote to this problem, so the engineer uses a national referral network to obtain the required expert resources. The referral network is browsed by many seeking new insights into their own design problems. Students, in particular, find the repository useful because it gives them realistic industrial problems to work on.

An electrical engineering student from Stanford has registered an interest in wearable computers with the referral network. He is automatically notified as soon as the design challenge is posted. He sends e-mail to several of his friends at other universities to find out whether they would be interested in working on this challenge. As they discuss the requirements of this new computer, they think of more students from other disciplines and universities to be drawn in. A team coalesces that decides to take on the challenge of this design.

The first task for the new team is to negotiate a design process (e.g., procedures for handling engineering changes) and to create a shared design environment. A standard engineering process model is selected and edited to reflect the structure and organization of the current project team. The shared design environment includes each individuals preferred tools plus additional tools and services available through the ACORN infrastructure. It also includes the ACORN translators and wrappers necessary for the tools to inter-operate. The centerpiece of this customized design environment is a shared information repository that each team member can access using their favorite graphical interface (thanks to the modularity and standard protocols defined in the ACORN reference architecture.)

The team begins an intense series of discussions using multimedia mail, MovieMail, a groupware tool that records workstation sessions so they can be mailed and played back by colleagues, and X-Share, a real-time video conferencing environment that enables them to share tools and data.

Rapidly, they converge on the preliminary specifications for VuMan 2 and generate an initial design. With the preliminary specifications agreed upon, the team members form subgroups to concentrate on different aspects of the design. To evaluate the design, they quickly simulate the design using a specialized simulator created by combining a simulation engine from Cornell and a visualization module available in the ACORN repository.

The groups that form are not limited by the geographic locations of the members. Each student becomes a part of one or more of the groups focussing on electronic, mechanical, and software design. The history of their conversations is captured on-line, and team members often replay earlier discussions as the design progresses. The availability of these comprehensive design records enables students to move from team to team as resource requirements change.

Engineers from both the electronic and mechanical design teams use the Information Navigator, a search environment that combines hypermedia, navigation, content-based retrieval, and formal querying to explore the hundreds of on-line design libraries, engineering handbooks, and part catalogs available on the Internet. They quickly locate similar designs and find out what parts are available. The technology assessment task that once took weeks, can now be done in hours.

Whenever they find a particularly interesting design or component, they copy the part into the team design notebook for everyone to see. They annotate the designs by adding hypertext links to relevant specifications, data, and analyses. Using PartNet, an electronic parts ordering network, they can post complete models of the parts including complete models of the parts including specifications, geometric

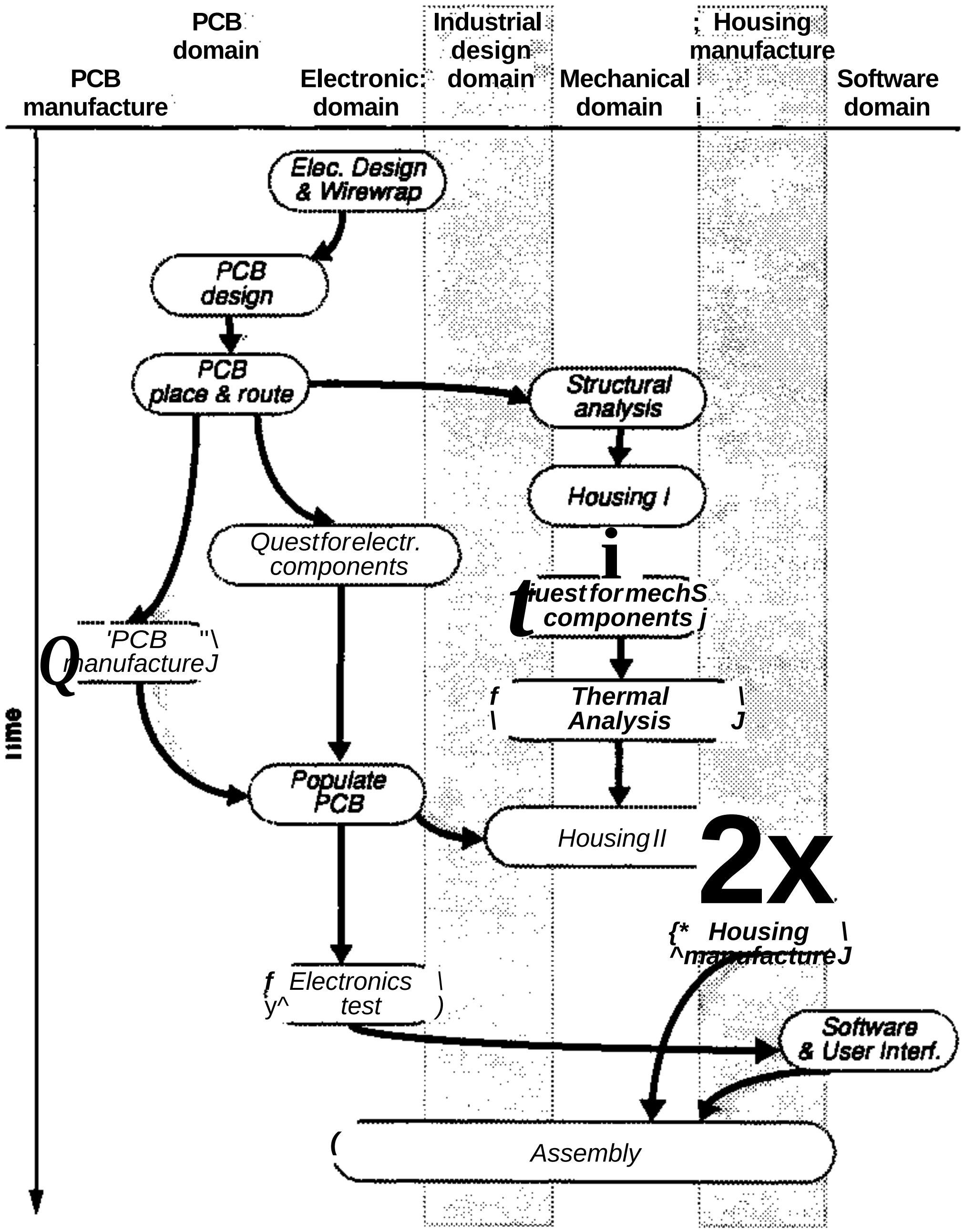

Figure 2: Design Tasks for VuMan 2: Geographically-Distributed

models, and simulations. The students use X-Share and MovieMail to show other members of their team what they are working on.

Using brokering services provided over the Internet, they acquire the electronic and mechanical parts they need in the right lot sizes. The parts arrive via express mail within two days at the Stanford lab where the first prototype will be assembled. Even though most of the students never see the parts, because of the visualization features available through the shared design notebook, problems like the 4" high back plane are noted and resolved before they become serious. In addition, as the team works, all their design decisions are captured in their notebooks.

None of the students is an expert in thermal analysis, but the team thinks they may have a problem with overheating. They return to the referral network to locate a registered expert in thermal analysis. They transmit the design to the expert using e-mail, who forwards it to a third party translation service that returns the format required by thermal analysis tools. The expert then runs the analysis, interprets the results, and within hours, sends back an animated simulation showing heating patterns, along with suggestions for design changes to alleviate hot spots. All of the simulations, discussions, decisions, etc. can be saved and organized seamlessly as the design progresses providing a design history that can be elaborated into design documentation.

An industrial design student at Berkeley has an inspiration that the weight and size of VuMan 2 could be reduced if only the shape of the PCB could conform to a person's body rather than being flat. She sends the specifications and drawings over the network to a broker, who forwards them to several dozen innovative fabrication services, with a request for quotes. She receives several responses, including an attractive bid from the MD* group at CMU, which requires certain design changes to be made. The team decides to modify the design to take advantage of the capabilities of the MD* process. However, since they are close to their deadline, they do not have time to learn a suite of new tools. They contract through the broker with a specialist performs who fine tunes the design, performs the required translation between their CAD system and MD*'s requirements, and then forwards it to the MD* manufacturing cell. The team receives the initial prototype the next afternoon by express mail.

3. Implications of the Scenario

This section based on the scenarios described draws out some of the needs, problems, and possibilities that must be considered in formulating requirements for the ACORN infrastructure.

The skunk works scenario illustrates a number of problems and implied needs that the ACORN infrastructure should address:

- provide access to external sources (other disciplines, different organizations) through the same infrastructure
- maintain organized records of the design history

The second scenario demonstrates the need for distributed asynchronous collaborations and interactions among designers.

The third scenario illustrate the services that could be available on ACORN. The following list of needs are extracted from this scenario:

- to access and browse wide-area information repositories, on-line catalogues for mechanical, electronic, software building blocks, and other information bases.
- for design environments that can be custom tailored by the designers themselves as their work and understanding evolves.
- for translators and mediators that ease inter-operability of tools and services so that users do not need to be concerned about application interfaces or internal representations.
- for brokering services that enable users to request and tender bids and generally to engage in

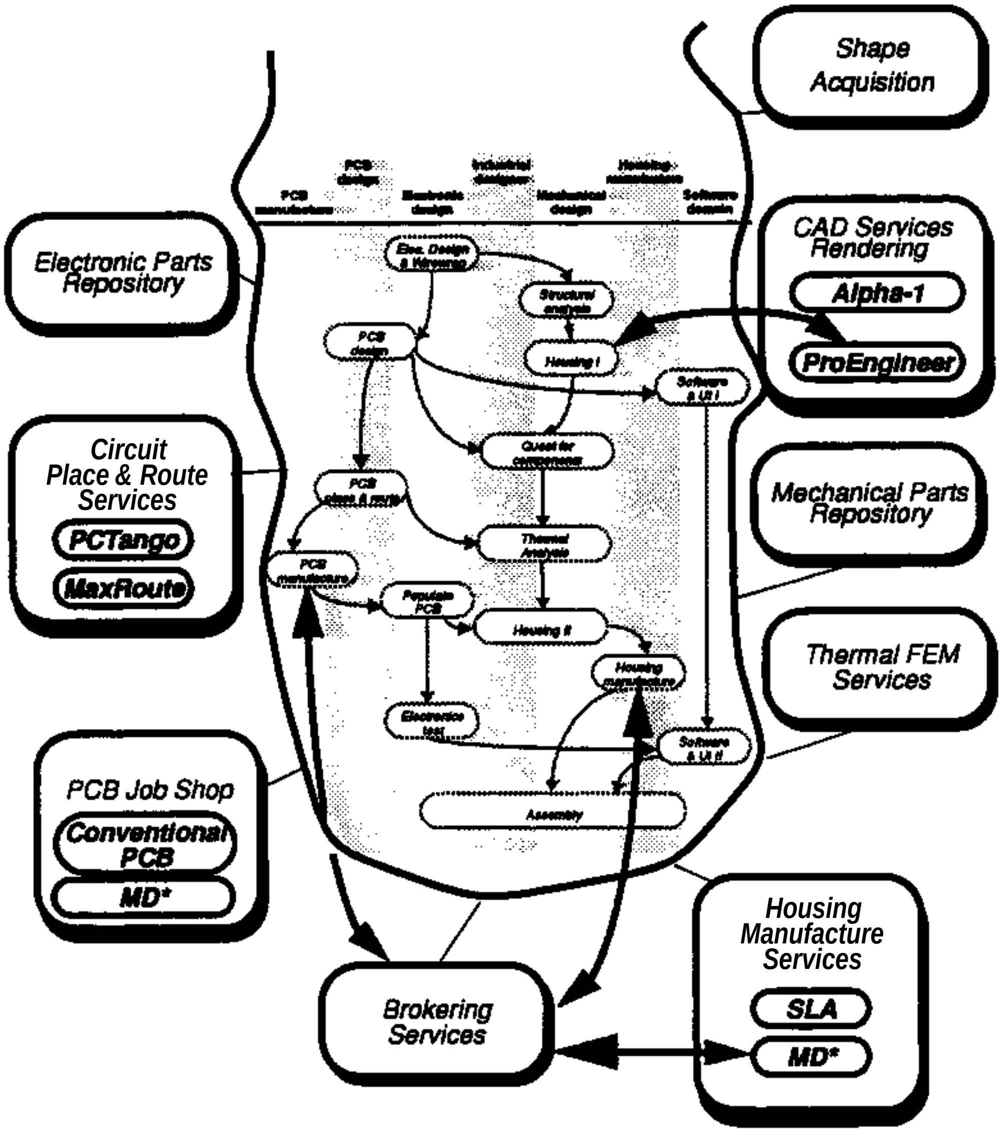

Figure 3: Design Tasks for VuMan 2: Geographically Distributed

electronic commerce.

- for collaboration services that enable product teams to interact at different levels.
- for sharing information and tools both within and outside the group through applications like creating design notebooks, and services which provide access to design history and replaying of design decisions.

4. ACORN Scope and Requirements

The scenarios illustrate some of the possibilities of an ACORN network. Here we elaborate on the scope and requirements of the netwoik.

The architecture of the network should not restrict the ways in which it is used by the members of the community. However, users must have incentives to participate in ways that also benefit the entire community. The community represents a collection of firms, government agencies, and universities, all of varying sizes. As a reflection of these communities, the electronic community would need an infrastructure to support the modes and intensity of interactions that are manifest in current day face-to-face paper and telephone based interactions.

The interactions that commonly exist vary from those with narrow, pre-defined conventions, such as searching an on-line catalogue, to those with open and continuous interactions, such as occur at a design review. Nevertheless, in any engineering firm, combinations of relationships are used in creating an organization to design a product. This organization itself is subject to change, for example, a manufacturer may change suppliers or even supplier relationships. In all of these relationships, the participants negotiate the level of information exchange and rate of interactions subject to constraints on time, resources, and long and short term objectives. Building the ACORN infrastructure itself will require such flexibility of interactions. Consequently, ACORN will be used to build ACORN.

The development of ACORN will include the creation of work groups to build products in electro-mechanical designs, wire harness design, and injection molding tools. The experiences of these work groups will allow us to: define some possible models of interactions, study creation and use of tools, and observe the establishing of conventions for services such as tool inter-operability. In addition to these work groups, more loosely organized structures, such as bulletin boards, will allow the participants to interact with one another. Part of the proposal is to identify how these groups collaborate and to implement the community's protocols as part of the infrastructure. Explicit experiments will be conducted, involving multi-organization design teams composed of ACORN experts from both industry and academe.

The ACORN requirements are a complex function of a number of interrelated factors. These factors include the bandwidth and levels of integration at which the members of the community interact, their differing models of design organizations, the processes by which they compose their working relationships, and the nature and scale of the tools and services that are provided.

The scope of ACORN requires that every application is a potential building block for another application or service. These building blocks can be organized in multiple ways since they are designed for inter-operational use with no pre-determined organization using flexible building blocks to create ad hoc design environments. As applications are created they will become grouped according to use, quality, and other dimensions that are important to the user community. These groupings serve as models and prototypes in that they show ways the building blocks can be connected. Once initial groupings are shown to be achievable and viable, new groupings can be imagined and developed based on previous results and methods. Applications and services are not just simply re-used in old ways, but re-used creatively and according to tested precedents that need to be documented. The creation of on-line communities in ACORN will allow us to discover new compositions of work and worker relationships heretofore not possible or difficult to achieve.

In addition to the needs that were described in the earlier paragraphs, ACORN must have the following technical and operational features:

- be an open network based on Internet standards and published interfaces with an ongoing standards process to coordinate contributions from multiple providers.
- allow users to disseminate their applications through ACORN as building blocks for others to build on.
- provide wrappers and translators so that application providers can easily plug in new applications and services without knowing minutia of Internet protocols.
- provide standard environments so that novices can easily use ACORN services.
- be scalable to hundreds of thousands of services, through directories, brokers, and other referral services,

5. ACORN Architecture

There are three layers in the ACORN architecture as illustrated in Figure 4. They are the applications layer, the services layer and the network/platform layer. The proposed work is directed toward creating the applications and services layers. The difference between these two layers is not based so much on which applications and services are indigenous to which layer but rather on the functional roles of the applications and services. The application layer supports the access of, distribution to, and use of applications by consumers and users. It is from the latter's perspective that services in the application layer are considered for inclusion in ACORN and evaluated. The services layer supports the design, composition, production and maintenance of services by producers and providers - the latter being intermediaries through whom some producers sell and install their services. It is from the producers' and providers' point of view that tools in the services layer arc considered and evaluated, ACORN, therefore, includes customers of services, providers of services, and intermediaries. Institutionally an ACORN member can play any or all of these roles in different working relationships. For the community to sustain itself and grow, the three categories of users will have to collaborate in contributing to the network, not just services, but also through involvement in the standards setting process and through reporting and exchanging experiences in using the services.

Just as ACORN requires a critical mass of participants, it also requires a critical mass of infrastructure, models, and prototypes. This includes not only the hardware and software of the network/platform layer, but also the tools to support producers and providers in composing and disseminating new services and applications as well as socioeconomic models and prototypes to support the use and evaluation of these new services and applications. Hence, both the applications and services layers of ACORN will be seeded appropriately.

The architectural layers should not limit what ACORN will include nor how applications and services in it will be organized and extended. Rather, the ACORN architecture is being designed to enable users and producers to compose and arrange tools and services in any way that is useful to them. It specifies the range of functional roles that are, from the current vantage point, plausible and indicative of the kind of opportunities ACORN will provide.

5.1. Applications and Environments Layer

The applications layer of the architecture focuses on applications and services from the point of view of how they are consumed and used. It populates ACORN with models, prototypes, and seed applications and services that provide a starting point for gaining experience on how ACORN will be used and a continuing basis for evaluating the needs of consumers and users. What follows is a preliminary grouping of the functional roles of the applications layer into collaboration, engineering, and information services.

Collaboration services in ACORN are provided through process models and environments. These are by

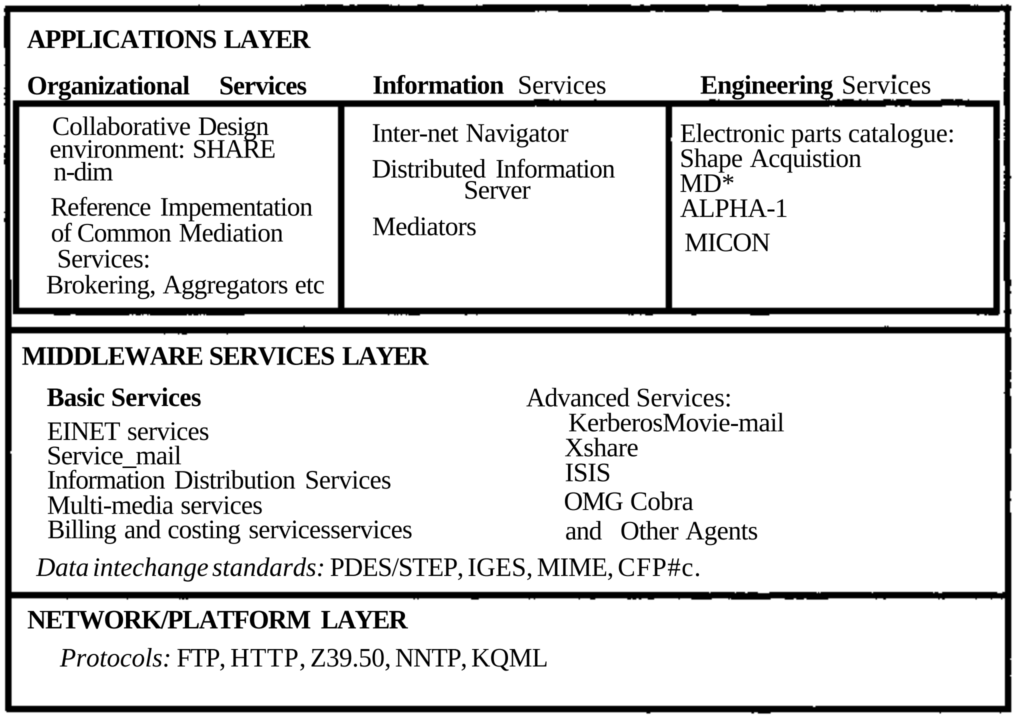

Figure 4: Reference Architecture Model

no means exhaustive since services dealing with, say, electronic commerce might also prove useful for collaboration.

Process models model interactions and coordination among members of a product development team. Examples of process models include:

- procurement models that deal with issuance of requests for quotations, purchase orders, and purchase requests.
- engineering models that address maintenance of schedules, commitments, engineering change orders, and other sign-off procedures.
- manufacturing models for tracking resources, work in progress, manufacturing process plans, etc.
- software development models for locating reusable modules, integrating, testing, and documenting them, and ultimately publishing interface specifications for the resulting module so that others may reuse it

Environments embody particular work relationships among members of the community. ACORN will provide standard environments for teams working at different levels of integration, ranging from a tightly-knot collaborative design team to a firm using ACORN for electronic commerce. All such environments can be custom tailored by the team.

Information services enable users to browse, retrieve, and organize data from the thousands of disparate sources available on the Internet. These include directories of resources, handbooks, manuals, repositories of previous designs, and so forth.

Engineering services include applications such as analysis and simulation tools, distributed design libraries, fabrication services, and services for scheduling resource allocation and planning. They also include brokering services, such as a mechanical MOSIS, analogous to the VLSI manufacturing service.

It should be noted that some functional roles are not easy to classify as collaboration, information, or engineering because they overlap two or perhaps all three groups. For example, catalog search, parts ordering, brokering, and advertising are services that overlap collaboration, information and engineering.

5.2. Services Layer

The services layer supports producers and providers with tools, methods, and models for creating applications and services. ACORN toolkits and services will make it easy for producers and providers to install their design, engineering, and prototyping applications as ACORN services. In addition, the services layer provides building blocks for the creation of the applications layer. It is the responsibility of ACORN developers to put enough tools, infrastructure, and building blocks in place so that applications producers and providers can effectively begin their work. Some of these will be found useful and maintained. Others will not be found useful and will be replaced or discarded.

Examples of infrastructure services, tools, and building blocks include directories, security, and financial services, toolkits for groupware, conventional and multi-media e-mail services, and other toolkits for creating distributed object management. Some of this work will be done by the service producers themselves, but some will be done by ACORN developers. For example, today's Internet is used largely for basic services such as e-mail, FTP, telnet, and news. These services must be enhanced by ACORN developers to support engineering, manufacturing, and electronic commerce applications. The enhancements include directories of people and services, privacy-enhanced, multimedia e-mail, authentication and access control, format translation, and payment facilities.

In addition to these basic tools in the services layer, a set of network services will be provided. These include distributed object management, information sharing, real-time collaboration, and software inter-operation services.

5.3. Network/Platform Layer

The third and bottom layer is the network/platform layer serving as support for the services layer. The ACORN effort will be based on the Internet backbone. We will use existing services and protocols to interface with the Internet (such as MIME, HTML, Z39.50, NNTP, KQML etc.). However, there will be cases where ACORN developers will have to extend these protocols, for example to support real-time engineering collaboration. Such extensions will be made available to the entire Internet community through the normal RFC process.

5.4. ACORN Implementation

The initial ACORN infrastructure will be built using EIT's ServiceMail because it provides virtually universal access, can contain multiple forms of data and, with enhancements, can protect sensitive data. Further, its distributed client/server architecture is necessary for scalability.

Later versions of ACORN will be implemented as a federated system using agents interacting as peers over the Internet. (See Figure 4). Each agent can represent one or more of the following: a designer with personal CAD tools, a database or other information service, a computational service that supports engineering or the engineering process. The messages are sent using standard e-mail and TCP/IP transport services.

The focus is on manufacturing products that are combinations of electrical and mechanical components. However, software engineering practice must be an integral part of the creation of the manufacturing community not only because software has become another component in electro-mechanical parts, but also because software is key to the creation of the parts themselves. ACORN has the potential to revolutionize the software development process for engineering applications, and for that reason, ACORN will be used extensively in the development of ACORN itself.

The software engineering community has recognized the need for re-use of software process models, applications, objects and services. The object-oriented community has evolved around the issues of inter-operability, extensibility and re-use to avoid re-implementations (coding, testing and documentation). The key to the promotion of re-use is clear specification of standard interfaces and message protocols augmented with libraries of re-usable objects. ACORN will implement a software development process that promotes the wholesale reuse of applications and services on a national scale over the Internet.

6. Conclusion

Creating an information infrastructure is not just building an efficient distribution system; it involves creating an electronic community that allows its members to build on each others work. The ACORN team will identify, catalogue, and publish plug-compatible building blocks (clients, servers, mediators) for others to use. The catalogue will contain test information on the objects and their use in different contexts. This will allow the community to build, integrate and test new functional blocks from well-tested components. The modules themselves may be bought, bartered, or freely distributed.