INTELLIGENT NETWORK SUPPORT SYSTEMS

Brian R. Gaines

Knowledge Science Institute, University of Calgary
Calgary, Alberta, CANADA T2N 1N4
gaines@calgary.cdn

ABSTRACT

This paper describes the development of an intelligent network support system in which an object-oriented knowledge base is used to model a heterogeneous network. The system is designed to be effective with a minimal knowledge base of: the processors and peripherals on the network; software packages and where they can be run; users and their normal activities; and active processes and current loading. The object structure allows this knowledge to be expanded to greater levels of detail in particular areas as required. The system is designed to support users of varying levels of skill and requirements to make more effective use of the network computing resources by advising them on software, processor and processing availability. It is also designed to support the network manager in resource planning, security violation detection and load balancing. The paper gives an overall framework for the design, develops knowledge structures for modeling heterogeneous networks, and illustrates our approaches to critical aspects of the system design.

INTRODUCTION

We have entered an era of *system integration* where the dominant theme in information systems development is the integration of diverse technologies into unified systems. In the late 1960s suppliers of computer-based information systems found it necessary to form separate system integration divisions to cope with the complexities of hardware and software interaction and integration. The ad hoc activities of such divisions have grown to become major cost centers, and it is no longer possible to treat system integration as a fairly informal customization stage in system production. This has focused developmental effort on system integration as a major technology in its own right. In the 1980s integration is the primary developmental thrust in all areas, and at all levels, of information systems:

- For personal computer applications it is the integration of spread-sheets, word processors, databases and telecommunications in packages such as *Symphony* and *Framework*, or through environments such as that of the Macintosh and *Windows* (Gordon 1984);
- For data processing it is the integration of all records within an organization into a uniformly structured database at the core of all applications (Nolan 1983);
- For network technology it is the integration of diverse services on a single broad-band carrier, such as a fibre-optic link, to provide integrated service digital networks (Bartee 1986);
- For network applications it is the integration of diverse multiple sub-systems within a single networking framework through protocols such as MAP (General Motors 1984), TOP (Boeing 1986) and MIB (Leopold 1986);
- For frontiers materials research it is the integration of the information processing architectures of digital computers with the high-density, high-speed capabilities of bio-chips (Jarvis 1985) and optical processing (Miller 1986);

- For frontiers computer research it is the integration of the fourth generation technologies of databases, distributed systems and supercomputers, together with the fifth generation technologies of pattern recognition, artificial intelligence and expert systems, to provide a new generation of knowledge-based systems (Moto-oka 1982, Gaines 1984);
- For frontiers information science research it is the integration of the multiple disciplines of knowledge processing—philosophy, neurology, psychology, linguistics, anthropology and sociology—to provide new foundations for knowledge science (Gaines 1986b).

As usual in the development of new technologies, however, the thrust towards integrated systems, while solving old problems also creates new ones:

- First, the overall complexity of the total system after integration can be very high. This is, of course, the objective—to design, develop and implement complex systems by integrating a number of simpler ones. Success in such an approach to system design is critically dependent on decoupling the system layers—only the functionality of the lower levels must be relevant to the upper levels—consideration of their implementation and operation should be irrelevant to the system design. However, this decoupling is not easily achieved—it becomes the major, and most problematic, design objective.
- Second, the problems of designing complex systems are exacerbated if the components include not only hardware and software but also people. The "liveware" component may be designed in part by selection and training but there remains a massive residual uncertainty about its functionality and performance that conventionally requires human "management." The integration of people and the associated management processes into large-scale integrated systems presents major new problems. In particular, the management structures already in place are unlikely to be appropriate to the overall integrated system. Major changes in organizational structure and corporate culture will generally be necessary to the effective implementation and operation of large-scale integrated systems.
- Third, the classic approach to the design of complex systems is based on homogeneous system concepts—that the overall system is designed as a set of related components developed with mutual interaction in mind. Heterogeneity may be introduced in the design of such systems, but its introduction is a design decision made in the light of overall system functioning, not a result of the fairly anarchical evolution of systems not originally intended to be integrated.

Fortunately, as often happens, the advances in technology that have created these problems also bring with them the means of solution (Gaines 1988a). Knowledge-based systems, in particular, provide a technology suited to modeling the diversity and complexity of integrated heterogeneous systems. The model can encompass as much detail as required to capture the important features of the system. The operational form of the model makes it possible to simulate system behavior at various levels of detail. The high-level representation makes it possible to use the model to aid designers and advise users. There are major problems involved in keeping the model up-to-date which place important constraints on the implementation and application of such knowledge-based systems. However, these constraints are intrinsic to the operation of complex integrated heterogeneous systems, and the knowledge-based modeling approach throws light on these fundamental problems.

The project described in this paper is targeted on the development of knowledge-based network support systems in which an object-oriented knowledge base is used to model a heterogeneous system. The support system is designed to be effective with a minimal knowledge base of: the

processors and peripherals on the network; software packages and where they can be run; users and their normal activities; and active processes and current loading. The object structure allows this knowledge to be expanded to greater levels of detail in particular areas as required. The system is designed to support users of varying levels of skill and requirements to make more effective use of the network computing resources by advising them on software, processor and processing availability. It is also designed to support the network manager in resource planning, security violation detection and load balancing.

The concepts underlying the intelligent network support system design are: that much can be achieved through the availability of small amount of processible knowledge *about* the network; that it is better to keep a small knowledge base correct than to provide a large knowledge base that is out of date; and that there will be a heterogeneity of knowledge requirements and the capabilities to keep the knowledge base up to date. The initial implementation is in Smalltalk on a network of some 30 workstations under NCS, including Sun, Apollo, Macintosh, Iris, and Symbolics, and access to Vax, Butterfly, Multics and Cyber 205 mainframes. These systems serve a community of research groups in vlsi, ai, graphics, simulation and networking. The overall design is targeted on wider communities without specialist computing knowledge using heterogeneous networks for purposes such as computer-integrated manufacturing, dispersed process instrumentation and control, and collaborative research.

INTEGRATED HETEROGENEOUS SYSTEMS

Figure 1 shows the information systems in a typical integrated heterogeneous system, such as a banking system using ATMs or a modern production unit using computer integrated manufacturing techniques. The outer ring comprises eight basic information technologies that together support the organization and its interface to the physical and organizational worlds. The center shows the organization itself with its corporate structure, and the rings surrounding it show the human interfaces and communication protocols necessary for the systems' management and integrated operation. The figure has been simplified to the minimum structure necessary to support an effective integrated system, and the residual complexity shows the need to consider the total context of the information technology involved. Such a system will not necessarily succeed if any one component, or even several, are advanced state-of-the-art technologies. However, it will probably fail if one component is inadequate or the total system is not well-integrated.

The Organization

The central core is the people that manage and operate the system, their organization and value systems. The overt operation of the system is part of the *technical culture* of the organization in Hall's (1959) terms. It can be communicated, documented and partially programmed. It can also be changed fairly readily in response to new requirements and technologies. The management structure of the organization is part of its *formal culture*. Only part of it is overt and documented and it is resistant to change. What has come to be called the "corporate culture" (Deal & Kennedy 1982) is in Hall's terms the *informal culture* of the organization. It pervades the organization, is communicated by example, undocumented, and changes only slowly or under pressures of survival. The managerial structures and social processes of the organization are not normally treated as part of its information technology. However, to be successful the technical systems must be consistent with the organizational structures and processes. This

applies particularly to inter-personal communications where computer networks will not be used if they violate the managerial and social norms of the organization, and to command and control paths which have to conform to existing authority and procedures that may not be those of the organization chart.

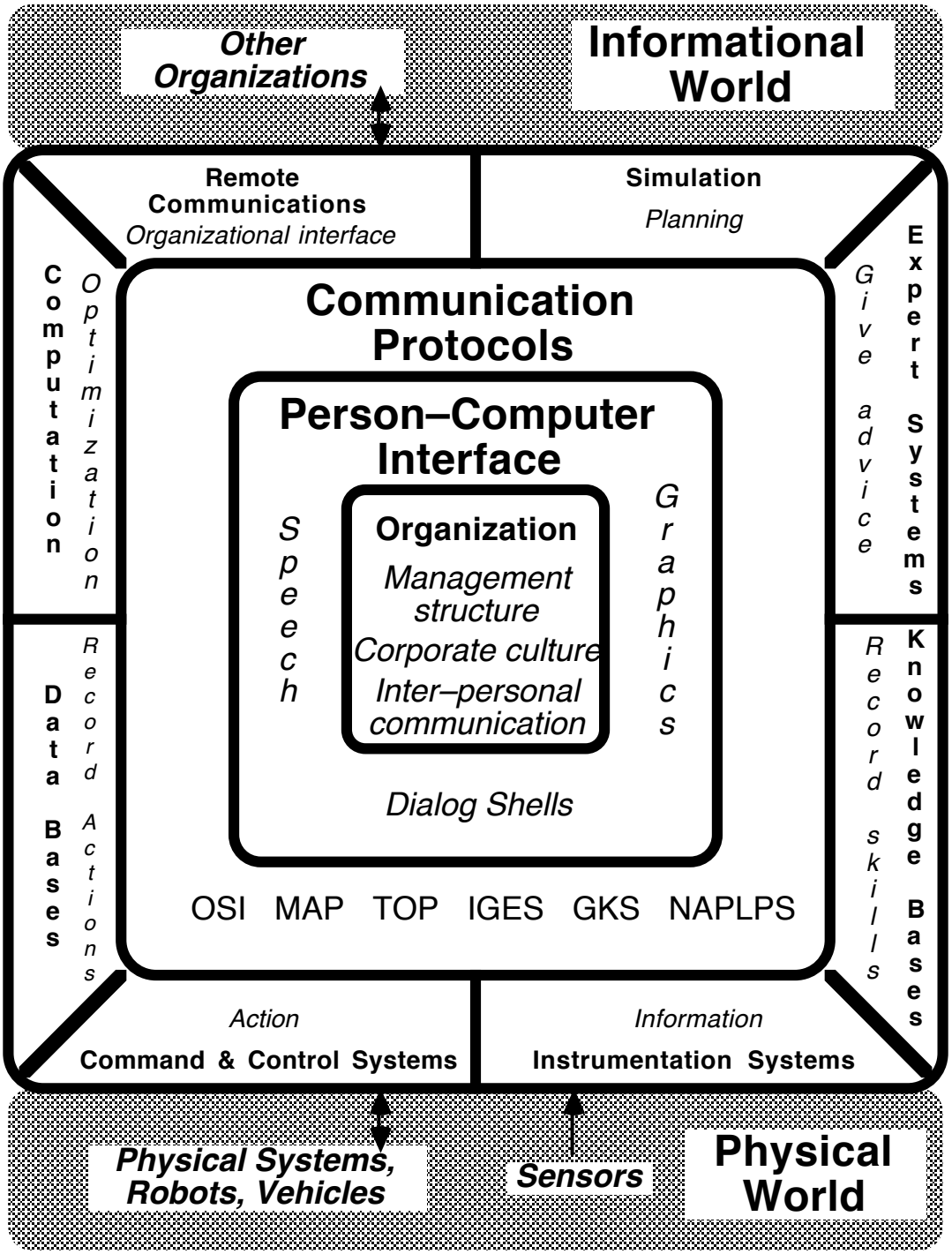

Fig.1 Structure of an integrated heterogeneous system

Person-Computer Interface

Recognition that the managerial structure of the organization dominates its style of operation draws attention to person-computer interaction as a critical technology. Advances in graphics, voice input/output and intelligent terminals have improved the technical repertoire for interaction. Widespread contact with low-cost personal computers has increased computer literacy and user acceptance of the technology. However, much remains to be done to bring the human engineering of computer-based systems to the same level of professionalism as hardware and software engineering (Gaines & Shaw 1986a,b). For command and control systems involving expert systems and simulation, information presentation and ease of communication are of prime importance. A technically powerful system will be wasted unless it is comprehensible and accessible to its users, and gives them confidence in the basis of its operations.

Communication Protocols

The days of single-vendor, special-purpose systems designed against a static, well-defined specification are long gone. Integrated systems involve all aspects of the operation of a organization. Equipment from many vendors, purchased at different times, for different purposes, by different parts of the organization, has to operate together. In this lies the major practical obstacles to the creation of effective integrated systems at present. The technical problems are beginning to be resolved at the lower levels by the adoption of industry standards in communication protocols such as the open systems interconnection (OSI) standard (Day & Zimmerman 1983), the provision in data-processing and database packages for the import and export of data from "foreign" systems, and the provision of data-communications for parameter-setting and data-collection in control and instrumentation systems. These alone are necessary but not sufficient to allow the widespread creation of integrated systems. They provide a syntactic framework for system integration but no coherent semantics in which to express it.

The Applications Ring: Basic Technologies

The outer ring in Figure 1 shows the main information technologies in integrated systems and their typical applications. Moving counter-clockwise around the ring:

Remote Communications are necessary to interface the organization with the organizational world in which it operations A particular organization is part of an infrastructure each part of which is trying to optimize its own operations.

Computation Systems perform the classical number-crunching tasks under the control of algorithms that with modern software engineering have a high degree of accuracy, conformity to specification and reliability.

Database Systems have become the classic core of an organization's information system, recording all its activities in a coordinated fashion and making the information available on a controlled basis to those systems and people with a legitimate need to know.

Control Systems are now increasingly integrated with information systems so that decisions can be implemented directly in the physical world.

Instrumentation Systems are also increasingly integrated to provide timely information on the state of the physical world and the effectiveness of operations.

The Applications Ring: Decision Technologies

Three technologies in the outer ring of Figure 1 are particularly relevant to decision aids in integrated systems.

Knowledge Bases are distinguished from databases by containing inference rules rather than facts. The rules may necessarily be heuristic in nature, generating plausible advice rather than hard facts. Whereas databases record an organization's activities, knowledge bases record its skills.

Expert Systems are the means to apply a knowledge base to the solution of a particular problem. They stand in relation to the knowledge base as does conventional computation to the data base. They differ in that their role as advisors and dependence on heuristics makes *accountability* and *comprehensibility* of major importance. An expert system has to be able to justify and explain the basis of its advice.

Simulation has been separated from other forms of computation because it is based on a model of some other system and it is the relation of the simulated model to that of the system modeled that is of paramount importance. The models used in simulation to infer the behavior of a system are generally deeper than the models used in expert systems to infer the proper advice about a system, even though both may be concerned with the same system (Gaines 1986a).

HETEROGENEITY AND KNOWLEDGE-BASED SYSTEMS

What is about knowledge-based systems technology that makes it better fitted to support the management of a heterogeneous system? The answer is best seen through contrast. A conventional corporate database might contain information relevant to most of the areas shown in Figure 1. It might contain powerful programs relating to each area. However, each of the programs and associated datasets would serve a specialist function within the corporation. There would be very little direct connection between them. They would be used in a connected fashion by the corporation but the connections would be created by the human management of their application, not through information processing within the database itself. Information technology has provided us with a collection of powerful, but disconnected, specialist tools. Knowledge-based systems provide the means of integrating these tools into a coherent *knowledge support system* aiding the management of complex systems. Human knowledge is based on *heterogeneous integration*—on the pooling of diverse sources of information and diverse resources to achieve desired objectives. Knowledge-based systems can model such heterogeneous integration using information-processing facilities, and hence can support human objectives through those facilities.

The essential features of a knowledge-based system are shown in Figure 2. Each specialist area of knowledge is represented as a *knowledge island* that has within it the datasets and procedures specific to that knowledge. Data from external sources triggers off processing within an island that eventually results in processed data being passed from one island to another. This is in turn triggers further data processing in the other knowledge island which eventually propagates to other islands. Some islands have external connections that pass conclusions, recommendations for action, out to the external world. Islands are distinguished by the relative homogeneity of the data and processing within them. Connections between them represent data which is relevant to

more than one island, but probably relevant in very different ways. The structure is fully recursive in that islands generally have sub-islands within them and group naturally into areas.

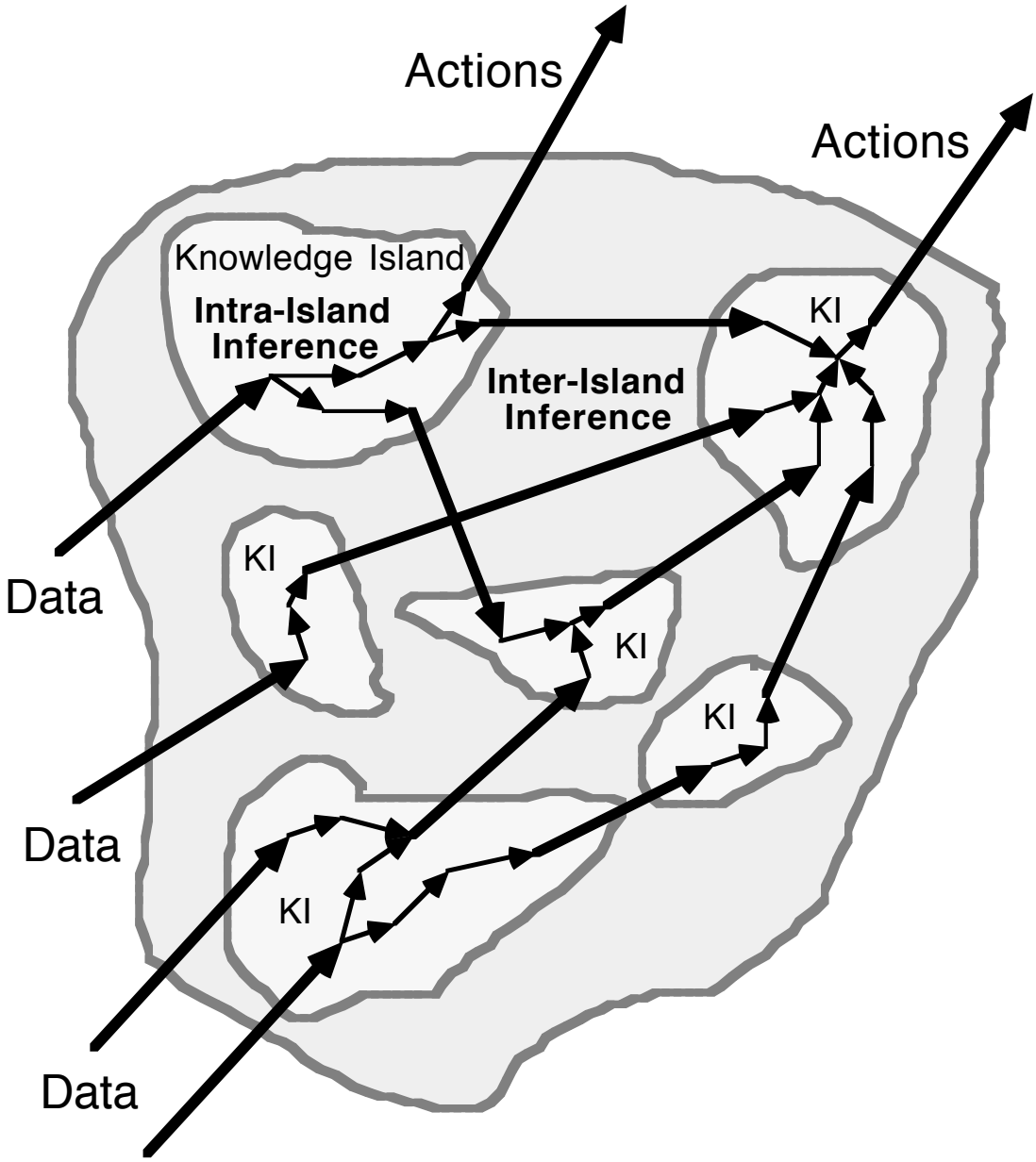

Fig.2 Modeling through heterogeneous knowledge islands

To make the abstract model of Figure 2 more concrete, Figure 3 shows an analysis of planning for flexible manufacturing in terms of the knowledge islands involved derived from interactive knowledge acquisition (Gaines 1986c). At the top left, the *overall requirements specification* is the starting point of any feasibility study of FMS. The key constructs here were found to be the obvious pair: *low-volume—high-volume* and *low-variety—high-variety,* but other significant constructs were also elicited that affect the feasibility of FMS implementation such as *FMS-already-in-use—FMS-not-in-use* and *commitment-to-existing-machines—new-plant*. The experience dimension and the difference between restructuring and green-field situations are

obviously relevant to the decision-making but could be missed in an analysis that concentrated only on quantitative parameters.

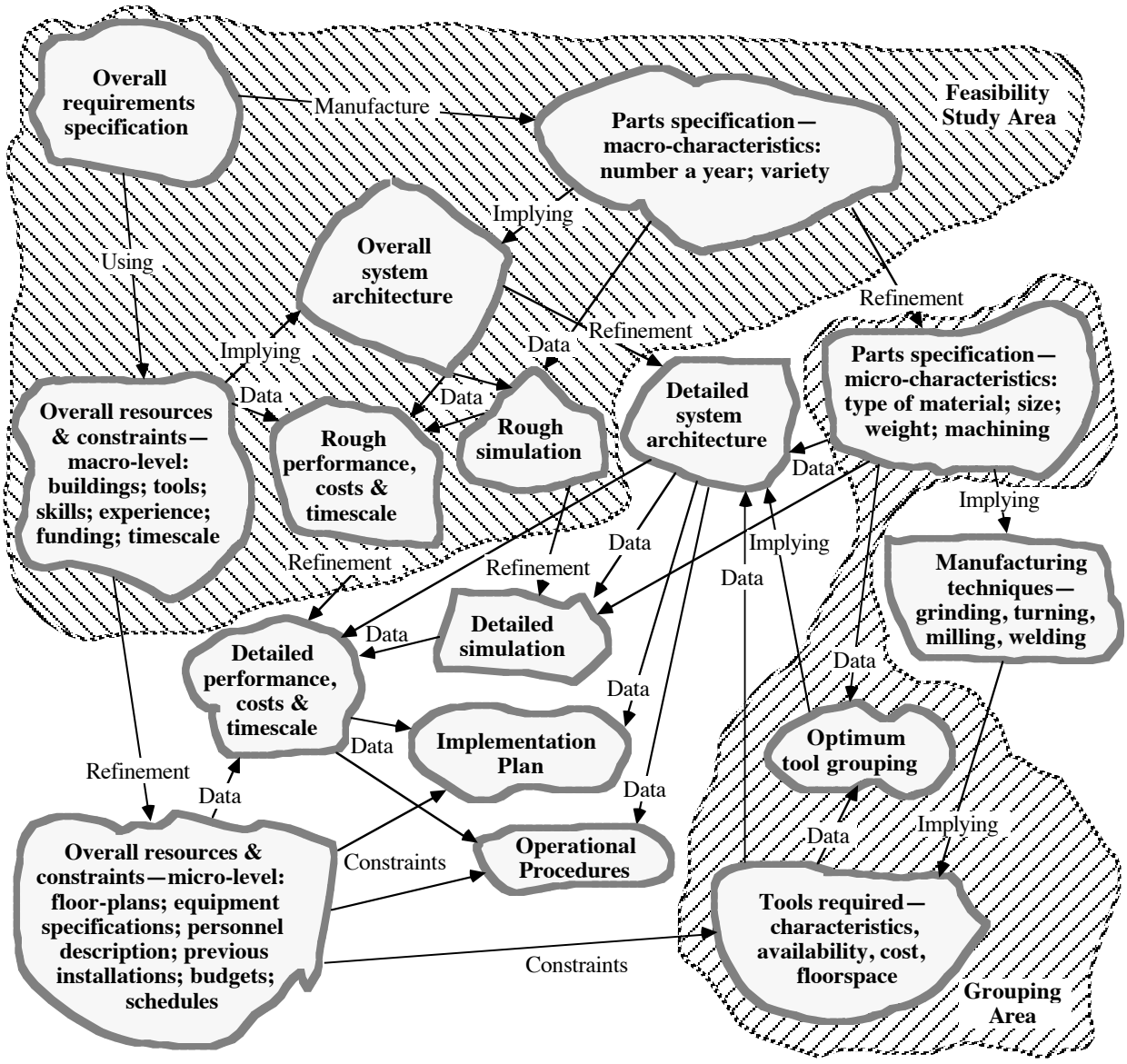

Fig.3 Knowledge islands in computer-integrated manufacturing

The experience/resources dimension refines naturally to the knowledge island concerned with *overall resources and constraints*. The volume/variety dimension refines to a quantitative description of the *parts specification* in parts per year and the number of different parts. Consideration of the resources and parts islands can give enough information to prime the *overall system architecture* island. This is in its turn can lead to *rough performance* estimates which can provide sufficient data for *rough simulation* using the fuzzy linguistic model approach (Wenstop 1981, Gaines & Shaw 1985). The performance estimates and simulation presentation form a basis for conclusions about the feasibility of an FMS for a particular requirement and the six knowledge islands involved may be grouped together as a *feasibility study area*. This sub-system is already a useful tool for the preliminary stages of FMS planning.

The knowledge islands described may be refined to a further level of detail on two main paths. The macro description of resources and constraints may be refined to a micro description that lists floor-plans, personnel, machines and so on. Much of this will be available in existing corporate administrative databases but some information will probably be missing. The macro description of parts may be refined to a micro description that completely characterizes each part and its manufacturing process. Much of this will be available in existing corporate engineering and production databases but again some will be missing. The *parts specification* leads naturally to a *manufacturing techniques* knowledge island which itself leads to that for *tools required*. The parts and tools islands form another useful sub-system, the *grouping area,* where the *optimum tool grouping* knowledge island takes the detailed specification of parts and their machine utilization requirements and uses these to derive nearly optimal groupings of machinery.

The central region of detailed architecture, performance and simulation is complicated more by the data requirements than by the complexity of computation required. It is unlikely to be cost-effective for any organization to gather the detail required on a wide range of machines and field-experience (although if widespread use of MAP develops much more data is potentially available at low cost). The knowledge islands in this area will evolve through successive refinements of those already described in a way determined by mainstream industry trends and cost-effectiveness. The tendency for leading suppliers tend to be used again because there is more knowledge of their products is one phenomenon of this region.

The importance of Figure 3 as an illustration of knowledge-based systems is the heterogeneity of the knowledge involved. The topic under consideration, FMS implementation, is relatively coherent, yet involves considering a diversity of requirements and constraints with little direct relationship between them. The whole of Figure 3 would itself be only one knowledge island in an overall manufacturing plan, which itself would be only one knowledge island in a corporate plan.

KNOWLEDGE STRUCTURES UNDERLYING HETEROGENEOUS SYSTEMS

What are the natural knowledge islands for integrated heterogeneous systems? Figure 1 captures the key problem areas for such systems and these should form the top level knowledge islands, but what about the relations between them? How do the corporate culture, person-computer interface, network protocols, information technologies, physical and informational worlds relate, and how does this affect operations? The only strongly formalized and standardized structures within Figure 1 are the network protocols, and it is useful to examine these in more detail because they involve concepts that are useful in formalizing more of topics encompassed in Figure 1.

The most important concept underlying the OSI Reference Model is that of *virtual circuits*. Designers of a particular level are concerned primarily with its relation with the equivalent level in the system with which it is communicating. They need have little concern with the levels above or below it. Hence they think in terms of messages passing directly between equivalent levels, through a "virtual circuit," rather than through the multiple levels of the actual system. The top level above the OSI defined layers is the "user level virtual circuit," representing the interactions between user programs through the communications network, and hence the interactions between users themselves, and their activities, if the programs are interactive.

This concept of virtual circuits between users, and user activities, is very powerful in enabling Figure 1 to be restructured to provide a more integrated framework for the other sections not directly included in the communications protocol. It can be redrawn in such a way that the relation of managerial and operations considerations to higher-level protocols in the network is apparent. Figure 4 shows the same conceptual blocks as those in Figure 1, but now in hierarchical form with the person-computer interface, information systems, organizational structure and external worlds shown as higher level "layers" above the network layer. In OSI terms these upper layers may be seen as subsuming and extending the presentation and application layers of the protocol, and give substance to the user level virtual circuit.

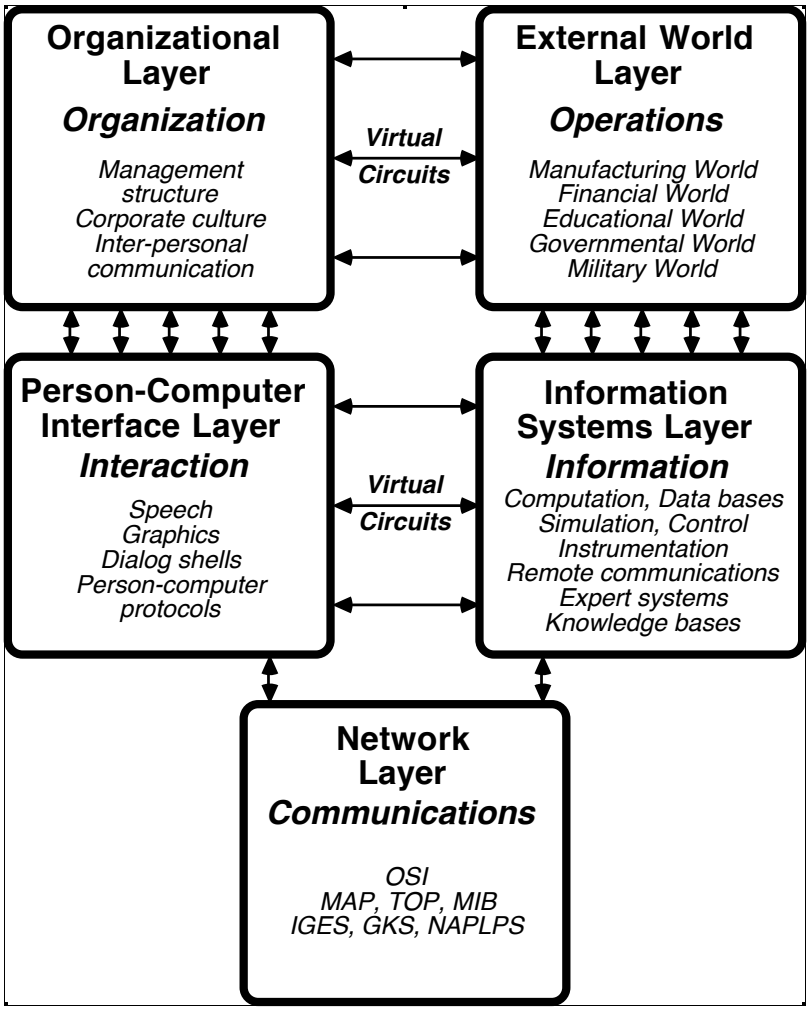

Fig.4 Knowledge structures for heterogeneous systems as a hierarchical protocol

The virtual circuits between the human-computer interface layer and the information systems layer in Figure 4 are easy to understand. Users of a computer terminal do not think of themselves as communicating through a network but rather as directly interacting with some data-processing sub-system. This layer may go beyond the OSI specification but it seems a natural extension of it to the technical aspects of the interaction between people and computing systems.

The virtual circuits at the level above, between the organizational layer and the external world layer, extend the OSI concept a step further. The sales manager does not see himself as interacting with a complex administrative and productive system, but rather as communicating directly with a customer organization by supplying it with goods. The chief executive does not see himself as concerned with that level of communication but rather as communicating directly with the board and shareholders through a flow of profits and dividends. The main problem in extending the model to the management level is that people are necessarily involved, and they have different dynamics from technical systems. However, this is also true for the level below where human-computer interaction is involved, and models have been developed at this level that integrate the human and technical components within a hierarchical layered communication model (Gaines 1988c).

KNOWLEDGE-BASED MODELING OF HETEROGENEOUS NETWORKS

Figure 4 is a major step towards modeling an integrated heterogeneous network as a structure of knowledge islands of the form in Figure 2. Each of the boxes represents a major conceptual structure in information and management science. The vertical links between boxes represent a change in levels of abstraction. The horizontal links between boxes represent conceptual information flows, or virtual circuits. The overall framework of Figure 4 partitions a heterogeneous systems into relatively well-known domains, and the links between them represent relatively well-defined flows. Note that the partitioning is functional rather than geographic. It is based on the type of activities being undertaken rather than their location in the network or organization. Note that the functionality is abstract rather than specific. It is based on the basic nature of the activities rather than their specific technical or organizational roles. These other dimensions, of the physical and logical organization of the network, and of its domain-specific functionality, are additional to those analyzed through Figure 4. Note also that Figure 4, although well-structured, encompasses highly heterogeneous systems—it conceptually packages functionality and flows but presupposes nothing about how they are implemented.

Figure 4 is the basis of our generic intelligent network support system. Each box represents an island of knowledge about the system: the communications; the information systems, hardware and software; the user interfaces; the users and their organization; and the world in which they are operating. Within each box there is knowledge about the items and issues relating to its operation. Between boxes there is knowledge about inter-relations, basically the answers to why? and how? questions. The following discussion outlines some of the issues we are addressing.

Network Layer

The network layer in practical systems is heterogeneous and fallible. There are gateways and bridges that attempt to be transparent to the user, and succeed for many transactions, but fail in others through delays, unreliability, address conflicts, lack of reciprocity, and so on. The network model needs to have a minimal, coarse-grained representation of connectivity, data-rates, protocols, and traffic. For critical activities the model may need to be more detailed, but this need should be minimized since there is no point in developing a uniform network substratum using the OSI model and then attempting to continually double-guess it.

The network layer is also significant for instrumentation. It is relatively easy to pull out information about traffic at a level sufficient to indicate the nature of the underlying activity. In

theory, it is best to instrument the higher level virtual circuits, but in practice the upper levels of the network layer are more accessible. The actual traffic flows may be used as a basis for updating the dynamic model of the network.

Information Systems Layer

The information systems layer encompasses the processes that may be run on the system and thus, incidentally, the hardware and software supporting them. This process-oriented framework is important because to the rest of the system it does not matter, in theory, on what platform or with which package a process is executed. In practice, it may matter in terms of cost, execution time, reliability, and so on, but these can be accounted as process parameters. In practice also, what this viewpoint brings to light, is that many process specifications are currently only available in behavioral terms—that this package should be executed on this platform. There is no difficulty in modeling this form of behavioral specification but its restrictions in terms of system robustness, flexibility and adaption are apparent in the model. There are often equivalences between process specifications that hold over a limited range of conditions—that this package on this platform is equivalent to that package on that platform for these specific purposes. This type of information is well-suited to a knowledge-based model.

Chalfan (1986) has shown how the use of a knowledge-based system encompassing input-output parameters and range of performance can make obsolescent software both reusable and also usable for new purposes. Her approach can be generalized to all software on a heterogeneous network. The main problem is keeping the knowledge-base up-to-date. We are cautious about depending on system administrators or developers to maintain the knowledge-base even when prompted by versioning tools. This voluntary updating needs to be automatically monitored by the system so that all changes are noted and traceable even when no explicit reasons are documented. Provision also needs to be made for documenting user experience through annotation, particularly when new software is mounted. In general, the problem is one of providing easy and natural channels for knowledge flows from users, developers and administrators to the knowledge-base. Artificial intelligence can only supplement, not replace, the massive knowledge acquisition and processing system which is the user community.

The problems of modeling the information systems layer should not be overstated, however. A knowledge-base which maintains only a fairly coarse record of the software is able to provide users with on-line advice, the means to trace problem dependencies on software changes, and the means to track their sources and possibly rectify them through backtracking to previous versions. This is a natural extension of the use of CASE tools in supporting software in the developmental phase to the use of similar tools in supporting system evolution (Gaines 1988b).

Person-Computer Interface Layer

The separation of user interfaces to processes from the processes themselves is essential to effective system modeling. To treat person-computer interaction as another process misses essential distinctions between computer-based and human-based activities. Computers are deterministic machines following strict causal dynamics—they are best modeled as state-determined systems. People are teleological entities having choice in their actions following goal-directed dynamics—they are best modeled as intention-directed systems. At the interface the anticipatory, intentional system that is the person manipulates the state-determined, causal

system that is the computer. We have to model both sides of this interaction and the conformity between them, that is, does the action achieve the goal?

In their interactions with others, people establish conventions which in many respects operate like the causal dynamics of physical systems. In particular, they are used to building models of the conventions being followed in a novel interaction. Much of human interface design consists of establishing and following conventions which are simple and natural to the user. However, there is a high degree of arbitrariness about these conventions and different systems may operate very effectively within their own domains with widely varying conventions. This is a major source of heterogeneity at the person-computer interface as these systems are integrated in a networked environment. The term 'user model' is generally used to refer to the knowledge of the family of conventions expected by a particular user.

The person-computer interface layer model has to make provision for information about interface conventions used by particular packages, the conventions expected by particular users, and the intentions of particular users, either expressed or inferred. This model can be used to manage the interface to convert between conventions if possible, to warn the user of unexpected conventions, to diagnose inconsistencies between user intentions and actions, and to advise the user about actions which may fulfil intentions.

Organizational Layer

A surprising number of phenomena in the organizational layer may be modeled using existing techniques and knowledge. For example, the concept of 'power' has a precise operational definition in terms of the potential to change the behavior of some other entity (Miles 1980). It is possible to use a model of the capabilities available to a user of the system to determine how much power that use has over any aspect of the behavior of the system. To exercise that power a user must have the knowledge to carry out the required actions, and determining whether the power is exercisable is far more difficult because the users' states of knowledge will generally be unknown. It is possible to infer some aspects of users' knowledge from their activities in carrying out their intentions if their intentions are known, and it is often possible to infer intentions from the outcomes of the activities.

The definition of power noted above may also be used to analyze the changes in the power structure of an organization brought about by the introduction of information technologies. To be the sole source of information is to have the power to force someone to communicate with you to obtain that information and hence to gain information about their activity leading to their need to know. If computer access to the information offers an alternative source then this power is lost together with its use to maintain awareness of others' activities. If this awareness is necessary to the performance of some task then it may be necessary to support it by some other means.

It is possible currently to model very complex organizational dynamics and behavior through message passing in object-orientated systems. A person may be modeled as having intentions, capabilities for action, and knowledge about these capabilities, the intentions of others and their capabilities. Some actions may be on the physical world whereas others may be on the social world, such as passing intentions or knowledge to others, and accepting or refusing intentions. Such models can reproduce complex social behavior involving group coordination in carrying out tasks. We are incorporating an organizational model of this type in the intelligent network

support system to investigate its utility in modeling the organizational layer of Figure 4. It could be thought of as a dynamic implementation of a 'Procedures Manual' for a corporation.

External World Layer

The external world layer of Figure 4 encapsulates most of the domain-specific aspects of the system operation—what is its overall role?—what are the values by which its performance is measured?—what is the terminology for the various system functions and activities? The model at this level needs to make provision for these to be 'plugged-in' and kept up-to-date. We have developed knowledge acquisition tools that elicit these domain-specific conceptual structures and terminology (Shaw & Gaines 1988). We are investigating the use of these as a front-end to a general, but coarsely structured, socio-economic model as a basis for model the external world layer.

CONCLUSIONS

Intelligent network support systems modeling a heterogeneous system to provide administrative and user support are feasible at the current state of the art in object-oriented systems. They require the development of overall organizational models that encompass corporate goals and management systems as well as the technological infrastructure supporting these. This paper has developed such an organizational model and discussed its application to the development of an intelligent network support system for integrated heterogeneous systems. Hopefully, the concepts discussed in this paper will be useful to some of the many research and development groups are now grappling with the opportunities and problems of large-scale networked systems. We welcome criticism, suggestions and collaboration in our own research. The developments reported here are part of a program of research in 'knowledge science' in which we are attempting to give a formal structure to knowledge-based systems, knowledge processing and acquisition, and to investigate the role and application of knowledge-based systems to major problem domains in science, engineering and organizational effectiveness.

ACKNOWLEDGEMENTS

Financial assistance for this work has been made available by the Natural Sciences and Engineering Research Council of Canada.

REFERENCES

Bartee, T.C., Ed. (1986). **Digital Communications**. Indianapolis: Sams.

Boeing (1985). **TOP: Technical and Office Protocols**. Seattle: Boeing Computer Services.

Chalfan, K. (1986). An expert system for design analysis. Kowalik, J.S., Ed. **Coupling Symbolic and Numerical Computing in Ex pert Systems**. pp.179-190. Amsterdam: North-Holland.

Day, J.D. & Zimmerman, H. (1983). The OSI reference model. **Proceedings IEEE**, **71**(12), 1334-1340 (December).

Deal, T.E. & Kennedy, A.A. (1982). **Corporate Cultures: The Rites and Rituals of Corporate Life**. Reading, Massachusetts: Addison-Wesley.

Gaines, B.R. (1984). Perspectives on fifth generation computing. **Oxford Surveys in Information Technology**, **1**, 1-53.

Gaines, B.R. (1986a). Expert systems and simulation in industrial applications. Luker, P.A. & Adelsberger, H.H., Eds. **Intelligent Simulation Environments**, **17**(1), 144-149 (January). La Jolla, California: Society for Computer Simulation.

Gaines, B.R. (1986b). Sixth generation computing: a conspectus of the Japanese proposals. **ACM SIGART Newsletter** (95), 39-44 (January).

Gaines, B.R. (1986c). Expert systems and simulation in the design of an FMS advisory system. Hurrion, R.D., Ed. **Simulation in Manufacturing**. pp.311-324. Bedford, UK: IFS Conferences.

Gaines, B.R. (1988a). Structure, development and applications of expert systems in integrated manufacturing. Kusiak, A., Ed. **Artificial Intelligence Implications for CIM**. pp.117-161. Bedford, UK: IFS Conferences.

Gaines, B.R. (1988b). Software engineering for knowledge-based systems. **Proceedings of CASE'88, Second International Workshop on Computer-Aided Software Engineering**. 14-1-14-7.

Gaines, B.R. (1988c). A conceptual framework for person-computer interaction in distributed systems. **IEEE Transactions on Systems, Man & Cybernetics**, to appear.

Gaines, B.R. & Shaw, M.L.G. (1985). From fuzzy sets to expert systems. **Information Science 36** (1-2), 5-16 (July).

Gaines, B.R. & Shaw, M.L.G. (1986a). From timesharing to the sixth generation: the development of human-computer interaction part I. **International Journal of Man-Machine Studies**, *1*(1), 1-27 (January).

Gaines, B.R. & Shaw, M.L.G. (1986b). Foundations of dialog engineering: the development of human-computer interaction part II. **International Journal of Man-Machine Studies**, **1**(2), 101-123 (February).

General Motors (1984). General Motors Manufacturing Automation Protocol. Warren, Michigan: General Motors Corporation.

Gordon, P. (1984). What is integrated software?. **PC World**, **2**(11), 72-77 (October).

Hall, E.T. (1959). **The Silent Language**. New York: Doubleday.

Jarvis, M.T. (1985). **Molecular Electronics: Beyond the Silicon Chip**. Fort Lee, New Jersey: Technical Insights.

Leopold, G. (1986). Bus standard for hospitals is next. **Electronics**, **59**(11), 17-18 (March).

Miles, R.H. (1980). **Macro Organizational Behavior**. Glenview, Illinois: Scott Foresman.

Miller, R.K. (1986). **Optical Computers: The Next Frontier in Computing**. Madison, Georgia: SEAI.

Moto-oka, T., Ed. (1982). **Fifth Generation Computer Systems**. Amsterdam: North-Holland.

Nolan, R.L. (1983). Computer data bases: the future is now. Salerno, L.M., Ed. **Catching Up With the Computer Revolution**, 225-248. New York: Wiley.

Shaw, M.L.G. & Gaines, B.R. (1988). A methodology for recognizing consensus, correspondence, conflict and contrast in a knowledge acquisition system. **Proceedings of the Third AAAI Knowledge Acquisition for Knowledge-Based Systems Workshop**. Banff (November).

Wenstop, F. (1981). Deductive verbal models of organizations. Mamdani, E.H. & Gaines, B.R., Eds. **Fuzzy Reasoning and its Applications**. pp. 149-167. London: Academic Press.