Market Analysis: Application Development Tools

INPUT®

MARKET ANALYSIS:
APPLICATION DEVELOPMENT TOOLS

MARKET ANALYSIS:
APPLICATION DEVELOPMENT TOOLS

ABSTRACT

This is the final report in a series on development tools. The first two reports of this series were:

- Market Analysis: Data Base Management Systems.

- Market Analysis: Fourth Generation Languages.

This report focuses on the competitive environment for tools, aids, and design methodologies. More specifically, this report analyzes the market requirements.

This report contains 78 pages, including 13 exhibits.

MARKET ANALYSIS: APPLICATION DEVELOPMENT TOOLS

CONTENTS

MARKET ANALYSIS: APPLICATION DEVELOPMENT TOOLS

EXHIBITS

I INTRODUCTION

- This is the third report in a series which includes:
  - Market Analysis: Data Base Management Systems.
  - Market Analysis: Fourth Generation Languages.
- These reports form an integrated set and should be viewed as such.
  - The first report, Market Analysis: Data Base Management Systems, emphasized the importance of DBMS in IBM's software strategy and pointed out important windows of opportunity based on projected hardware/software technological developments.
  - The second report, Market Analysis: Fourth Generation Languages, emphasized the need for structure in software market analysis and presented a set of systems categories which are useful in establishing a frame of reference in discussing both markets and competition.
- This report will concentrate on the competitive environment in the overall market for tools, aids, and methodologies to improve productivity in applications development. In doing so, INPUT will emphasize market requirements, not current products. It is our opinion that we have currently gone past the point where the "solutions" can be used to define the problem (and the market).

- Since this report series was specified, a rather nasty controversy has developed around the definition of a relational data base system. It all started when E. F. Codd, the primary inventor of the relational model, published an expanded version of definitions contained in his 1981 ACM Turing Award Lecture.

  - It is doubtful that anyone disputed Codd's right to define his creation after the Turing Award Lecture (or any of his other technical publications). In fact, INPUT has specifically stated that: "Since Codd is so closely identified with the relational model, it seems only reasonable to accept his definitions of the relational model and what constitutes a relational data base system." (Relational Data Base Developments, INPUT, August 1983.)

  - Unfortunately, the term "relational" has been applied to many successful (and unsuccessful) products rather indiscriminately. In recent Computer World articles and letters, some DBMS vendors saw fit to take issue with Codd's definitions. (This is unfortunate since the real value of the relational model rests with its solid theoretical foundation as much as it does with its external characteristics.)

  - In Codd's response to some of the criticism of his articles, he emphasized precise definitions for software product evaluation and market analysis for not only DBMSs, but for languages as well. Specifically, he states:

    . "There is no fourth generation language definition worth its salt, let alone any theoretical foundation. James Martin's purported definition fails to mention what capabilities a fourth generation language should have. . ."

. "Thus, any vendor can claim to provide a product that supports a fourth generation language, and there is no basis for checking or challenging such a claim."

- This is precisely the point INPUT made in its report on fourth generation languages--the term has a fuzzy meaning.

- This lack of structure at the most fundamental level is precisely the reason INPUT saw fit to combine these three reports. DBMSs, FGLs, and other application development tools (ADTs) do not have definitions and are all competing for the same market which can be roughly defined as "the market for ways to improve productivity in the systems (applications) development process."

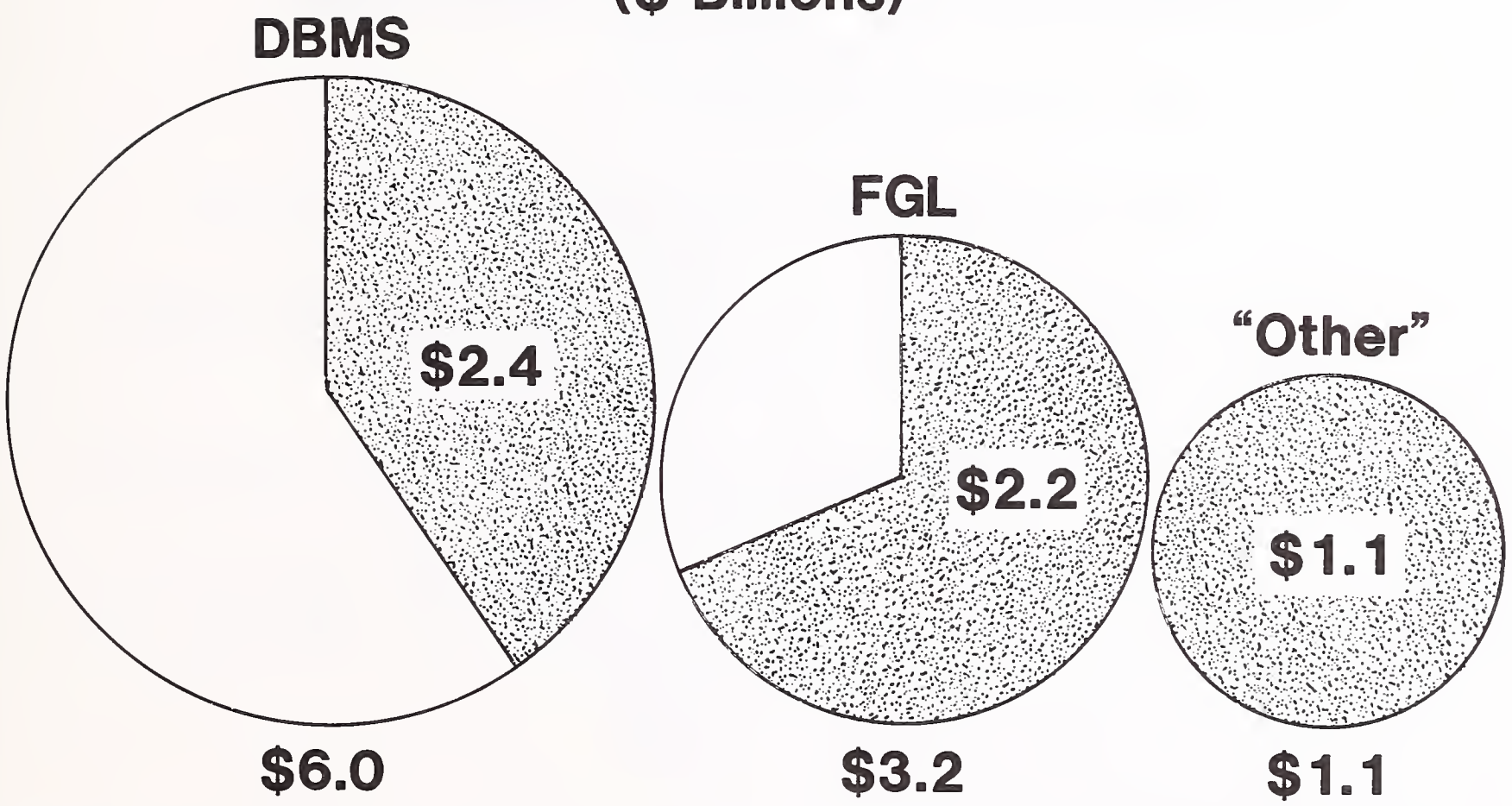
II EXECUTIVE SUMMARY

- This executive summary is designed in a presentation format in order to:
  - Help the busy reader quickly review key research findings.
  - Provide an executive presentation and script that facilitates group communications.
- The key points of the entire report are summarized in Exhibit II-1 through II-6. On the left-hand page facing each exhibit is a script explaining the exhibit's contents.

A. ADT FORECASTS - 1990

- The total market for application development tools for 1990 is $10.3 billion.
- The effective market is that remaining after IBM extracts its "share."
- The DBMS and FGL market were analyzed in detail in previous reports of this series.
- The market for "other" tools and aids is projected to be $1.1 billion.
  - INPUT believes the most promising of these tools and aids will address the quality assurance issues of the distributed systems development (DSD) environment.
  - The entire market will be effective because IBM will approach quality assurance as an extension of operating systems.

ADT FORECASTS - 1990
($ Billions)



B. THE COMPETITIVE ENVIRONMENT

- There is a timeless quote that "IBM is always the competition whether they have a product or not." It is true.

- However, the second tier of competitors also present a formidable barrier for small vendors in the effective market.

  - IBM has 43% of the total market.

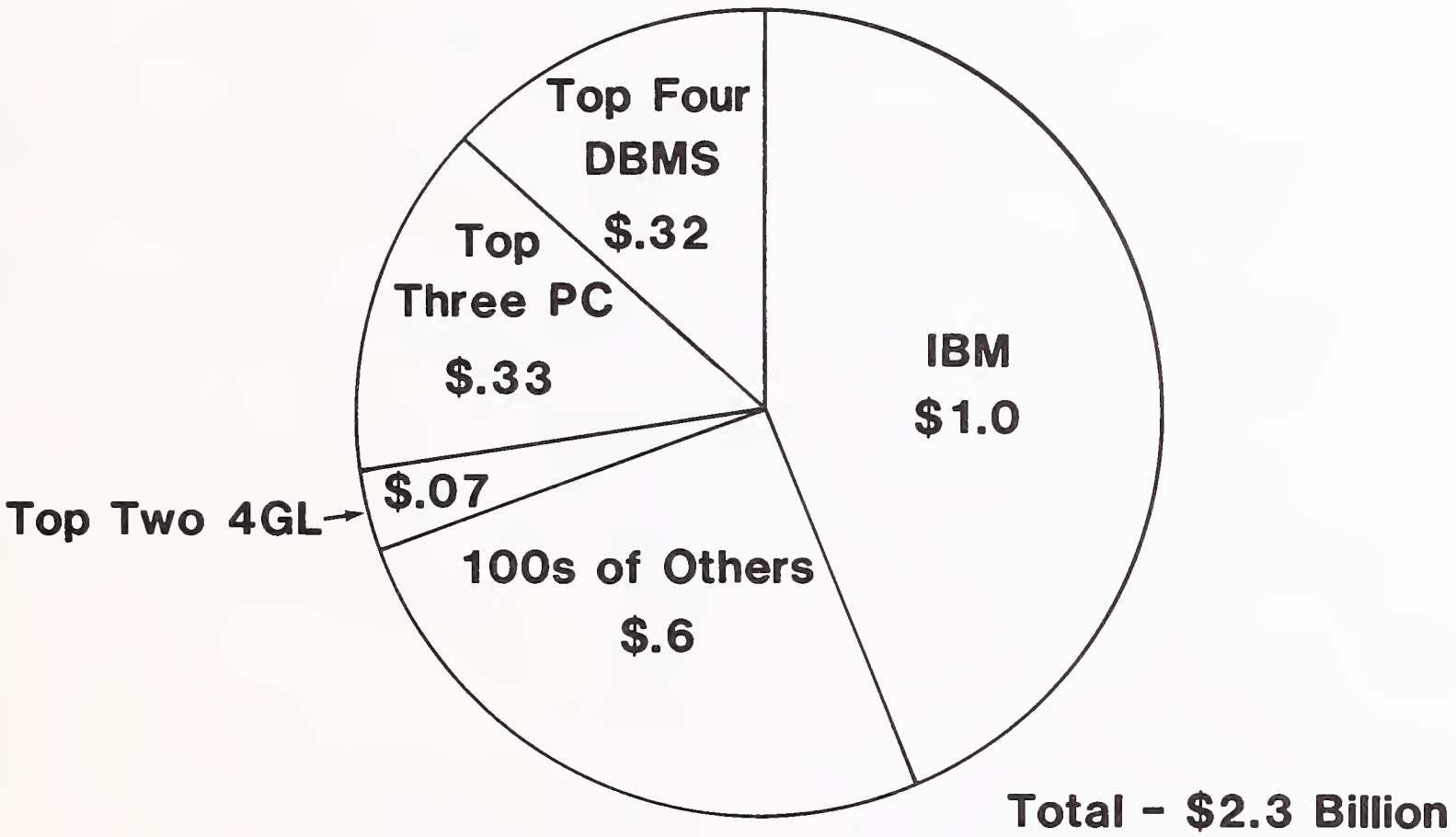
  - Nine major vendors have 54% of the remaining effective market.

- It will be extremely difficult for a new, improved DBMS or FGL vendors to achieve any significant market penetration.

- This is especially true because the market is already cluttered with ill-defined products.

INPUT®

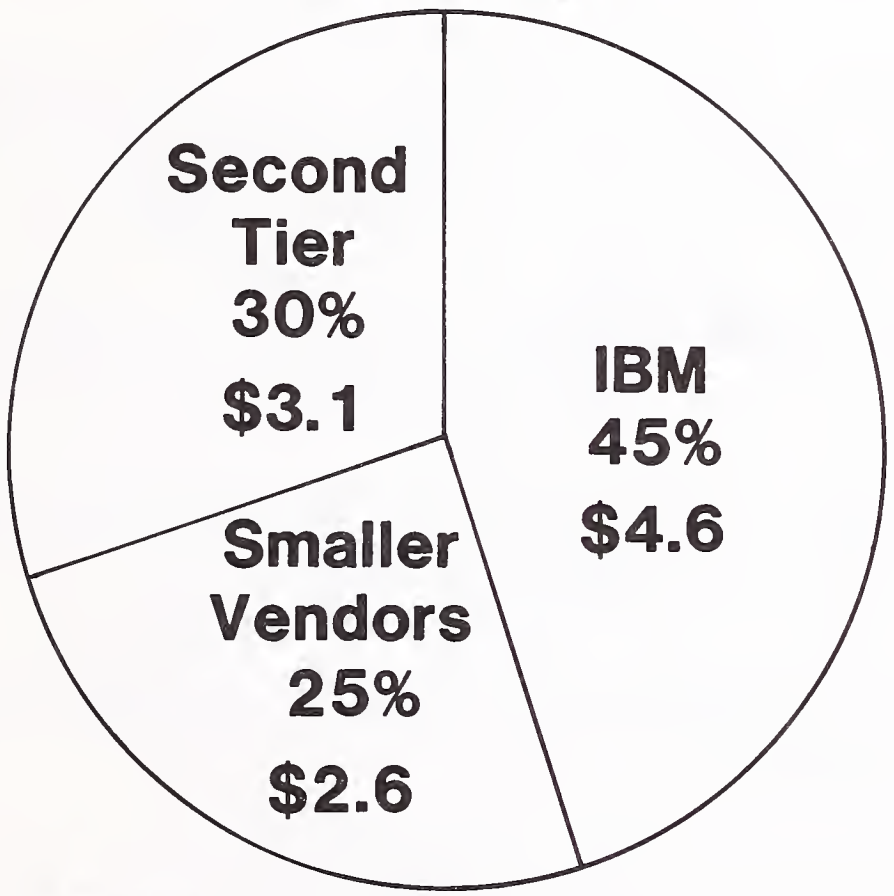
THE COMPETITIVE ENVIRONMENT - 1984
Application Development Tools Market
($ Billions)



C. APPLICATION DEVELOPMENT TOOLS—MARKET SHARE, 1990

- Assuming relatively stable market shares among IBM, the second tier (of 9 or 10 competitors), and other software companies in 1990:
  - IBM will have $4.6 billion in revenue from the ADT market.
  - The second tier will have $3.1 billion.
  - Smaller vendors will share a $2.6 billion market.
- INPUT believes IBM and second-tier vendors will virtually control the DBMS market.
- Therefore, the market available to the smaller vendors will consist primarily of FGLs and "other."

APPLICATION DEVELOPMENT TOOLS MARKET SHARE - 1990 ($ Billions)



Total Market - $10.3 Billion

D. THE MARKET FOR INNOVATION

- Most significant innovation springs from small organizations. Therefore, it is possible to view the market share for smaller companies or divisions within companies as being the market for innovation.

- INPUT's FGL definition includes expert systems since they can be considered another level of language differentiation.

- It is projected that most new products to satisfy the market requirements for quality assurance will come from small companies and this is essentially the "other" category of ADTs.

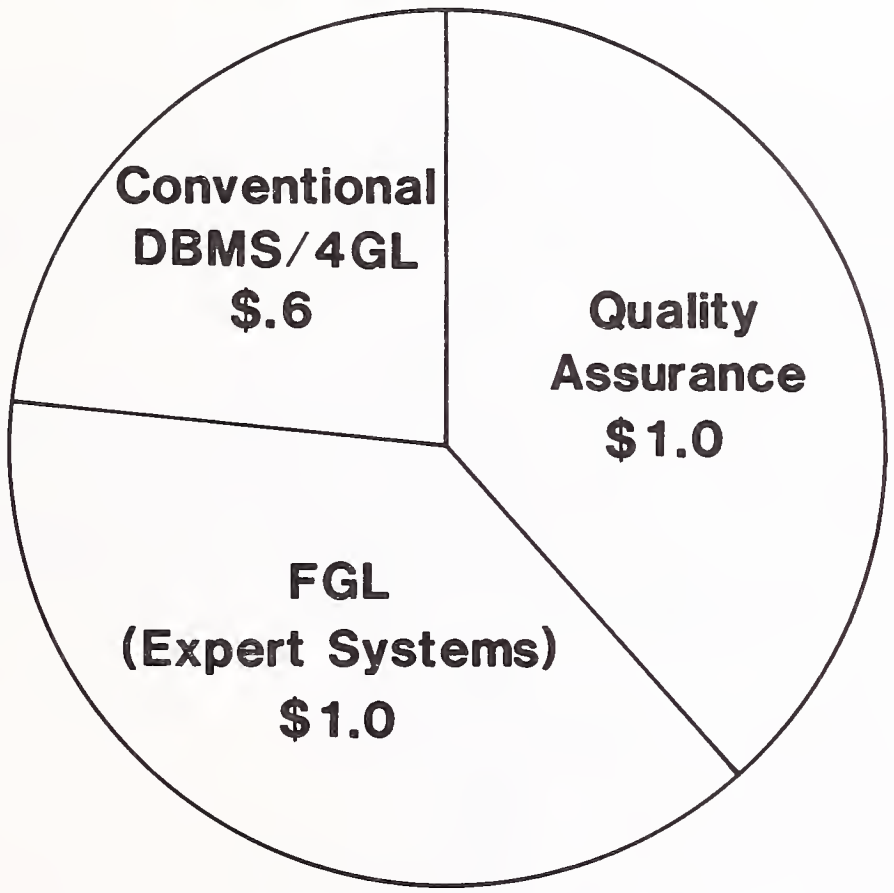
  - IBM has little motivation to improve hardware/software performance except to meet competitive threats.

  - Second-tier competitors will be busy competing against IBM and each other in the DBMS and FGL markets.

- The market for innovation in conventional DBMS/FGL products will remain relatively stable.

INPUT®

THE MARKET FOR INNOVATION, 1986-1990
($ Billions)



$2.6 Billion (1990)

E. CHARACTERISTICS OF INNOVATIVE ADT PRODUCTS

- The requirements for quality assurance in the distributed systems development environment have been clearly established in earlier INPUT reports.

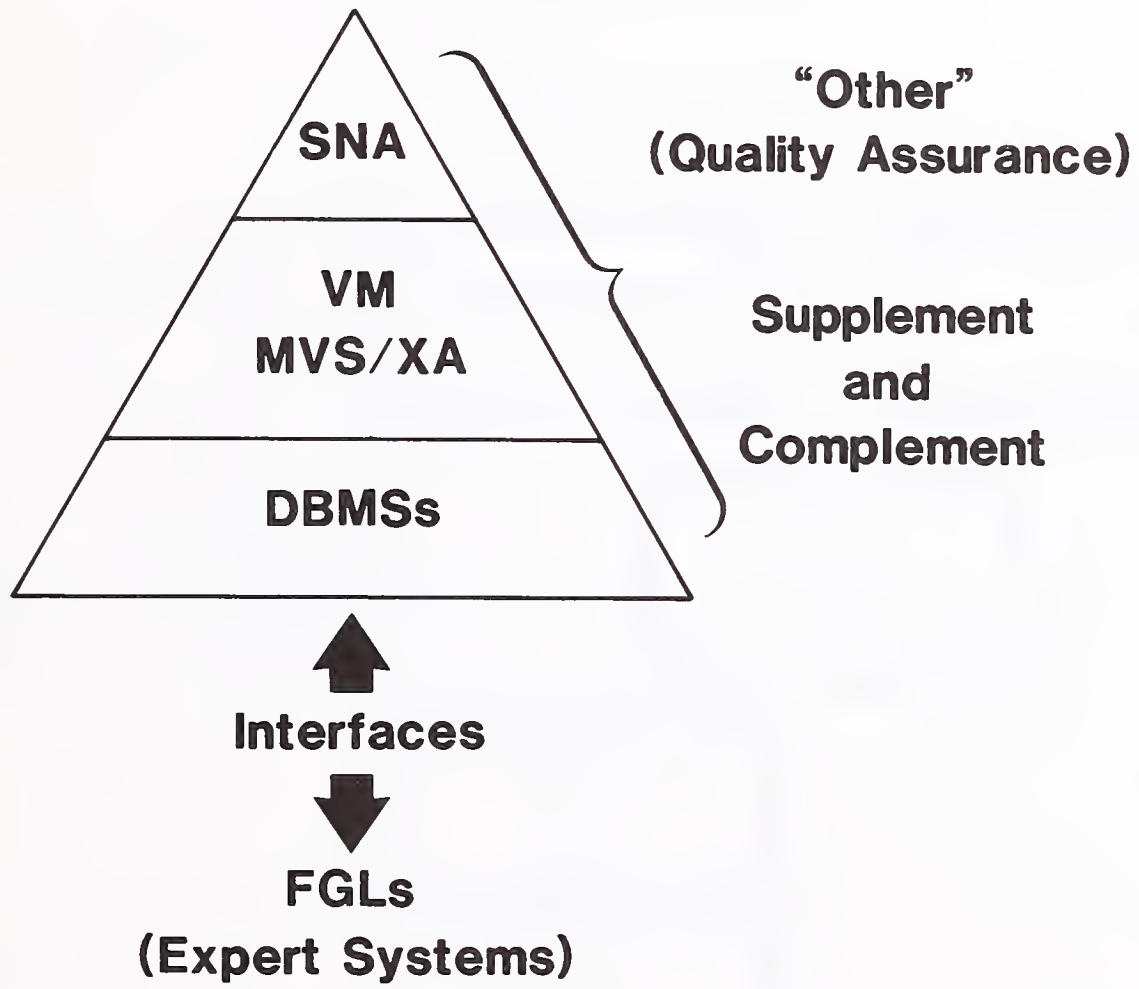
- It is concluded that tools to support quality assurance will supplement and complement the top of the software hierarchy (SNA, operating systems, and DBMS).

- It has also been concluded that unless issues of quality are addressed, the projected market for conventional ADTs could be severely impacted because applications development tools themselves contribute to quality problems if they are used improperly.

- It is also important to note that expert systems (FGLs) must interface with established DBMSs in order to obtain statistical data, but not with current 4GLs which are inadequate for either developing or establishing communications at the human-machine dyad.

INPUT®

THE CHARACTERISTICS OF INNOVATIVE ADT PRODUCTS

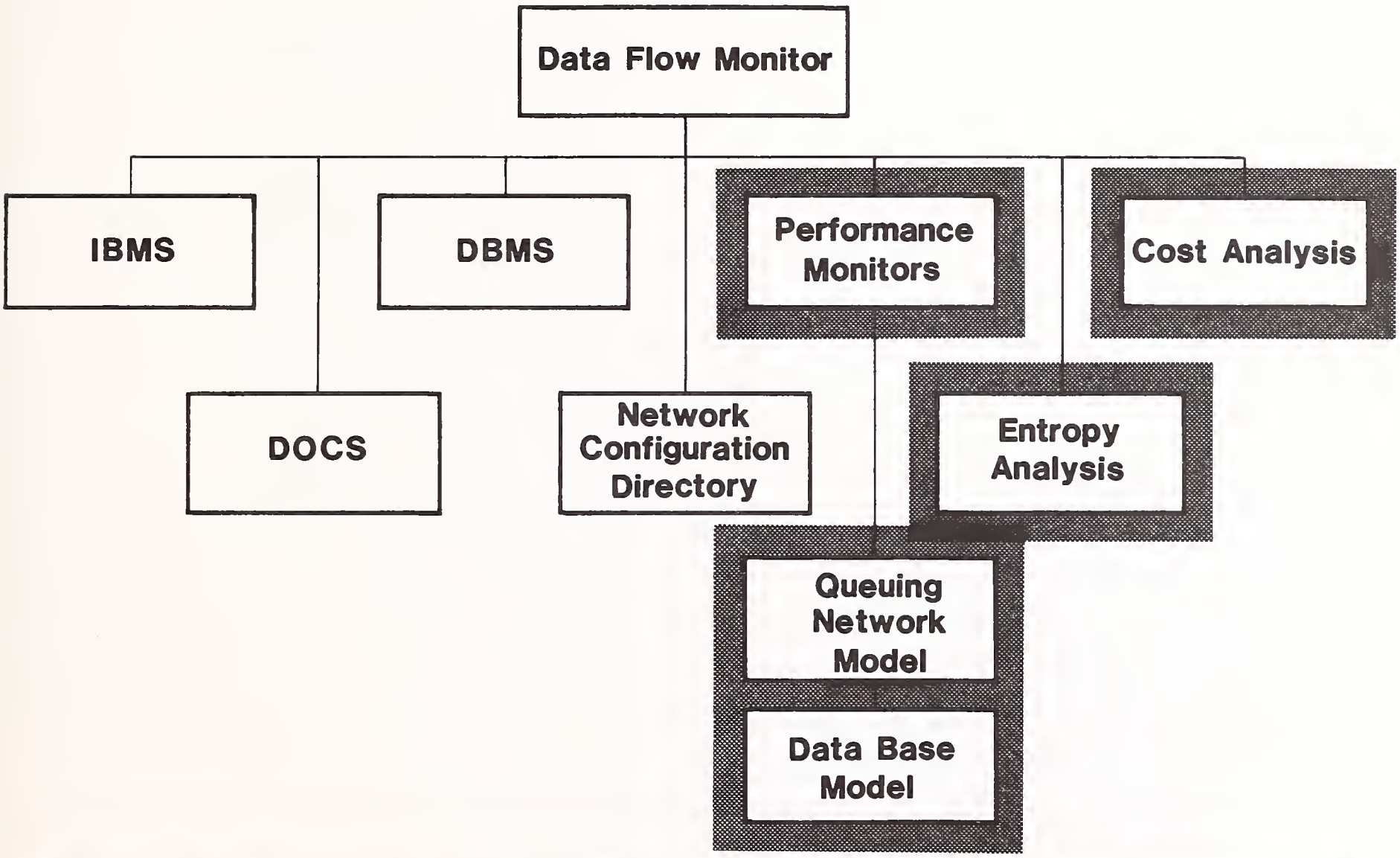


F. HOW THE ADT "OTHER" CATEGORY FITS

- INPUT defined some general productivity tools for the DSD environment in Market Impact of New Software Productivity Techniques, 1984.
  - An information base management system.
  - A document control system.
  - A data flow monitor.
  - Operations research and artificial intelligence tools, and tools providing for added security, protection, and privacy.
- The "other" category of ADTs fits under data flow monitor, which obviously must interface with the top of the software pyramid.
- Specific subsystems which comprise the "other" category are those shaded in Exhibit II-6.

INPUT®

A DATA FLOW MONITOR (DFM)



III MARKET ANALYSIS - APPLICATION DEVELOPMENT TOOLS

A. MARKET REQUIREMENTS

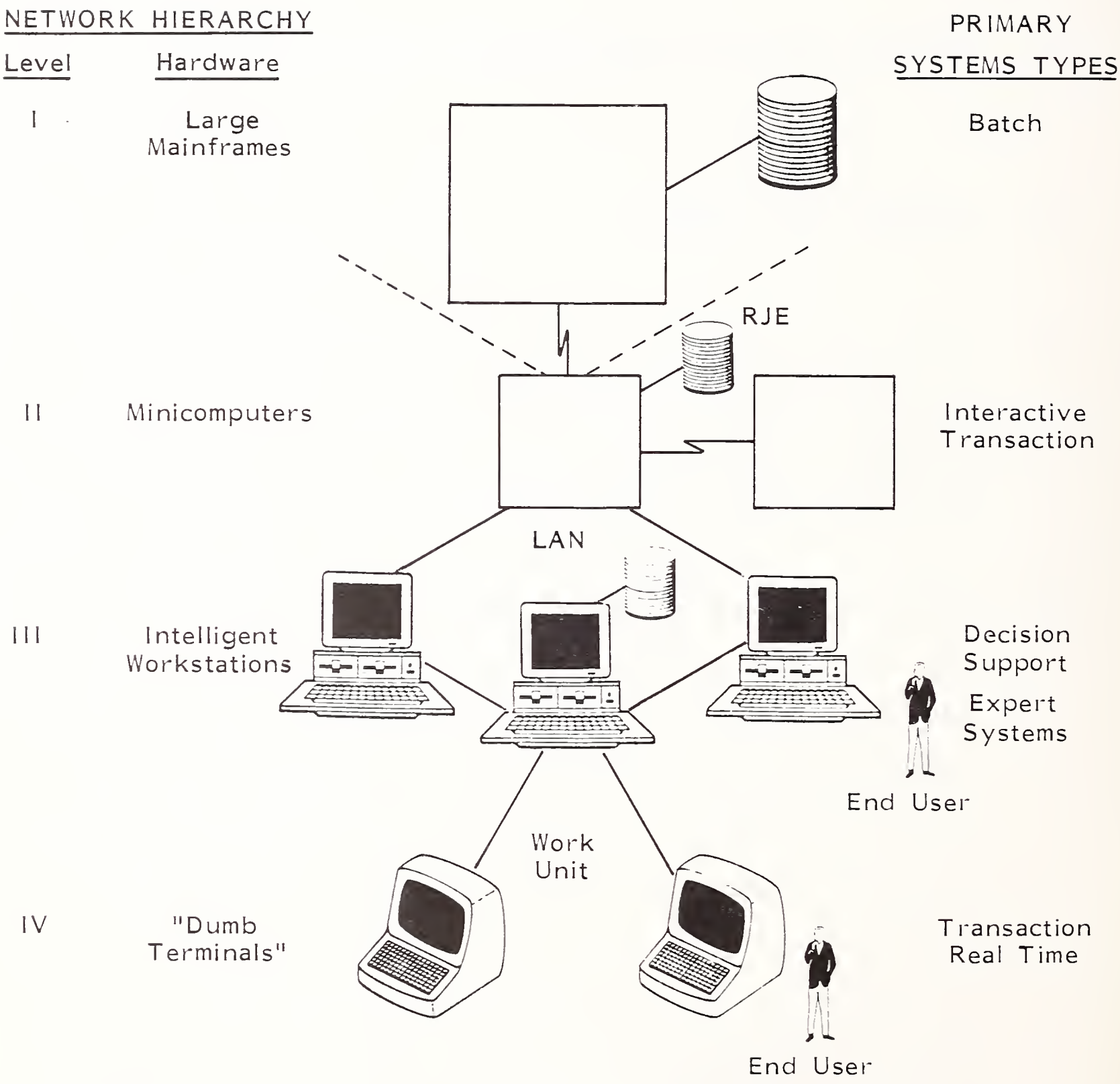
- INPUT starts with the assumption that the market for application development tools is determined by the applications which will be developed with those tools rather than the other way around (where the tools determmine the application). Sometimes this does not seem to be the case; our industry is becoming noted for changing problems to fit solutions.

- Having made that assumption, it is possible to gain considerable insight into market requirements by relatively simple analysis of what users are saying, which is essentially this: "I want to be able to sit at an intelligent workstation (or have my employees sit at their workstations) and have ready access to all of the data and processing power of the network without regard for where the work is actually done." In other words, the applications and their necessary data will be distributed over a computer/communications network and will flow over that network freely with only minimal navigational direction from the end user.

- Using INPUT's network hierarchy systems category (see Appendix A of Market Analysis: Fourth-Generation Languages for a list of systems categories), it is possible to visualize the market requirements quite clearly (see Exhibit III-1).

EXHIBIT III-1

APPLICATION STRUCTURE



- INPUT has assumed that this is a "natural" or "proper" network hierarchy associated with computer/communications networks. This proper hierarchy is dictated by its cost-effectiveness and determines the appropriate functions (applications) at the various levels. This hierarchy was first presented by INPUT 10 years ago, and it has remianed relatively unchanged except for terminology.

- The functions originally assigned at Level I were as follows:

  . Heavy computation.

  . Transaction processing against large data bases.

  . RJE replacement of standalone batch systems.

- The original functions at Level II were as follows:

  . Network control.

  . Scientific timesharing.

  . Program development and maintenance.

  . Simple transaction processing.

- The original functions at Level III were:

  . Collection, editing, and display of data and information.

  . Control of Level IV terminals.

- The original functions at Level IV were:

  . Data entry and display (including printing).

  . Sensing and control devices.

- Level V mobile terminals were not shown in the original hierarchy, but they were mentioned in the body of the report.

- While it is obvious that the network depicted falls conveniently into the established hierarchy and fits with today's terminology, there might be some dispute when the systems type systems category is assigned at various levels. It is INPUT's position that any discrepancies are primarily the result of IBM's strategy (for example, running UNIX on mainframes is in keeping with IBM's reluctance to distribute interactive processing to its proper level). However, from the point of view of end users at Levels III, IV, and V, usage (applications) patterns will soon develop which support this distribution.

  - It is all well and good for users to state they do not want to be concerned about where data is or where processing is done, but there is one thing they are concerned about and that is cost; and as users become directly involved in the operations (and development) of their systems, they will be in a much better position to understand relative costs. It is INPUT's opinion that this increased user awareness will encourage--if not force--the cost-effective distribution of functions and systems types into a "proper" hierarchal network through usage patterns.

  - Perhaps the simplest illustration of the natural tendency toward cost-effective use is found with the use of today's public information networks, such as The Source and CompuServe. Early usage patterns may find users composing correspondence and browsing through information on-line, but it does not take many monthly bills to convince most of them that file transfer makes a lot more sense.

- It will not take long for users to become aware of the cost of constantly inquiring into corporate data bases and there will be a natural tendency to upload and download files and data base subsets based on both response time and cost. A few JOINs and SELECTs against large DB2 data bases on the host computer will be more than enough to convince most users that there must be a better way. (This statement is made with all due respect for Dr. Codd, whose relational model will become the data model upon which these distributed systems will be built.)

- Then, or course, with IBM's multiple data base strategy, the extract runs against IMS data bases, VSAM files, and sequential files to create relational tables are going to be far from interactive. A user requesting data from an archival tape file may not have to know where the data resides (or on what media), but the application should be designed to tell him/her when he/she can expect to receive it (hours, days, or whatever), and how much it will cost.

- The current, popular emphasis among IS management is upon "connectivity," and there is an awareness that future systems will be communications-based. Unfortunately, both users and vendors associate anything connected with a computer/communications as being interactive, which is not necessarily the case. The telephone system has normally been interactive because it was necessary for both parties to be connected at the same time, but it must be recognized that the bulk of communications still takes place on paper and the U.S. Postal Service is also a communications network. It may be possible to reduce the massive paper output of computer systems (most of which has been distributed through internal or external mail), but the protocols between Levels I and II and II and III (or micro to mainframe) are going to look more like 2780 batch processing than they are interactive timesharing.

- Both IS management and many vendors prefer to ignore batch processing, but it is not going to go away. If the term has become an anathema, perhaps we can refer to communications between various network nodes as being in "blast" or "flash" mode (we are good at changing terminology and not concepts), but the requirement for tools and aids to develop flexible batch applications remains.

- Systems designers and application developers need tools which will:

  - Assist them in accommodating a variety of user languages at Levels III, IV, and V and a variety of data base systems at all levels in the hierarchy.

  - Provide facilities for identifying sources of data/information/knowledge and for incorporating data/information/knowledge into the applications systems being developed.

  - Guide them in how data/information/knowledge should be distributed over the network hierarchy in order to achieve balanced productivity improvement as defined by INPUT's performance systems category which includes the following levels:

    . Hardware/software.

    . Human/machine dyad.

    . Work unit network.

    . Institutional.

  - Help them in determining which tools to use to achieve balanced performance improvement (productivity) in terms of the systems requirements systems category which includes the following sub-categories:

  - . High/low transaction rates.
  - . High/low processing requirements.
  - . Large/small data base size.
  - . High/low functionality.
  - . Many/few decision rules.
  - . High/low responsiveness.

- Facilitate the development of quality assurance programs for the applications systems being developed, including the subcategories contained under INPUT's quality systems category:
  - . Objectives.
  - . Data/information/knowledge.
  - . Auditability.
  - . Measurement.
  - . Feedback loops.
  - . Validity/reliability/predictability (of achieving objectives).
  - . Security/privacy.
- Provide flexibility and facilitate change in all of the above.

- Vague requirements definitions such as "connectivity," LANs, and micro-mainframes links and universal solutions such as relational DBMSs, 4GLs, "information engineering," and data-driven prototyping all become part of the problem when the systems analyst and development manager are confronted with developing quality systems (much less integrating the hodgepodge of prototypes, expert systems, and communications systems being developed). The chaos which curently exists in the systems development process (and among the "experts" in the industry) is the direct result of improperly applied tools (both hardware and software). What is needed are tools, techniques, and approaches which facilitate, direct, and/or force intelligent application of many of the tools already available.

- Therefore, if the requirements above do not correspond with your particular "solution," appear complex or impossible, and are not being specifically articulated by systems personnel, it is not surprising. Both the problem and some more detailed requirements were presented in Market Impact of New Software Productivity Techniques, INPUT, 1984, and events of the last year have only confirmed the findings of that report. The specific recommendations of that report will be summarized later.

B. CURRENT PRODUCTS

- There are a great variety of productivity tools available to address specific aspects of the requirements outlined above. INPUT classified these tools into some general categories by systems development phase in a 1983 Vendor Watch Report on Software Productivity Tools: Update and Outlook (see Exhibit III-2). The report then attempted some additional clarification by regrouping the tools into "pre-implementation," "implementation," and "revolutionary" categories.

EXHIBIT III-2

CLASSIFICATION OF SYSTEM PRODUCTIVITY TOOLS (SPTs)

STP	EXAMPLES	SYSTEMS DEVELOPMENT PHASE ADDRESSED		
		Requirement Definitions	Design	Implementation
Artificial Intelligence	LISP, SMALLTALK	X	X	X
Data Dictionary	Datadictionary (ADR) DB/DC (IBM) UCC-10 (UCC)	X	X	X
Data-Driven Prototyping	PDM-80	X	X	X
Design Methodologies	Structural Design (De Merco) Structural Analysis	X	X	
Information Planning	Business Systems Planning Information Modeling	X		
Modeling/Non-procedural Languages	Focus, Express, Easytrieve	X	X	X
Programming Aids	Program Utilities (CAPEX) Structural Programming			X
Visual Programming	MAPPER, VisiCalc		X	X

- Pre-implementation (requirements definition/design) tools were listed as including the following:

  . Business or information systems planning (e.g., IBM's BSP).

  . Data gathering/analysis techniques (e.g., information modeling).

  . Structured analysis/design (e.g., DeMarco, Yourdon, SofTech, HIPO, and PRIDE).

  . DBMSs.

  . Software aided (e.g., DDI - J. Martin; DDSD - K. Orr).

  . Application prototyping.

  . Data dictionaries.

- Implementation tools were listed as follows:

  . Structured programming (e.g., SPF).

  . Program code generators.

  . Higher level retrieval languages (e.g., DYL 280, Easytrieve).

  . Fourth generation languages (e.g., Focus, INTELLECT).

  . DBMSs.

  . Programming utilities (e.g., Capex, Optimizer).

  . Systems management aids (e.g., JARS).

  . Telecommunications monitors (e.g., CICS).

- Revolutionary techniques were described as spanning both the pre-implementation and implementation phases and were listed as follows:

  . Visual programming (MAPPER, VisiCalc, etc.).

  . Data-driven prototyping (PDM-80 from DACOM).

  . Artificial intelligence (exploratory programming: LISP, SMALLTALK, etc.).

- INPUT then stated: "As depicted in the chart, the earliest SPTs (system productivity tools) were intended to support programmers. These tools increased programming productivity, but they did not increase systems development productivity. IS management concluded that programming the wrong system faster would not solve the problem. New tools were developed to better define requirements. Still more tools are being developed to cover all phases of systems development life cycles, starting with systems planning and needs analysis and continuing through performance monitoring."

- The above listings made no pretense of being comprehensive, but did point out the wide variety of products competing in the market for software productivity tools. More importantly, they provided the insight to identify the orientation of the major competitive thrusts in the marketplace.

- Fundamentally, there are thrusts coming from three directions (the ones which cover all phases of the systems development life cycle) and they are all directed at penetrating the same market.

  - There are competitors coming basically from a DBMS orientation.

  - There are those whose primary emphasis has been language oriented (as manifested by 4GLs).

- And, there are the new brands of "PC jockeys" who have started from standalone personal computers ("visual programming" really originated here) and now find themselves hooked into the network hierarchy at Levels III, IV, and V.

- Regardless of the original orientation, all tools are confronted with integration problems as their use is extended into markets where others have established early penetration. Each of the three major competitors have something to learn from the others. For example, developers of DBMSs and 4GLs may think they understand something about "ease of use" until they encounter the new user or those weaned on PC software, and integrated PC software vendors may think they understand DBMSs until they run into the data base integrity and security problems associated with shared use and distributed data bases. Then, overlaying the whole problem is the fact that the targets keep changing. Consider the following quote which was recently published.

  - "The industry has been very slow to recognize how quickly people become power users." (Attributed to Richard Rabins, president of Alphs Software.)

- Of course, there are those who recognize the complexity of the problem and the deficiencies of putting too much dependency on past solutions. New, comprehensive solutions (such as data-driven prototyping) frequently have merit, but have great difficulty becoming accepted for the following reasons:

  - Users have been exposed to so many solutions and promises that they will not take the time to consider (much less understand) a new approach or tool.

  - The developers of the new tool cannot find qualified personnel to market, sell, or install their product.

- The cost of launching new products (even software products) is increasing substantially as there are more and more announcements of new "solutions" and established vendors enhance their products and increase their customer base.

- Venture capitalists have been burned so often recently that they are extremely reluctant to provide the funds to launch a new software product.

- All of the above lead INPUT to believe that the primary competition will center around established vendors from the three primary product areas (DBMSs, 4GLs, and PC-oriented software) competing against each other. In addition, only the well established (or those with a deep-pocketed patron) have a chance of surviving in today's complex, disillusioned marketplace.

C. COMPETITIVE ENVIRONMENT

- Before discussing the competitive environment for application development tools, it is important to emphasize INPUT's analysis of the overall software marketplace. This analysis was summarized quite succinctly in Software News, September 1985 ("Software News' Top 50 Independent Software Vendors" by Peter Cunningham and Bonnie Digrius of INPUT).

  - "Increasing applications and systems software integration into a single product offering. Thus, the most successful products of the last half of the decade will have almost as much value added from systems software components as they will from applications software parts."

    - "For example, the full value of a general ledger system will be as much due to integrated systems software components such as DBMS, micro-mainframe links, and 4GLs (fourth generation language) as to the basic accounting functions."

- "Emergence of a true distributed data processing environment (DDP). This new, more complex universe, with its multi-layered processing and data base locations, is causing product obsolescence. (Bad news for vendors with older product offerings and/or limited resources, but multiple new opportunities for major new product offerings.)"

- The INPUT made the following observations relating to competitive structure:

  . Fortune 1000-type firms are becoming more aggressive marketers of software products. (For example, McGraw Hill and Continental Telecom.)

  . IBM is becoming more aggressive in pricing, developing, and making joint marketing agreements for software products for all sizes of computers in all major markets.

  . Non-software information services vendors (e.g., Martin Marietta Data Systems and Dun & Bradstreet) are expanding beyond their traditional offerings to include software products as part of their total offering.

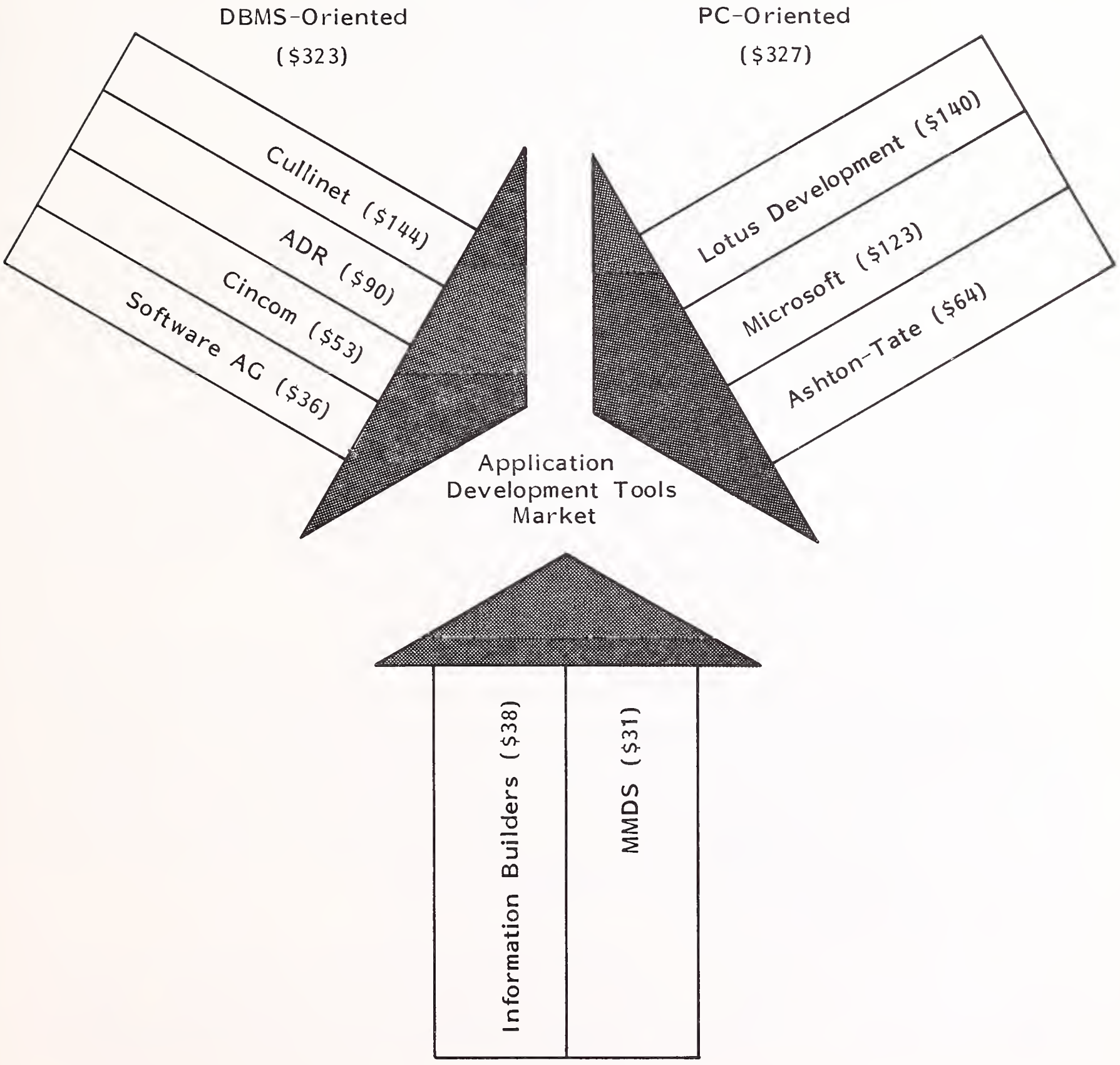
  - INPUT concluded that there were many opportunities for "innovative, forward looking vendors. . .willing to make major investments in quality management as well as marketing and technical resources."

- Remembering the ever-ominous presence of IBM, the following will be an overview of the major independent competitors from the perspective established in this series of reports (see Exhibit III-3).

  - There are four major vendors coming from a DBMS orientation with combined U.S. revenues of $323 million in 1984. All four are currently giving attention to both 4GLs and PC links of some kind.

EXHIBIT III-3

WHERE ADT COMPETITORS ARE COMING FROM



- Cullinet is the largest independent software vendor in the world with $144 million in revenue. It offers not only IDMS and IDMS/R (a less-than-relational DBMS according to Dr. Codd), but also ADS/OnLine (which is billed as being "more than a fourth generation language"). After limited acceptance of its Goldengate integrated package for micros, it reached an agreement with Lotus to link Symphony to Cullinet's mainframe software--a product introduced in early 1985 and not yet on the market in early 1986.

- Applied Data Research Inc. (ADR), with software revenue of about $90 million, provides Datacom/DB (which did not fare any better than IDMS/R when subjected to Dr. Codd's scrutiny) as its base product along with Ideal (a successful 4GL despite some adverse publicity), PC Datacom 2.0 (a micro-mainframe link to Datacom/DB), and PC Peer (a five-function integrated software package for the IBM PC). In late November, ADR was acquired by Ameritech, a Bell system spinoff with $8 billion in revenue and $1 billion in profits, which should make Ameritech's fellow independent competitors pause.

- Cincom Systems Inc., with $53 million in U.S. revenue ($90 million worldwide), shifted emphasis from the venerable TOTAL to an integrated set of products including TIS (a reported relational DBMS which has not been subjected to Dr. Codd's scrutiny in public), Mantis (a fourth generation language), Intellinet Query (a network management system), and PC Contact (a micro-mainframe link).

- Software AG Systems Group, Inc., with $36 million in U.S. revenues, rounds out the DBMS-oriented competitors. Having started with Adabas, Natural (a fourth generation language) has

been added along with Natural Connection (a micro-mainframe link between Adabas on the host and several micro packages).

- The three "whiz kids" of the PC-oriented competitors have revenue approximately equal to the four older DBMS-oriented firms (327 million). However, they came from different orientations and their future directions are less clear except to state that all must face the inevitability of communicating with other levels in the network hierarchy. Application development tools suited for early standalone PC applications are not going to survive as IBM moves from the SNA/DDP strategic period toward the electronic office period. Fortunately, all three seem to be maturing rapidly based on their early experience with integrated packages and micro-mainframe links.

  . Lotus Development Corporation, with $140 million in revenue, is the king of the integrated business packages with 1-2-3 (spreadsheet, data base, and graphics). Lotus 1-2-3 has an installed base of over one million copies, but the follow-on package (Symphony) only sold 100,000 copies in 1984 and customers are beginning to resist the cost of new releases of 1-2-3. (However, Symphony's link to Cullinet mainframe software was a positive step for both companies and it is unfortunate that the link was severed.)

  . Microsoft, starting with an implementation of Basic for microcomputers, has grown to a $123 million company by extending into operating systems: PC-DOS to IBM and Xenix for the UNIX market are its most significant products. Close affiliation with IBM practically assures Microsoft's success during the SNA/DDP period, but it does not mean IBM will share very much of the electronic office market.

  - Ashton-Tate had 1984 revenue of $64 million and comes from a DBMS orientation with its dBase II and III products, but attempts to compete against Lotus' Symphony with Framework have been "disappointing," with only 45,000 copies shipped in 1984.

- The early non-procedural languages (Ramis and Focus) have not experienced the type of growth exhibited by the DBMS and PC-oriented competitors (even after receiving the catchy description of fourth generation languages from James Martin). The combined U.S. software revenue of the two companies, Information Builders (Focus) and Martin Marietta Data Systems (which acquired Mathematica, the developer of Ramis), was only $69 million in 1984. The conclusion is simple--it is easier to add a 4GL to an established language (not surprising). However, both of the 4GL-oriented competitors have as a strength a loyal customer base.

  - Information Builders, Inc. had $38 million in revenue in 1984 with Focus, but it has introduced PC/Focus (designed for the IBM, TI, and Wang PCs), Foctalk, and Foccalc (a micro-mainframe link and a spreadsheet) and is porting micro functions to the mainframe as well as providing 4GL capability--a reasonably intelligent strategy in today's environment.

  - Martin Marietta Data Systems (MMDS) had $31 million in software revenue in 1984 and Ramis contributed approximately 60% of this; however, MMDS is bringing together Ramis with UFO (an application development system from Oxford Software Corporation which is also part of MMDS), and this shows some appreciation for the fact that there is not a magic solution to the productivity problem.

- In addition to those companies, which seem to be directing their strategies toward direct competition in the projected market for application develop-

ment tools, there are those who are more or less on the periphery with specific products (see Exhibit III-4). We will only comment on a few of them.

- Informatics was the subject of a hostile takeover by Sterling Software Inc. and it is INPUT's opinion that the distraction occurred at a critical time; otherwise, Informatics would have been listed with the major competitors.

- SAS produces quality product and it has been astute enough to acquire a DBMS (System 2000 from Intel), but it has been going through some growing pains and some of the technical personnel have wandered off. With careful planning, good management, and a little luck, SAS could outflank current competitors by providing integrated data/information/knowledge bases along the hierarchy.

- Candle Corporation specializes in performance monitors for IBM systems software, and predicting performance (and controlling) at the hardware/software performance level is a key element in the development of quality systems. But performance measurement tools have traditionally been the concern of operations (in other words, they are employed after the fact), and today's DSD environment is based on studied disdain for the hardware/software performance level.

- CGA Computer Inc. provides a security package (Top Secret) and security will receive renewed attention during the SNA/DDP strategic period, but today's tools do not necessarily work in tomorrow's environment and IBM has an inside track on comprehensive security systems.

- In this series of reports, INPUT has made it clear that the market for application development tools is heavily dependent upon a solid data base foundation and that those vendors with a DBMS product orientation will be the primary beneficiaries of that market growth. Those with a language orientation will

EXHIBIT III-4

FRINGE COMPETITORS

COMPANY	REVENUE ($ Millions)	PRODUCTS
Computer Associates International	$81	Performance Improvement (Apex, Optimizer)
Informatics (Sterling)	74	Programmer Aids (4GL) (Mark Series)
Pansophic Systems Inc.	45	Application Development Tools Operations Management (Panvalet, Easy Third)
SAS Institute	45	Information Management (SAS, S2000)
Candle Corporation	40	Performance Monitoring (MVS, IMS, VM)
CGA Computer Inc.	20	Security (Top Secret)
Boole & Babbage	17	Performance, Productive, and Security

lose market share to the DBMS-oriented vendors; vendors of other application development tools (such as those operating around the fringe of the ADT market) will represent a relatively small percentage of the total market.

- It has also been pointed out that IBM's primary emphasis during the SNA/DDP period will be to depend upon operating systems and DBMSs as means of establishing and maintaining control of the emerging distributed processing environment. In addition, to the degree that IBM concentrates DBMS, the market for advanced language development will be more open to other competitors. This was reflected in the forecasts in Market Analysis: Fourth Generation Languages. IBM's attention to the "other" application development tools will be directly related to the strategic importance of the particular tool. For example, it can be anticipated that IBM will be less than interested in tools to monitor and predict performance, but will consider tools to develop highly secure systems to be of strategic importance.

- Since DBMS is so important to the ADT market, we would like to review some of the thinking (or lack thereof) which has surfaced in various publications since the DBMS report of this series was published a few months ago. Essentially, there seems to be a school of thought developing which says the following:

  - The relational "craze" will give way to more "robust" (which seems to be the latest craze in terminology) systems which look more like the ANSI three-schema architecture. Therefore, pure relational systems will never achieve substantial market penetration.

  - Distributed data bases have substantial unsolved problems (agreed) and the need for them is limited. Besides, centralized processing power has been able to keep up with the demands being made on central hosts (or clusters of hosts) at a reasonable cost.

- Data base machines will continue to develop very slowly and IBM will not be putting essential DBMS (and/or operating systems) functions in microcode because non-IBM DBMS software has become too important to IBM's major customers.

- IBM does not have its data base act together and is threatened with substantial loss of market share.

- INPUT's comments on these conclusions are as follows:

  - The relational "craze" was thinking the entire world would ever be relational to begin with, but it now appears that those who embraced the "relational" cause in order to belabor IBM about its dual DBMS approach at the time DB2 was announced are now backing off. It is probable that this is the direct result of Dr. Codd taking dead aim at the relational clay pigeons which have proliferated in the marketplace (or perhaps it is just a question of some of the experts beginning to understand what a relational DBMS is).

  - Unfortunately, this sudden 180-degree reversal comes at precisely the time when the relational model is most important, and that is when distributed data bases begin to develop. Which brings us to the second point--whether there is a "true need" for distributed data bases or not, they are going to develop in the DSD environment and this will occur regardless of whether or not the problems associated with them are solved. The stringent rules associated with Dr. Codd's relational DBMS definition become especially important in maintaining integrity, and the flexibility and ease of use of a relational DBMS become essential at Levels II and III in the network hierarchy. A substantial portion of projected DBMS market is going to be associated with those levels.

  - It is INPUT's opinion that micro-mainframe links are going to place enormous processing demands on central host processors. This in turn

will encourage (and even force) both the geographic and architectural distribution of processing (the term geographic distribution refers specifically to distributed data bases, and the term architectural distribution connotes some type of data base machine). The same processing crunch will also encourage IBM to relieve the systems software overhead by putting more functions into microcode. Then the obsolete software DBMSs, both IBM's and others, will be left to die on the vine and continue to absorb any excess processing power which might have existed otherwise.

- In Market Analysis: Data Base Management Systems, INPUT stated that it did not believe that IBM was going to lose DBMS market share and gave the reasons for that conclusion. Events since that time have only tended to confirm this opinion, and it is probable that IBM will actually gain market share over the 1985-1986 timeframe.

- It has been customary to view IBM as being primarily a hardware peddler, and there were those in IBM who seriously questioned IBM's decision to unbundle software nearly 20 years ago. (The usual question was: "Who would pay for it?") However, there are a few facts which discredit this point of view.

  - Everyone knows that IBM controls the mainframe market, yet the top 10 mainframe competitors worldwide have 77% as much mainframe revenue as IBM but only derive 45% as much software revenue as IBM.

  - The top 10 independent software vendors in the U.S. only have 30% as much software revenue as IBM, and the top 50 only account for 62% of IBM's software sales.

  - If software earnings figures were available for comparison, the dominance of IBM would be even more striking. In addition, the argument that IBM's software sales are primarily derived from "captive" systems software sales is not a sign of vulnerability in other areas--it merely means competitors' products (whether ADTs or applications) depend upon IBM for their very existence.

IV OPPORTUNITIES AND CHALLENGES

A. ADT FORECAST

- INPUT projects the total application development tools market to be $10.3 billion in 1990 (see Exhibit IV-1). This will be broken down as follows:

  - $6 billion of the market will develop from DBMS-based products.

  - $3.2 billion of the market will develop from language-based products (FGLs).

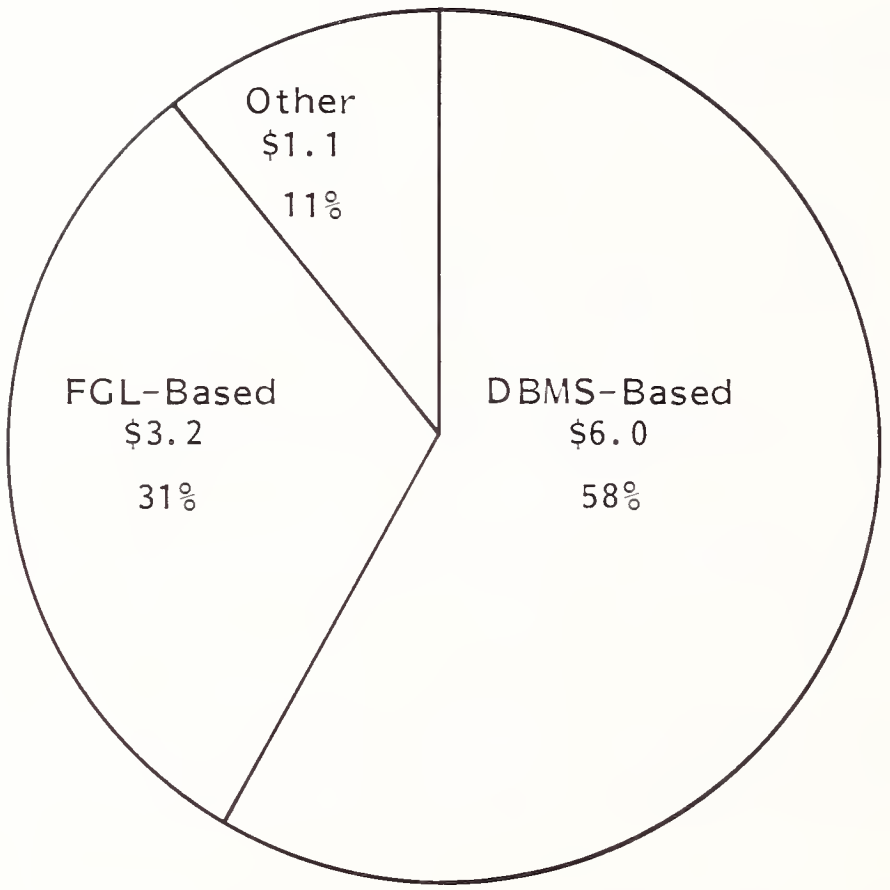
  - The remaining $1.1 billion is classified as "other," a category which includes both the specialized and highly advanced anticipated tools.

- The primary distinguishing factor of the "other" category is that the tools are directed toward the problems which remain after more generalized tools, such as DBMSs and 4GLs, are applied, and it is anticipated that most such tools can be viewed as being either complementary or supplementary to the major categories of ADTs and to vendor-provided operating systems.

- Using various systems categories as specific frames of reference, it is not difficult to isolate these remaining problems and even anticipate how tools themselves contribute to these problems. For example:

EXHIBIT IV-1

ADT FORECAST, 1990
($ Billions)



Total ADT Market: $10.3 Billion

- INPUT's productivity hierarchy (pyramid) has always emphasized that "commitment to quality" is of primary importance in any productivity improvement program; and, to the degree that tools and aids facilitate the development of "quick and dirty" systems, they can contribute to the problem. (This was the essential theme of Market Impact of New Software Productivity Techniques.)

- Using data-driven systems development methodologies (prototyping) may address the early phases of the development/life cycle systems category (requirements, specifications, etc.) and an important level of the productivity hierarchy (end-user involvement, second only to commitment to quality in importance), but they can cause serious problems in some of the quality subsets such as:

  . Objectives.

  . Measurement.

  . Auditability.

  . Validity/reliability/predictability.

- Then, of course, the balancing of GST (general systems theory) directions varies over time in terms of emphasis. (IBM's highly centralized DBMS approach may be appropriate for the SNA/DDP period, but creates substantial problems, or opportunities, in the electronic office period.)

- Numerous other potential challenges for current application development tools can be isolated by reviewing the systems categories (Appendix A of Market Analysis: Fourth Generation Languages), but perhaps none is quite so evident as the performance category which has repeatedly been emphasized in this series of reports. It is essential that productivity be viewed in terms of

performance at all four levels--hardware/software, human/machine dyad, work unit, and institutional. The emerging market for "other" application development tools will address specific performance levels and the balance across those levels.

B. COMPETITIVE CHAOS

- There is a natural tendency to extend the use of specific tools beyond their intended or practical purpose. While a certain amount of this testing of market limits is both inevitable (and even desirable), the market for application development tools is being adversely impacted by both specific and general misuse of existing tools. Although some of this is clearly the responsibility of users who have had a persistent propensity to seek one simple solution to an extremely complex problem, vendors must accept a major share of responsibility for the general "buyer beware" atmosphere which pervades the marketplace.

- The claims which have been, and are being, made for various tools, aids, techniques, approaches, and methodologies are legend going right back to that old granddaddy, COBOL. If 1% of the accummulated claims had actually been achieved there would not be a productivity problem today. Both good and bad tools suffer when a single hammer is advertised as being just right for driving tacks, nails, and pilings or a Swiss army knife is used to build a house.

- There are reputable and knowledgeable people (the two terms are not necessarily synonymous) who are stating that "anyone who speaks of productivity improvements on the order of 50% to 100% or more is a fraud." Contrast that with advertised and/or reported claims in the trade press, or even technical journals, and then think of what it means in the marketplace.

- Appendix A contains a listing of nearly 50 recently advertised DBMSs and 4GLs, and new products are being announced on practically a daily basis. It is very difficult for any user (or consultant) to make any meaningful functional analysis of this vast array of products, much less any qualitative evaluation concerning performance. There is a natural tendency to look primarily to established vendors who advertise extensively and have a solid customer base from which they can reference-sell. It will be extremely difficult for new vendors of conventional application development tools to achieve any penetration of the market.

- This chaos in the market for application development tools is the reason INPUT believes there is a substantial market for tools which will integrate and place boundaries on the use of already existing tools. Essentially, this is the "other" category which has been forecast.

C. ENVIRONMENTAL CHAOS

- INPUT has emphasized the need for market structure because of the environmental chaos which has resulted from distributed systems development. The best way to illustrate the problem is to list four of the systems categories which have been proposed for structuring the market.

  - In the DSD environment, the development structure (design, program, work unit organization, operational, and rigidity/flexibility) itself has become substantially more fluid.

    . Systems can be designed from the top down or be evolved from the bottom up.

    . Programs can be either structured or the most horrible hodge-podge imaginable. In addition, the unpredictable meanderings of

exploratory programming (expert systems) will make algorithmic programs employing GOTOs appear to be relatively structured.

. Work units established for development may be under highly centralized control (with established standards) or casual structures running both horizontally and vertically across established organizational boundaries.

. Source and target operating environments can be at any combination of levels in the network hierarchy (mainframe, minicomputer, intelligent workstation, etc.) and subject to dynamic reallocation.

. The general objectives of the developed system are also subject to change--the quick and dirty "pumpkin" is expected to change into an elegant "carriage" when the prototyping princess is ready to go the the production ball. However, the very flexibility which permitted a bumper crop of pumpkins may not meet the rigid expectations and standards of the grand event (ultimate or final system).

- The systems type category is already complicated enough by the different requirements of batch versus interactive, and the gradations are becoming more complicated as finer distinctions are made and expert systems begin to appear.

- Systems requirements are getting complex (and of broader range) as more terminals go on-line, new analytical tools are used, data bases continue to grow astronomically in terms of size and content, and programs become more complex logically (more decision rules).

- In addition, the user set is ever expanding. It is INPUT's opinion that the inexperienced, first-time users include both the "dumb" and the

"smart" and that among them are those who are more intelligent than either the sellers or developers of application development tools (or at least they have been paying the bills). These users will question the tools (DBMSs, 4GLs, etc.) in terms of both function and cost.

- The point is that the market for ADTs is determined by the development environment (at present, the trend is toward distributed systems development), and this complex and changing environment can be roughly illustrated by the systems categories in Exhibit IV-2. To expect any tool, or set of tools, to address all of the possible combinations of development structures, systems types, systems requirements, and user sets would be foolhardy, and intelligent users and reputable vendors know this. However, the tendency to overextend the barriers of reason and good sense seems to be constantly with us.

- Environmental chaos in the marketplace is a direct result of IS management looking for the magic bullet to take care of the productivity problem as manifested by the ever-growing backlog of user requests. This represents a level of naivete on the part of IS management which should be alarming because it is probable that many would not recognize the solution to their particular problem even if one were available. Then, even if they found the right tool, it would not work because the solution to the development problem (regardless of tools employed) is management, and there are not many managers around with the technical expertise to manage today's complex development projects.

D. IBM

- INPUT has often stated that chaos in the marketplace (hardware, software, technology, or the general economy) can only benefit one vendor and that is IBM. The current competitive environment in the use and/or misuse of application development tools definitely falls under the category of chaos and

EXHIBIT IV-2

CONFLICTS IN THE DSD ENVIRONMENT

- Development Structure
  - Conflicting Design Objectives and Implementation Methodologies
  - Conflicting Development Organizations and Work Units
  - Dynamic Source and Target Operating Environments
  - Prototype Versus Production Version
- Systems Type
  - Batch Versus Interactive
  - Decision Support Versus Expert
- Systems Requirement
  - Function Versus Performance
  - Data Base Size and Distribution
- User Set
  - "Dumb" Versus "Smart" Users

IBM will benefit. When users do not know what to do, they naturally turn to IBM, which has the deserved reputation for "making things work" and being around after any technological or economic upheaval.

- Last year (1985) helped strengthen IBM's competitive position in software regardless of any actual changes in market share, quality of products, or improved IBM strategies or tactics. When times get tough, and they were, the wild ducks of the data processing industry turn into homing pigeons, and home is located in Armonk, New York. There they are taught an old IBM lesson--even wild ducks are expected to fly in formation. (IBM does not care if you use UNIX as long as UNIX flies under VM.)

- The fact of the matter is that IBM's cautious approach in many areas (such as LANs and micro-mainframe links) not only makes good business sense, but good technical sense as well. There are literally times when IBM's highly centralized strategy during the current SNA/DDP strategic period makes more sense than it has in the past, and that statement is a direct result of the chaos which exists in the DSD environments.

- However, this does not mean that IBM has all the answers or even the resources to solve the problems existing in the marketplace. In fact, it is not at all certain that IBM either recognizes all of the problems or even wants to solve them. The point is that IBM is in a position to dominate the markets it chooses pretty much on its own timeframe. This is not meant to be threatening--it is a simple statement of fact which must be recognized if one is to take advantage of the opportunities which do exist.

- At the present time, IBM is going to be most aggressive in the DBMS market, less active in FGLs, and somewhat disinclined to pursue opportunities in the "other" category.

V CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

- IBM strategic periods, with GST trends and product emphasis, have been mentioned earlier in this series of reports and were established as a frame of reference in Market Impacts of IBM Software Strategies. There are usually good reasons why IBM's product strategy lags that which is technically possible--even IBM has limited resources and cannot do everything at once. Because of the wide range of IBM product offerings, a more complex technical challenge confronts IBM, and IBM may have legitimate concerns about the application of the technology, etc. However, there is no question that IBM's strategy is based primarily upon its best business interests. For example, it would be unrealistic to expect IBM to embrace optical storage (at one-tenth the cost) as a replacement for magnetic disk which is currently essential to its revenue (and earnings) growth.

- There is no question that IBM's business interests are primarily associated with hardware sales, and the constant complaint that IBM is only interested in selling iron is about as meaningful as saying General Motors is only interested in selling automobiles as opposed to batteries, tape decks, and horns. However, there is one important difference between batteries, tape decks, and horns and IBM systems software--IBM systems software determines the size and price of the vehicle required to operate it. No car has ever been so

loaded down with accessories that the engine could not power it out of the driveway, but the classic thrashing problem with early virtual storage systems did just that.

- IBM's revenue from software is only the tip of the iceberg. For every dollar of visible software revenue, there are probably five dollars of additional hardware necessary to run it. If there were ever the serious threat of an alternative to MVS/XA, IBM could not only afford to give their systems software away--they could practically afford to pay their customers to run it. In other words, IBM has enormous residual hardware income from systems software and no one can compete against them head-to-head on the basis of price unless IBM permits them to do so.

- In the specific case of application development tools, IBM can view the mystical applications backlog as pent-up demand for hardware, and if anyone really solved the problem and the backlog disappeared, there is no question who would derive the most revenue from the solution. On the other hand, IBM can have it both ways because the productivity solutions sell even more hardware. Programmer workstations with subsecond response time, dedicated development systems, relational data base systems, and production systems written in 4GLs all add substantially to hardware sales. Therefore, IBM has every motivation to see that productivity in the systems development process is improved.

- However, the computer-communications systems being produced out of this process have no value unless they in turn improve the productivity of the enterprise for which they are being developed. When reviewed against the performance levels necessary to measure enterprise productivity, there are points of divergence between improved productivity and IBM's business objectives.

  - Hardware/software performance is an important, and even essential, aspect of systems quality which all vendors (and especially IBM because

of its hardware orientation) prefer to discount when discussing enterprise productivity and the use of application development tools. There is a great and increasing need for application development tools which will predict, control, and tune (improve) hardware/software performance at (and across) all levels in the network hierarchy and during all phases of the development/life cycle.

- Human/machine dyad performance poses a particular challenge to IBM not only because of the hardware/software side of the dyad, but also because of the inherent difficulty humans have interfacing with the complexity of IBM's software strategy (multiple operating systems, data base systems, office automation products, etc.). Being everything to everybody is not easy, and even with the best of interfaces, the human being is aware that he/she is driving a bulldozer rather than a sports car. Application development tools (or environments) which smooth interfaces and improve responsiveness at the human/machine dyad will find ready acceptance from the IBM customer base (and even IBM) provided the integrity of IBM systems software and products is maintained.

- Work unit network performance will be heavily dependent upon systems which facilitate the flow of quality information between and among individuals. Application development tools which assure the quality of the information flowing over the network (which can be geographically distributed) with minimal attention from the applications developer are necessary. In other words, questions of data/information integrity and security, queuing problems at nodes (including workstations), and complex resource allocation problems cannot be left to individual development projects using today's application development tools. (It is INPUT's opinion that IBM's current strategy for addressing LANs (departmental processing) is stopgap at best and the issue will not really be addressed until the early 1990s.)

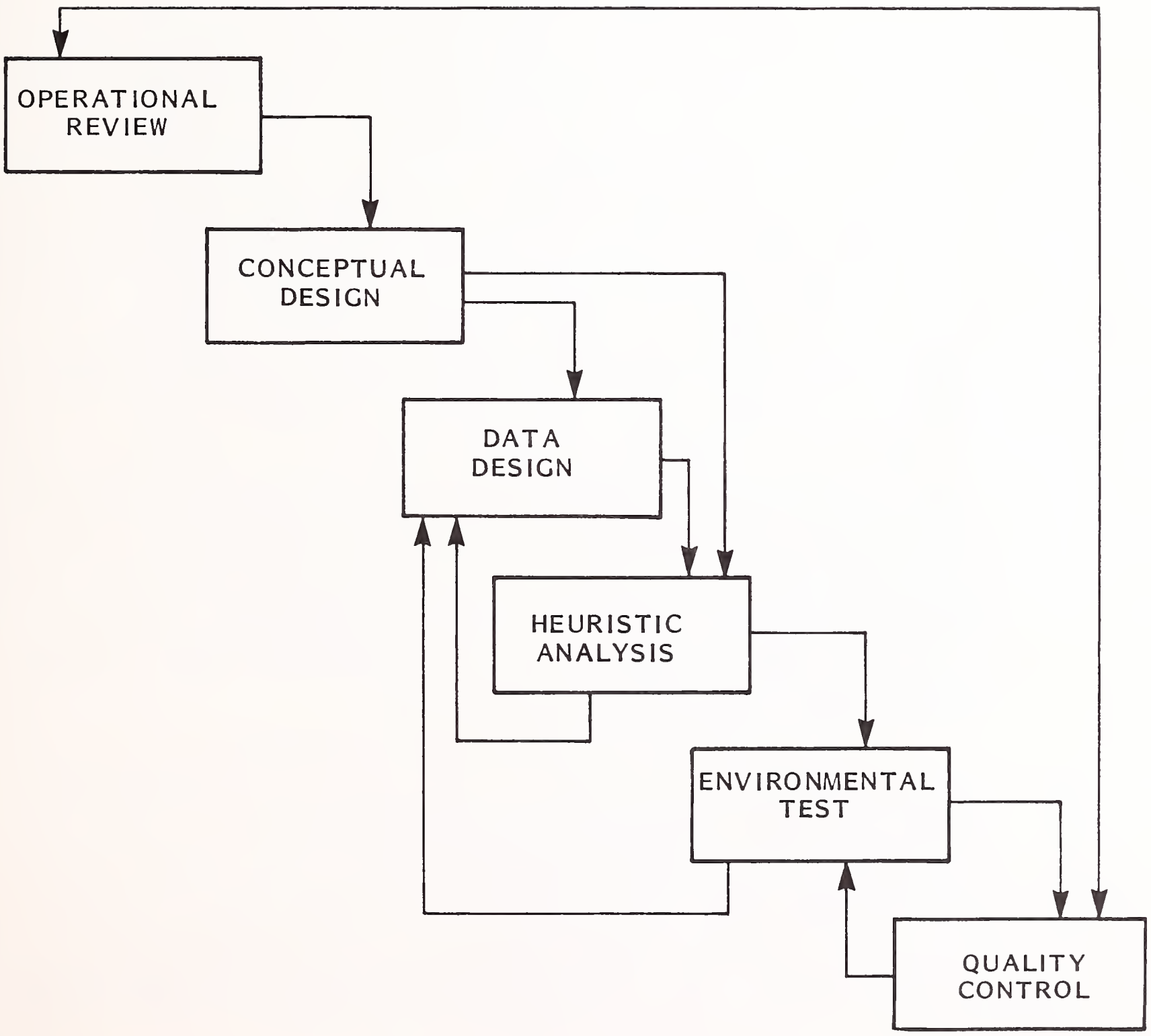
- Institutional performance cannot be expected to improve merely because enormous data bases and general purpose DBMSs are installed. Improved tools for data/information analysis (including data reduction) are required before current DBMSs and FGLs can be interfaced with knowledge-based systems without the probability that expert systems will wander off on endless searches and/or return "solutions" which are so far from optimal that they enhance (rather than diminish) the problem. Vendors encourage the global view that information is a corporate asset with little regard or appreciation for the tools necessary to facilitate intelligent use of operations research and statistical techniques, much less the inadequacy of conventional mathematics for solving many business problems. In other words, it is not in the best interests of vendors of current "solutions" to recognize their inadequacy in improving institutional performance at the highest level. (See Impact of Office Systems on Productivity for more detailed analysis of this problem.)

- In Software Productivity Tools: Update and Tools (1983), INPUT presented a schematic of data-driven prototyping (see Exhibit V-1). It is INPUT's conclusion that numerous tools already exist to facilitate the data design and heuristic analysis steps. What is needed now are tools which facilitate the environmental test and quality control steps.

- The final conclusion is that tools for quality control (and real productivity improvement) are essentially complementary and supplementary to the highest levels of the software hierarchy--specifically to SNA and operating systems. Since these areas will be emphasized by IBM during the SNA/DDP strategic period (along with DBMS), some of the problems will be addressed by IBM enhancements at those levels. However, it is INPUT's opinion that IBM progress will be slow because priorities which promote hardware/software will be higher than for those involved specifically with quality and performance improvement. Therefore, IBM's presence in the "other" ADT market will be negligible and it will be left to the independents (with whatever work IBM

EXHIBIT V-1

SIX ITERATIVE STEPS IN DATA-DRIVEN PROTOTYPING



does being classified under operating systems or DBMS). It is also probable that it will be especially attractive for the smaller vendors who are not already locked in a life and death struggle for survival in the DBMS and FGL markets.

B. RECOMMENDATIONS

- Recognize (accept), analyze, and understand the impact of current application development tools on all four levels of the performance systems category. (Special attention should be given to adverse impacts at the hardware/software level because it is not high on IBM's priority list.) Develop performance monitors and models which will assist in predicting and measuring hardware/software performance during iterations through the quality control and environmental test steps of the prototyping process (see Exhibit V-1). Notice that operational review and quality control are directly connected in the diagram--commitment to quality should be the primary emphasis throughout the system life cycle.

- It is INPUT's belief that queuing network models will become desirable and even essential in predicting and controlling acceptable performance of applications structured in the manner depicted in Exhibit III-1. For those not already familiar with the theory of queuing networks, some research is recommended before developing performance measurement tools for either internal product development or as an end product.

- There is also another area which requires research if better application development tools for quality assurance are to be developed. For lack of a better term, INPUT has used entropy to describe the natural tendency toward disorder of both data and information. It is important because it adversely impacts both hardware/software performance (more processing power is required to maintain order) and institutional performance (more information is

not better when management gets conflicting reports--more human analysis is required to determine the relative quality of the information). Practical tools for data/information entropy measurement and control are becoming increasingly important and should have a ready market, but additional research is needed. (See Market Impact of New Software Productivity Techniques, INPUT, 1984, for additional information.)

- In addition to the tools required to project, measure, and control performance characteristics of the systems being developed (or evolved), there is a need for substantially improved cost accounting tools for purposes of estimating, project control, operating cost analysis, and determination of the "residual costs." It is recommended that such cost analysis systems be integrated with the top three levels of the software hierarchy (SNA, operating systems, DBMS), but they should not be confused with conventional systems accounting information; e.g., the costs at the human/machine dyad would include personnel costs.

- Finally, it is concluded that IBM's endorsement of VM has opened up a vast array of opportunities for improving the environment in which applications systems are developed. If the objectives of operating systems are remembered, and current implementations are evaluated against those objectives, numerous additional targets of opportunity should present themselves. It is recommended that extant operating systems implementations be analyzed against original objectives and with the knowledge of the current DSD environment.

  - There are three broad objectives of operating systems:

    . Maximum ease of use.

    . Maximum use of equipment.

  - . Effective development, testing, and introduction of new system functions without interfering with service.

- There are five major abstract areas of operating systems implementation:

  - . Process (multiprogramming, multiprocessing, transaction processing, timesharing).

  - . Storage management.

  - . Protection and security.

  - . Resource allocation (the interaction between service and performance).

  - . System structure (e.g., VM facilitates the introduction of new operating systems functions without disruption).

  - In addition, INPUT has defined a sixth area of layered hardware/firmware/software (H/F/S) implementation which pinpoints additional areas such as data base machines.

- Specific targets of opportunity are numerous and varied, and INPUT suggests that none is sacred. The first 1986 strategic issue report from INPUT will analyze these opportunities in more detail.

APPENDIX A

REPRESENTATIVE DBMS AND 4GL PRODUCTS

VENDOR	DBMS AND 4GL PACKAGES
Advanced Data Management	DRS DBMS
Advanced Relational Technology	CENTRE
Applied Data Research	DATACOM/DB, IDEAL DATA DICTIONARY
Aritficial Intelligence	INTELLECT
Burroughs Corporation	LINC
CGI Systems	PACBASE
Cincom Systems	TIS, ULTRA, MANTIS
Cognos Corporation	POWERHOUSE
Computer Associates, Inc.	CA-UNIVERSE, CA-EXECUTIVE
Computer Corp. of America	MODEL 204 DBMS, PROD/NET WORKSHOP 204
Comshare	SYSTEM W
Cortex	APPLICATION FACTORY
Cullinet Software	IDMS/R, ADS/O, GOLDENGATE
D&B Computing Services	NOMAD 2
Digital Equipment Corporation	VAX INFO ARCH., VAX Rdb
Henco Software	INFO
Hewlett-Packard Co.	IMAGE
IBM Corporation	IMS, SQL/DS, DB2, IMS ADF II, CSP
Information Builders	FOCUS, PC/FOCUS
I.P. Sharp Associates	VIEWPOINT
KnowledgeWare, Inc.	DATA DESIGNER II
Leading Software Technologies	INTELLIGENT ASSISTANT

Continued

APPENDIX A (Cont.)

REPRESENTATIVE DBMS AND 4GL PRODUCTS

VENDOR	DBMS AND 4GL PACKAGES
Linkware Corp.	LINKWARE INFORMATION SERVER
Logica Database Products	RAPPORT
Mainstay Software	MAINSTAY
Management and Planning Software	THEMIS
Management Science America	INFORMATION EXPERT
Martin Marietta Data Systems	RAMIS II, CONSENSUS, UNISON
McCormack & Dodge	MILLENIUM
Multiplications Software	ACCOLADE, IMAGINE
Oracle Corporation	ORACLE
Pro Computer Science	PRO-IV
Relational Database Systems	INFORMIX
Relational Technology	INGRES
Seed Software Corp.	SEED
Signal Technology	OMNIBASE, SMARTSTAR
SIR, Inc.	SIR DBMS
Software AG of N.A.	ADABAS, NATURAL
Software House	SYSTEM 1022, 1032
Sperry Corporation	MAPPER
Sybase	DATASERVER, DATAWORKBENCH
Tarkenton Software	GAMMA
Teradata Corporation	DBC/1012

Continued

REPRESENTATIVE DBMS AND 4GL PRODUCTS

VENDOR	DBMS AND 4GL PACKAGES
3CI	InFoCen
Transform Logic Corp.	TRANSFORM
Unify Corporation	UNIFY
Wang Laboratories	PACE

APPENDIX B: DEFINITIONS

- INFORMATION SERVICES--Computer-related services involving one or more of the following:

  - Processing of computer-based applications using vendor computers (called "processing services").

  - Services that assist users in performing functions on their own computers or vendor computers (called "software products" and/or "professional services").

  - Services that utilize a combination of hardware and software, integrated into a total system (called "turnkey systems").

A. USER EXPENDITURES

- All user expenditures reported are "available" (i.e., noncaptive, as defined below).

- NONCAPTIVE INFORMATION SERVICES USER EXPENDITURES - Expenditures paid for information services provided by a vendor that is not part of the same parent corporation as the user.

- CAPTIVE INFORMATION SERVICES USER EXPENDITURES - Expenditures received from users who are part of the same parent corporation as the vendor.

B. DELIVERY MODES

- PROCESSING SERVICES - This category includes remote computing services, batch services, processing facilities management, and value-added networks (VANs).

  - REMOTE COMPUTING SERVICES (RCS) - Providing computer processing to a user by means of terminal(s) at the user's site(s) connected by a data communications network to the vendor's central computer. There are five submodes of RCS, including:

    . Interactive - Characterized by the interaction of the user with the system for the purpose of problem-solving, data entry, and/or transaction processing. The user is on-line to the program/files. Computer response is usually measured in seconds or fractions of a second.

    . Remote Batch - A service in which the user hands over control of a job to the vendor's computer, which schedules job execution according to priorities and resource requirements. Computer response is usually measured in minutes or hours.

    . Data Base - Characterized by the retrieval and processing of information from a vendor-provided data base. The data base may be owned by the vendor or a third party.

- User Site Hardware Services (USHS) - Offerings provided by RCS vendors that place programmable hardware on the user's site (rather than in the vendor's computer center). USHS offers access to a communications network, access through the network to the RCS vendor's larger computers, and significant software as part of the service.

- BATCH SERVICES - This includes computer processing performed at vendors' sites of user programs and/or data that are physically transported (as opposed to electronically by telecommunications media) to and/or from those sites. Data entry and data output services, such as keypunching and computer output microfilm processing, are also included. Batch services include those expenditures by users who take their data to a vendor site that has a terminal connected to a remote computer for the actual processing.

- PROCESSING FACILITIES MANAGEMENT (PFM) (also referred to as "resource management" or "systems management") - The management of all or a major part of a user's data processing functions under a long-term contract (more than one year). This would include both remote computing and batch services. To qualify as PFM, the contractor must directly plan, control, operate, and own the facility provided to the user, either on-site, through communications lines, or in a mixed mode.

- VALUE-ADDED NETWORKS (VANs) - VANs typically involve common carrier network transmission facilities that are augmented with computerized switching. These networks have become associated with packet-switching technology because the public VANs that have received the most attention (e.g., Telenet and TYMNET) employ packet-switching techniques. However, other added data service features such as store-and-forward message switching, terminal interfacing, error detection and correction, and host computer interfacing are of equal importance.

- Processing services are further differentiated as follows:

  . Cross-industry services involve the processing of applications that are targeted to specific user departments (e.g., finance, personnel, sales) but that cut across industry lines. Most general ledger, accounts receivable, payroll, and personnel applications fall into this category. Cross-industry data base services, for which the vendor supplies the data base and controls access to it (although it may be owned by a third party), are included in this category. General-purpose tools such as financial planning systems, linear regression packages, and other statistical routines are also included. However, when the application, tool, or data base is designed for specific industry use, then the service is industry-specific (see below).

  . Industry-specific services provide processing for particular functions or problems unique to an industry or industry group. Specialty applications can be either business or scientific in orientation. Industry-specific data base services, for which the vendor supplies the data base and controls access to it (although it may be owned by a third party), are also included under this category. Examples of industry-specific applications are seismic data processing, numerically controlled machine tool software development, and demand deposit accounting.

  . Utility services are those for which the vendor provides access to a computer and/or communications network with basic software that enables users to develop and/or process their own systems. These basic tools often include terminal-handling software, sorts, language compilers, data base management systems, information retrieval software, scientific library routines, and other systems software.

- SOFTWARE PRODUCTS - This category includes users' purchases of applications and/or systems software that is sold by vendors as standard products intended for use by different organizations. Included as user expenditures are lease and purchase expenditures, as well as fees for work performed by the vendor to implement and maintain the package (when such fees are either bundled as part of the product price or offered on an annual subscription basis). Fees for work related to education, consulting, and/or custom modification of software products are counted as professional services, provided such fees are charged separately from the price of the software product itself. There are several subcategories of software products, including:

  - APPLICATIONS SOFTWARE PRODUCTS - Software that performs a specific function directly related to solving a business or organizational need. Applications software provides information directly for use by the end user. Applications software products classifications are:

    . Cross-Industry Products - Used in multiple-user industry sectors. Examples are payroll, inventory control, and financial planning.

    . Industry-Specific Products - Used in a specific industry sector such as banking and finance, transportation, or discrete manufacturing. Examples are demand deposit accounting, airline scheduling, and materials resource planning.

  - SYSTEMS SOFTWARE PRODUCTS - Software that enables the computer/communications system to perform basic functions, which are interim steps to providing the end user with "answers" sought. Systems software product classifications are:

    . Systems Control Products - These products function during applications program execution to manage the computer system

resource. Examples include operating systems, communications monitors, and emulators.

- Data Center Management Products - These products are used by operations personnel to manage the computer system resources and personnel more effectively. Examples include performance measurement, job accounting, computer operations scheduling, and utilities.

- Application Development Products - These products are used to prepare applications for execution by assisting in design, programming, testing, and related functions. Examples include languages, sorts, productivity aids, data dictionaries, data base management systems, report writers, and retrieval systems.

- PROFESSIONAL SERVICES - This category is made up of services in the following categories:

  - SOFTWARE DEVELOPMENT - This service develops a software system on a custom basis. It includes one or more of the following: user requirements, system design, contract, and programming.

  - EDUCATION AND TRAINING SERVICES - These services help people acquire new skills, techniques, or knowledge related to computers. This definition does not include services to educational institutions. (This latter market is included in the education (industry-specific) segment.)

  - CONSULTING SERVICES - Consultants advise clients on computer-related issues that are usually management oriented. Feasibility studies and computer audits are examples of services provided.

- PROFESSIONAL SERVICES FACILITIES MANAGEMENT (PSFM) - This is counterpart to processing facilities management, except that in this case the computers are owned by the client, not the vendor; the vendor provides human resources to operate and manage the client facility.

- TURNKEY SYSTEMS (also known as Integrated Systems) - A turnkey system is an integration of systems and applications software with hardware, packaged as a single entity. The value added by the vendor is primarily in the software. Most CAD/CAM systems and many small business systems are turnkey systems. This does not include specialized hardware systems such as word processors, cash registers, or process control systems. Nor does it include Embedded Computer Resources for military applications. Turnkey systems are available either as custom or packaged systems.

  - Turnkey systems revenue is divided into two categories.

    . Industry-Specific systems--that is, systems that serve a specific function for a given industry sector such as automobile dealer parts inventory, CAD/CAM systems, or discrete manufacturing control systems.

    . Cross-Industry systems--that is, systems that provide a specific function that is applicable to a wide range of industry sectors such as financial planning systems, payroll systems, or personnel management systems.

  - Revenue includes hardware, software, and support functions.

- SYSTEMS INTEGRATION - Services associated with systems design, integration of computing components, installation and acceptance of computer/communications systems. Systems integration can include one or more of the major information services delivery modes--professional services, turnkey systems, and software products. System components may be furnished by

separate vendors (not as an integrated system by one vendor, called the prime contractor); services may be furnished by a vendor or by a not-for-profit organization. Integration services may be provided with related engineering activities, such as SE&I (Systems Engineering and Integration) or SETA (Systems Engineering and Technical Assistance).

C. HARDWARE/HARDWARE SYSTEMS

- HARDWARE - Includes all computer/communications equipment that can be separately acquired, with or without installation by the vendor, and not acquired as part of a system.

  - PERIPHERALS - Includes all input, output, communications, and storage devices, other than main memory, that can be locally connected to the main processor and generally cannot be included in other categories, such as terminals.

  - INPUT DEVICES - Includes keyboards, numeric pads, card records, bar-code readers, lightpens and trackballs, tape readers, position and motion sensors, and A-to-D (analog-to-dialog) converters.

  - OUTPUT DEVICES - Includes printers, CRTs, projection television screens, microfilm processors, digital graphics, and plotters.

  - COMMUNICATION DEVICES - Modems, encryption equipment, special interfaces, and error control.

  - STORAGE DEVICES - Includes magnetic tape (reel, cartridge, and cassette), floppy and hard disks, solid state (integrated circuits), and bubble and optical memories.

- TERMINALS - There are three types of terminals:

  - USER PROGRAMMABLE (also called "intelligent terminals"):

    . Single-station or standalone.

    . Multistation-shared processor.

    . Teleprinter.

    . Remote batch.

  - USER NONPROGRAMMABLE:

    . Single-station.

    . Multistation-shared processor.

    . Teleprinter.

  - LIMITED FUNCTION - Originally developed for specific needs, such as POS (point-of-sale), inventory data collection, controlled access, etc.

- HARDWARE SYSTEMS - Includes all processors, from microcomputers to super (scientific) computers. Hardware systems require type- or model-unique operating software to be functional, but the category excludes applications software and peripheral devices, other than main memory and processor or CPUs not provided as part of an integrated (turnkey) system.

  - MICROCOMPUTER (or personal computer or PC) - Combines all of the CPU, memory, and peripheral functions of an 8- or 16-bit computer on a chip, in the form of:

. Integrated circuit package.

. Plug-in board with more memory and peripheral circuits.

. Console--including keyboard and interfacing connectors.

. Personal computer with at least one external storage device directly addressable by CPU.

- MINICOMPUTER - Usually a 12-, 16- or 32-bit computer, which may be provided with limited applications software and support and may represent a portion of a complete large system.

. Personal business computer.

. Small laboratory computer.

. Nodal computer in a distributed data network, remote data collection network, connected to remote microcomputers.

- MAINFRAME - Typically a 32- or 64-bit computer, with extensive applications software and a number of peripherals in standalone or multiple CPU configurations for business (administrative, personnel, and logistics) applications, also called a General-Purpose Computer.

. Large computer mainframes are presently centered around storage controllers but are likely to become bus-oriented and to consist of multiple processors (CPUs) or parallel processors; they are intended for structured mathematical and signal processing and are generally used with general-purpose von-Newmann-type processors for system control.

. Supercomputer mainframes are high-powered processors with numerical processing throughout that is significantly greater than the largest general-purpose computers, with capacities in the 10-50 MFLOPS (million floating point operations per second) range, in two categories:

- REAL TIME - Generally used for signal processing.

- NONREAL TIME - For scientific use, with maximum burst-mode (but sustained speed) capacities of up to 100 MFLOPS, in one of three configurations:

  . Parallel processors.

  . Pipeline processors.

  . Vector processors.

  Newer supercomputers--with burst modes approaching 300 MFLOPS, main storage size up to 10 million words, and on-line storage in the one-to-three gigabyte class--are also becoming more common.

- EMBEDDED COMPUTER - Dedicated computer system designed and implemented as an integral part of a weapon or weapon system, or platform, that is critical to a military or intelligence mission, such as command and control, cryptological activities, or intelligence activities. Characterized by MIL SPEC (military specification) appearance and operation, limited but reprogrammable applications software, and permanent or semipermanent interfaces. May vary in capacity from microcomputers to parallel-processor computer systems. Information services forecasts in this report do not include applications for this type of computer.

D. TELECOMMUNICATIONS

- NETWORKS - Interconnection services between computing resources. Provided on a leased basis by a vendor to move data and/or textual information from one or more locations to one or more locations.

  - COMMON CARRIER NETWORK (CCN) - Provided via conventional voice-grade circuits and through regular switching facilities (dial-up calling) with leased or user-owned modems (to convert digital information to voice-grade tones) for transfer rates between 150 and 1,200 baud.

  - VALUE-ADDED NETWORK (VAN) - (See listing under Section B, Delivery Modes.)

  - LOCAL AREA NETWORK (LAN) - Restricted limited-access network between computing resources in a relatively small (but not necessarily contiguous) area, such as a building, complex of buildings, or buildings distributed within a metropolitan area. One of the two types:

    . BASEBAND - Voice bandwidth at voice frequencies (same as telephone, teletype system) limited to a single sender at any given moment and limited to speeds of 75 to 1,200 baud, in serial mode.

    . BROADBAND - Employs multiplexing techniques to increase carrier frequency between terminals, to provide:

      - Multiple (simultaneous) channels via FDM (Frequency Division Multiplexing).

- Multiple (time-sequenced) channels via TDM (Time Division Multiplexing).

- High-speed data transfer rate via parallel mode at rates of up to 96,000 baud (or higher, depending on media).

- TRANSMISSION MEDIA - Varies with the supplier (vendor) and with the distribution of the network and its access mode to the individual computing resource location.

  - MODE - may be either:

    . ANALOG - Typified by the predominantly voice-grade network of AT&T's DDD (Direct Distance Dialing) and by operating telephone company distribution systems.

    . DIGITAL - Where voice, data, and/or text are digitized into a binary stream.

  - MEDIA varies with distance, availability, and connectivity:

    . WIRE - Varies from earlier single-line teletype networks to two-wire standard telephone (twisted pair) and balanced line to four-wire full-duplex balanced lines.

    . CARRIER - Multiplexed signals on two-wire and four-wire networks to increase capacity by FDM.

    . COAXIAL CABLE - HF (High Frequency) and VHF (Very High Frequency), single frequency or carrier-based system that requires frequent reamplification (repeaters) to carry the signal any distance.

- MICROWAVE - UHF (Ultra High Frequency) multichannel, point-to-point, repeated radio transmission, also capable of wide frequency channels.

- OPTICAL FIBER - Local signal distribution systems employed in limited areas, using light-transmitting glass fibers and using TDM for multichannel applications.

- SATELLITES - Synchronous earth-orbiting systems that provide point-to-point, two-way service over significant distances without intermediate amplification (repeaters), but requiring suitable groundstation facilities for up- and down-link operation.

- CELLULAR RADIO - Network of fixed, low-powered, two-way radios that are linked by a computer system to track mobile phone/data set units; each radio serves a small area called a cell. The computer switches service connection to the mobile unit from cell to cell as the unit moves among the cells.

E. OTHER CONSIDERATIONS

- When questions arise about the proper place to count certain user expenditures, INPUT addresses them from the user viewpoint. Expenditures are then categorized according to what users perceive they are buying.

- The standard industrial classification (SIC) codes are used to define the economic activity contained in generic sectors such as process manufacturing, insurance, or transportation.

About INPUT

INPUT provides planning information, analysis, and recommendations to managers and executives in the information processing industries. Through market research, technology forecasting, and competitive analysis, INPUT supports client management in making informed decisions. Continuing services are provided to users and vendors of computers, communications, and office products and services.

The company carries out continuous and in-depth research. Working closely with clients on important issues, INPUT's staff members analyze and interpret the research data, then develop recommendations and innovative ideas to meet clients' needs. Clients receive reports, presentations, access to data on which analyses are based, and continuous consulting.

Many of INPUT's professional staff members have nearly 20 years' experience in their areas of specialization. Most have held senior management positions in operations, marketing, or planning. This expertise enables INPUT to supply practical solutions to complex business problems.

Formed in 1974, INPUT has become a leading international planning services firm. Clients include over 100 of the world's largest and most technically advanced companies.

Offices

NORTH AMERICA

Headquarters
1943 Landings Drive
Mountain View, CA 94043
(415) 960-3990
Telex 171407

New York
Parsippany Place Corp. Center
Suite 201
959 Route 46 East
Parsippany, NJ 07054
(201) 299-6999
Telex 134630

Washington, D.C.
11820 Parklawn Drive
Suite 201
Rockville, MD 20852
(301) 231-7350

EUROPE

United Kingdom
INPUT
41 Dover Street
London W1X 3RB
England
01-493-9335
Telex 27113

Italy
Nomos Sistema SRL
20127 Milano
Via Soperga 36
Italy
Milan 284-2850
Telex 321137

Sweden
Athena Konsult AB
Box 22232
S-104 22 Stockholm
Sweden
08-542025
Telex 17041

ASIA

Japan
ODS Corporation
Dai-ni Kuyo Bldg.
5-10-2, Minami-Aoyama
Minato-ku,
Tokyo 107
Japan
(03) 400-7090
Telex 26487

INPUT®
Planning Services For Management

INPUT®
1943 Landings Drive, Mountain View, CA 94043, (415) 960-3990