COMPUTER AIDED SOFTWARE ENGINEERING IN EUROPE 1987 - 1992

INPUT

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
1280 Villa Street
Mountain View, CA 94041
(415) 961-3300
Telex: 171407
Fax: (415) 961-3966

New York
Parsippany Place Corp. Center
Suite 201
959 Route 46 East
Parsippany, NJ 07054
(201) 299-6999
Telex: 134630
Fax: (201) 263-8341

Washington, D.C.
8298C, Old Courthouse Rd.
Vienna, VA 22180
(703) 847-6870
Fax (703) 847-6872

EUROPE

United Kingdom
INPUT
41 Dover Street
London W1X3RB
England
01-493-9335
Telex 27113
Fax 01-629-0179

ASIA

Japan
FKI
Future Knowledge Institute
Saida Building,
4-6, Kanda Sakuma-cho
Chiyoda-ku,
Tokyo 101,
Japan
03-864-4026
Fax: 011-03-864-4114

INPUT®
Planning Services for Management

DECEMBER 1987

COMPUTER AIDED SOFTWARE ENGINEERING IN EUROPE

1987-1992

INPUT®

41 Dover Street, London W1X3RB 01-493-9335

Researched by
INPUT
41 Dover Street
London W1X3RB
England

Published by
INPUT
1280 Villa Street
Mountain View, CA 94041-1194
U.S.A.

Information Service Programs in Europe (ISPE)

Computer Aided Software Engineering in Europe, 1987 - 1992

SDTE • 529 • 1987

Abstract

Computer Aided Software Engineering (CASE) has been heralded for many years as the answer to the maiden's prayer in the software industry to the perennial problems of chronic staff shortages and developing quality software on time, within a budget that meets business requirements.

This report examines trends, issues, and opportunities for both user and vendor participants in the market for CASE software products and services. Developments in country markets, industry sectors, and product/service market segments are reviewed together with an analysis of key issues impacting the development of the user environment. Forecasts of country and market segment size and growth are also included.

The report also reviews the dynamics and evolutionary trends in the competitive environment together with an analysis of some leading CASE products and vendor organisations.

Recommendations for existing and potential CASE vendors and users are provided.

The report contains 181 pages, including 71 exhibits.

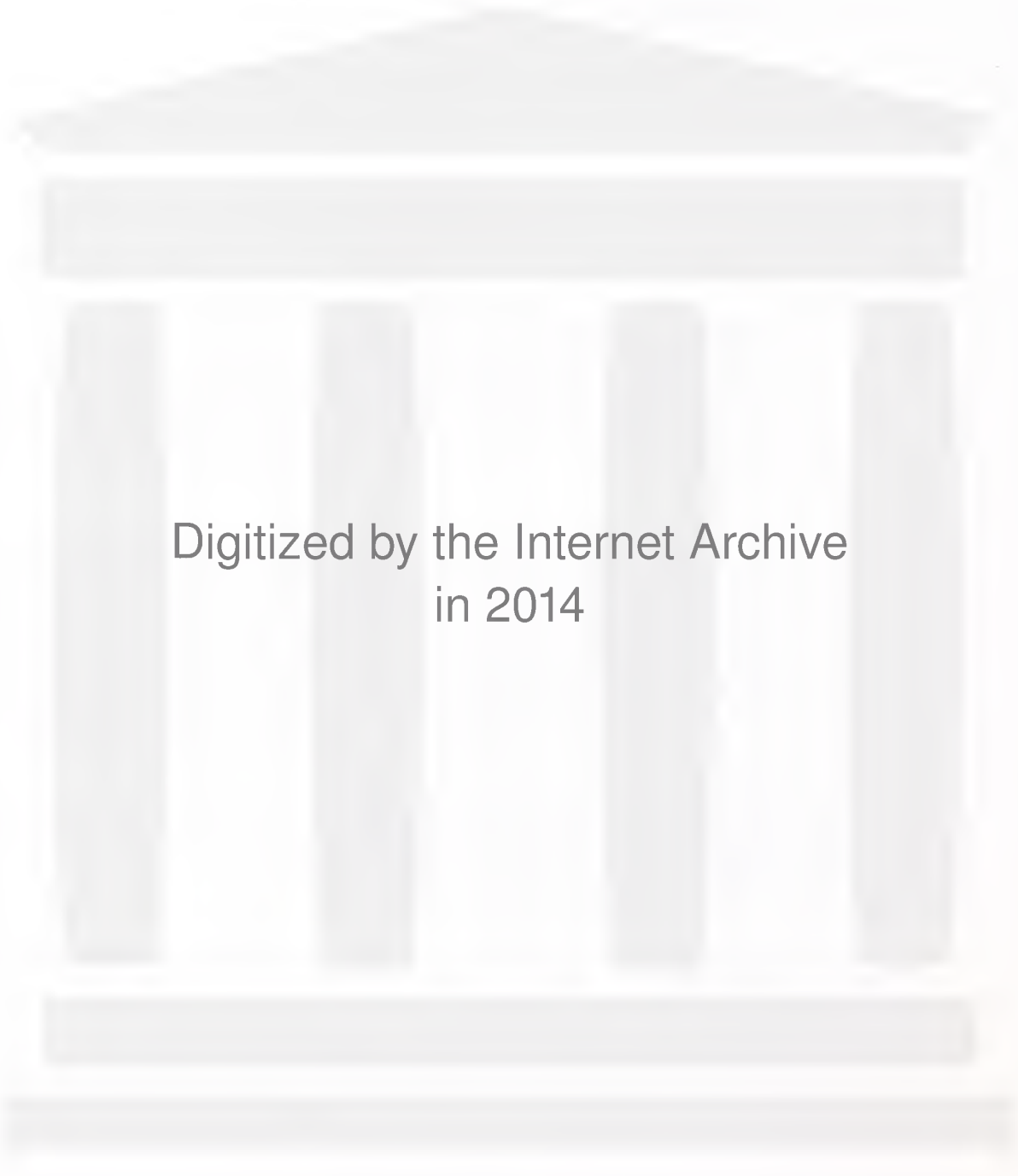


Table of Contents

I Introduction 1

A. Scope 1
B. Objectives 3
C. Methodology 4
D. Report Structure 6

II Executive Overview 7

A. Computer Aided Software Engineering 7
B. The Industry Crisis 8
C. The Catalyst to Change 10
D. A Window of Opportunity 10
E. Evolution Not Revolution 12
F. Strategic Directions for Vendors 13
G. INPUT's Solution for Users 15

III Market Analysis and Forecast 19

A. Market Overview, Structure, and Evolution 19
1. Introduction - The Software Development Crisis 19
2. Market Structure and Development 20
B. Market Forecast 26
1. Forecast Definition 26
2. Market Pressures Influence Size and Growth 32
a. Systems Integration—A Major Driving Force 33
b. Examples of SI Contracts Awarded/To Be Awarded in 1987 38
c. New Methods of Working—A Major Inhibitor 39
3. Market Size and Growth by Country and Delivery Mode 42
C. Trends, Issues, and Opportunities in Key Market Segments 44
1. Overview 44
2. Country Market Commentary 45
a. U.S.A. 45
b. West Germany 45

Table of Contents (Continued)

c. Italy 45
d. United Kingdom 46
e. France 47
f. Scandinavia 47
g. Benelux 48
h. Rest of Europe 48
3. Market Sector Commentary 48
4. Delivery Mode Commentary 50
a. Professional Services 50
i. Training 50
ii. Consultancy 53
b. Software Products 56

IV User Environment 63

A. Macro Issues–User Environment 64
1. The Challenge of Integration 64
2. The Search for Productivity 65
3. The Management of People 65
B. Trends in the Applications Backlog 66
C. Users' Perceptions of Major Constraints on Their Ability to Reduce the Applications Backlog 69
D. Users' Perceptions of Major Factors Enabling Reduction of Applications Backlogs 71
E. Usage and Attitude towards Integrated Project Support Environments (IPSEs) 72
F. Structured Methodologies—Market Penetration and Opportunities 73
1. West Germany 73
2. United Kingdom 73
3. France 75
4. Scandinavia 76
5. Italy 76
6. Benelux 76
7. General Market Commentary 76
8. Summary Review of Methodologies 78
9. The Future—CASE Tools and Methodologies 80

Table of Contents (Continued)

G. Users' Attitudes towards CASE 80
1. The Case Against CASE 81
2. The Case For CASE 84
H. The Implementation of CASE 88
I. Incremental Non-CASE Approaches towards Systems Development Productivity—Languages, Programme Development Tools, and DBMS 92
J. Users' Ratings of Potential Approaches towards Improving Data Processing Development Productivity 100
K. Users' Ratings of Potential Approaches towards Supporting End-User Development Productivity 104

V Competitive Environment 109

A. Market Overview 109
1. CASE Product Evolution 109
a. Schematic Design Products 109
b. Code Generators 111
c. Complete CASE Systems 112
d. The Way Forward—Integrated Project Support Environments 113
e. The Market Will Stabilise into Two Segments 114
2. A Worldwide Market 114
3. The United States—A Major Opportunity 115
B. Some U.S.-Based Players 115
1. Cortex Corporation 117
a. General Background 117
b. Products 117
i. Application Factory 117
ii. CorVision 117
2. Knowledgeware Inc./Arthur Young 118
a. General Background 118
b. Product 118
c. Users 119
3. Texas Instruments 119
a. General Background 119
b. Product 119
c. Marketing 120
d. Price 121

Table of Contents (Continued)

4. Sage Software Inc. 121
   a. Company Background 121
   b. APS Product 121
   c. Markets 122
   d. Price 122
5. McDonnell Douglas 122
   a. Company Background 122
   b. Products 123
6. Arthur Andersen 123
   a. Company Background 123
   b. Products 124
   c. Markets 125
   d. Price 125

C. Some U.K.-Based Players 126
1. BIS (Nynex) 127
   a. Company Background 127
   b. BIS/IPSE Product 128
   c. Users 129
2. GEC Software 130
   a. Company Background 130
   b. Products 130
   c. Marketing 131
   d. Pricing 131
3. Imperial Software Technology 132
   a. Company Background 132
   b. ISTAR - IPSE 132
   c. Markets 133
   d. Pricing 134
   e. Outlook 134
4. Learmouth and Burchett Management Systems (LBMS) 134
   a. Company Background 134
   b. Products 135
   c. Marketing 136
5. Peat Marwick McLintock 136
   a. Company Background 136
   b. Product 136

D. Some West German-Based Players 138
1. Softlab GmbH 138

Table of Contents (Continued)

a. Company Background 138
b. Maestro Product 139
c. Markets 139
d. Product Positioning 139
e. Features 140
2. GPP GmbH 140
a. General Background 140
b. Product 141
c. Markets 141
d. Price 142
3. GEI Gmbh 142
a. Company Background 142
b. Product 142
c. Price 143
d. Market Strategy 144
E. Some French-Based Players 144
1. CGI Informatique 145
a. General Background 145
b. Product 145
c. Users' Perceptions of Product 146
d. Marketing 146
2. Cap Gemini Sogeti 146
a. General Background 146
b. Product 147
c. Marketing 147
F. Some Western European Research-Based Projects 147

VI Conclusions and Recommendations 151

A. Recommendations to Vendors of CASE Products and Services 151
B. Recommendations for Users of CASE Products and Services 153

Table of Contents (Continued)

A	Appendix: Software Engineering Terminology	159
B	Appendix: Analysis of Research Sample	165
C	Appendix: User Questionnaire	169
D	Appendix: List of Related INPUT Reports	177
E	Appendix: Key Factors in Software Quality	179

Exhibits

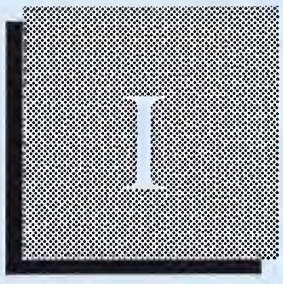
II	-1	Computer Aided Software Engineering	8
	-2	The Industry Crisis	9
	-3	The Catalyst to Change	11
	-4	A Window of Opportunity	12
	-5	Evolution Not Revolution	14
	-6	Strategic Directions for Vendors	15
	-7	INPUT's Solution for Users	16
III	-1	Applications Development Tools (ADTs) - Market Structure and Evolution	21
	-2	Software Development Approaches–Technology Life Cycle	22
	-3	Complete CASE Systems–Schematic	23
	-4	Development Alternatives –Tooling Alternatives	24
	-5	CASE Software and Services Forecast Definition	27
	-6	Comparative Economic Statistics, 1986	29
	-7	Comparative Economic Statistics, 1987	31
	-8	Market Pressures Influencing Size and Growth	33
	-9	Western European Systems Integration Market Forecast, 1987-1992	35
	-10	Commercial Systems Integration Market Forecast, 1987-1992	36
	-11	Government Systems Integration Market Forecast, 1987-1992	37
	-12	CASE - New Tools, Old Approaches	41
	-13	Western European CASE Software and Services Market Forecast, 1987-1992 (by Country Market)	43
	-14	Western European CASE Software and Services Market Forecast, 1987-1992 (by Delivery Mode)	44
	-15	CASE Products and Services - Hot Verticals	49
	-16	CASE Training - Key Trends and Issues	51
	-17	CASE Consultancy - Key Trends and Issues	54
	-18	Software Development Methodology	55
	-19	Basic Operation of a Project Control System	56
	-20	CASE Software Products - Trends, Issues and Opportunities	57

Exhibits (Continued)

	-21	Activities Encompassed by an IPSE	60
IV	-1	User Environment - Macro Issues	66
	-2	Changes in the Applications Backlog, 1986-1987 (by Industry Sector)	67
	-3	Changes in the Application Backlog, 1986-1987 (by Country Market and Established Size)	68
	-4	Users' Perceptions of Major Constraints on Their Ability to Reduce the Applications Backlog	69
	-5	Users' Perceptions of Major Factors Enabling Reduction of Applications Backlogs	71
	-6	Structured Methodologies - Market Penetration and Opportunities (by Industry Sector)	74
	-7	Structured Methodologies - Market Penetration and Opportunities (by Country Market)	75
	-8	Users' Reasons for Not Using Proprietary Methodologies	77
	-9	Methodologies - Four Evolutionary Threads	79
	-10	Users' Reasons for Not Using CASE Tools and Techniques	82
	-11	Users' Perceptions of Problems Associated with the CASE Approach	83
	-12	Users' Perceptions of Benefits Associated with a CASE Approach	85
	-13	Users' Estimates of Time Saved on Projects by Using CASE Tools and Techniques	88
	-14	The Software Development Life Cycle	89
	-15	Traditional Approaches towards Software Development - The Plan and the Reality	90
	-16	The Impact of CASE - A Learning Curve	91
	-17	Fourth Generation Languages - Market Penetration and Opportunities - Comparison between 1986 and 1987 (by Country Market)	93
	-18	Fourth Generation Languages - Market Penetration and Opportunities (by Industry Sector)	94
	-19	Fourth Generation Languages - Market Penetration and Opportunities (by Country Market)	95
	-20	Trends in Non-CASE Applications Development Tools	96

Exhibits (Continued)

	-21	Users' Estimates of Time Saved on Projects by Using Fourth Generation Languages	97
	-22	Users' Plans for Installing Fourth Generation Language Products	98
	-23	Market Penetration of 4GL Products	100
	-24	Users' Ratings of Importance of Potential Approaches towards Improving Data Processing Development Productivity	101
	-25	Users' Ratings of Importance of Potential Approaches towards Improving Data Processing Development Productivity (by Country Market)	102
	-26	Optimum Level of Control	104
	-27	Users' Ratings of Potential Approaches Supporting End-User Development Productivity	105
	-28	Users' Ratings of Potential Approaches towards Supporting End-User Development Productivity (by Country Market)	106
	-29	User View of Connectivity	107
V	-1	CASE Product Evolution	110
	-2	Software Development Model - The Need for Integrated Support	111
	-3	U.S.-Based Vendors - CASE Products	116
	-4A	U.K.-Based Vendors - CASE Products	126
	-4B	BIS–Turnover Analysis by Industry Sector	128
	-4C	BIS–Turnover Analysis by Geographic Region	129
	-5	West German-Based Vendors - CASE Products	138
	-6	French-Based Vendors - CASE Products	144
	-7	Some Research-Based CASE Products/Projects	148
VI	-1	Recommendations for Vendors of CASE Products and Services	154
	-2	INPUT's Solution for Users - Productivity Improvement Plan	155
	-3	Recommendations for Users of CASE Products and Services	157
B	-1	Corporate User Interviews - Sampling Frame	166
	-2	Analysis of Vendor Interviews	167



Introduction

Introduction

This report has been written as part of INPUT's comprehensive Information Services Program - Europe (ISPE). Other special interest topics covered during 1987 include Value-Added Network and Data Services, Software Pricing and Support, Professional Services (focusing on Systems Integration), and the Annual European Industry Report which will focus on the quantitative aspects of market sizing forecasting and industry structure.

A Scope

Computer Aided Software Engineering (CASE) has been heralded as the answer to the maiden's prayers in the software industry to the problems of developing quality software on-time and within budget.

The term CASE is used very loosely within the industry in connection with a wide variety of application development tools, techniques, and approaches.

CASE and hence the scope of this report is best defined by stating what it is not:

- It is not merely fourth generation languages and code generators.
- It is not merely incremental improvements in retrieving information from databases.
- It is not merely enhanced diagramming tools for systems analysts.

Software engineering is essentially a change in peoples' methods of working. The essence of CASE is the development and use of systematic strategies for the production of software that:

- Meets application requirements.
- Exploits existing methods and supportive tools.
- Is available on time.
- Is available within budget.
- Allows for easy maintenance and support.

CASE is more than just the application of sophisticated tools. It implies an evolution from traditional manual methods of software production and inflexible, unprofessional approaches towards managing and organising software engineers.

To define CASE in terms of application development tools (ADTs): it is a total development environment of RDBMS, a data dictionary or repository, fourth generation languages (4GLs), two-dimensional graphical programming languages and an integrated workbench covering all phases of the development lifecycle.

CASE products allow an engineer to build a graphical image of the desired application on a workstation and then automatically generate the code to produce it.

Under CASE the process of building applications becomes a pure specification exercise, in which engineers never write code but simply use tools to bring together dictionary code.

Whereas the current generation of MIS managers have adopted incremental solutions to the problem of productivity via integrated RDBMS/ 4GL approaches, INPUT foresees a migration towards the CASE approach which is already beginning to build momentum.

The major factors in the user environment that are driving this trend are:

- The increasing gap between applications demand and skills supply.
- The increasing size and complexity of development projects.

- The increasing need of information systems and IS developers to respond to changes in a complex and competitive business environment.

B

Objectives

INPUT's particular objectives in conducting this research programme and preparing this report have been to:

- Establish an overall view of the software development environment focussing on CASE approaches and the market for CASE products and services.
- Analyze perceptions of the key trends and issues that are impacting on the CASE market.
- Analyze risks and opportunities associated with market development both from the perspective of specialists CASE product/service vendors and from the perspectives of professional services companies (i.e., independant training companies, software/systems houses and consulting companies) as well as the hardware manufacturers.
- Isolate trends in the structure and dynamics of the competitve environment including a review of some key international players in the CASE market and commercially available CASE products.

Although this report focuses on the Western European market (i.e., all E.E.C. affiliated countries plus Norway, Finland, and Sweden) consideration has been given to CASE developments in the U.S.A. when reviewing the competitive environment and analyzing market opportunities.

Although the analysis covers the market for applications development tools (ADTs) in general, i.e., DBMS, program development tools (PDTs), languages and advanced products (CASE and Expert Systems), INPUT has focused the report on areas of new opportunity and high potential growth. Consequently, CASE has been highlighted for special attention and focused analysis.

In addition, the report does not include detailed analysis of leading edge developments such as artificial intelligence (AI) interfaces, parallel programming, and object-orientated programming. These will be dealt with in a report planned for INPUT's 1988 SSPE programme entitled "Trends and Opportunities in Fifth Generation Software Engineering".

Organisations with a particular interest in well-established but evolving markets in database management systems are recommended to see INPUT's U.S. report entitled "Future DBMS Markets, 1987-1992".

In addition, as systems integration is a major driving force behind the uptake of advanced CASE products, readers are recommended to see INPUT's report entitled "European Market Trends and Opportunities in Professional Services, 1987-1992" which focuses on SI developments.

A number of INPUT's international reports on subjects related to this area are listed in Appendix D.

Enquiries and comments are invited by INPUT regarding this report and any related topics of interest.

INPUT expresses its gratitude to all those individuals and companies that participated in the research programme upon which this report is based.

C Methodology

Field research for this report was obtained form an interview programme that was conducted during the period of June through September 1987 which consisted of:

- Corporate user interviews.

Structured interviews were conducted with 210 senior data processing personnel in a wide cross-section of organisations.

The interviews were conducted as part of INPUT's annual extensive user research survey and addressed the following issues:

- Trends and issues impacting the applications backlog.
- Usage and attitude towards Integrated Project Support Environments (IPSEs), structured methodologies, CASE tools, and other application development tools, e.g., Fourth Generation Languages (4GL).
- Estimates of time saved on projects using productivity tools and techniques.
- Perception of the importance and attitude towards various potential approaches towards improving development productivity and supporting end-user development productivity.

- Plans and budgets for applications development tools.

- Vendor interviews.

  - In-depth interviews (mostly face-to-face discussions) were con ducted with thirty-five senior personnel amongst hardware manufacturers, systems integrators, software houses, consultancy firms, independent training companies, and specialist software products vendors.

  - An analysis of the overall research sample is included as Appendix B.

  - The questionnaire used to conduct the corporate user interviews is given as Appendix C.

- Other Studies

  - INPUT's continuous research programme on the information services markets has been used where appropriate to further understanding of the issues and markets discussed.

- Other Public Domain Sources

  - Company press releases, press articles and reports have been used where appropriate to obtain background data on market develop ments.

- Country Market Comparison

  - For convenience of comparison between markets, local currencies have been converted to U.S. dollars on the basis of average exchange rates for 1986 which are illustrated in Exhibit III-6.

  - Owing to the volatility of international exchange rates anticipated for the forecast period 1987 to 1992, INPUT has not attempted to forecast future exchange rates. Currency conversion for the period 1987-1992 has been made at average 1986 rates.

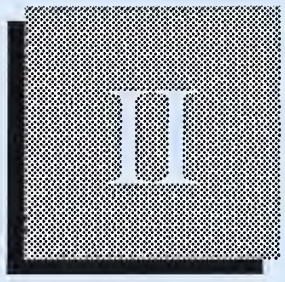
  - INPUT suggests that vendors exploiting export opportunities should carefully hedge their risk exposure to international currency movements.

D

Report Structure

The remaining chapters of this report are organised in the following way:

- Chapter II is an Executive Overview providing a summary of the contents of the entire report.
- Chapter III includes an overview and definition of CASE and its relationship with other segments of the market for applications development tools. Issues and trends impacting market development are included together with INPUT's assessments of market size and expected growth.
- Chapter IV provides a comprehensive review of user attitudes, needs and approaches towards software development.
- Chapter V provides an analysis of the competitive environment for CASE products.
- Chapter VI provides INPUT's conclusions and recommendations for both users and vendors of CASE products and services.
- The appendices contain a list of definitions of software engineering terms, an analysis of the research sample, the user survey questionnaire, a list of related INPUT reports, and a list of key factors in software quality.



Executive Overview

Executive Overview

A

Computer Aided Software Engineering

This is the essence of software engineering and hence CASE is a change in peoples' methods of working.

The CASE approach implies the development and effective use of appropriate management methods, personnel organisation, automation tools, and systematic techniques that permit teams of software developers to produce software that meets business requirements, allows for easy maintenance and support, and is available on time and within budget.

CASE is probably best defined by what it is not. It is not merely fourth generation languages and code generators. It is not merely enhanced diagramming tools for systems analysts. It is not merely project management tools.

To define CASE in terms of software tools, it means an effectively implemented complete CASE environment which automates all of the activities (including management) of the software development lifecycle.

CASE environments allow teams of engineers to build graphical images of the desired application on networked workstations and then use customizable tools to bring together code from a central dictionary to automatically generate the application programme.

An effectively implemented CASE environment is the answer to the problems in the software industry. With the CASE approach software development becomes a specification exercise only.

EXHIBIT II-1

COMPUTER AIDED SOFTWARE ENGINEERING (CASE)

The essence of CASE is the development and effective use of appropriate management approaches, automation tools, and systematic techniques that permit teams of software engineers to produce software that:

- Meets Business Requirements
- Is Available On Time
- Is Available within Budget
- Allows for Easy Maintenance and Support

B

The Industry Crisis

The current momentum behind CASE and the effective implementation of CASE techniques and technology has arisen out of a building crisis in the computer industry.

Software development approaches have simply not kept pace with growing demand for new and increasingly complex systems.

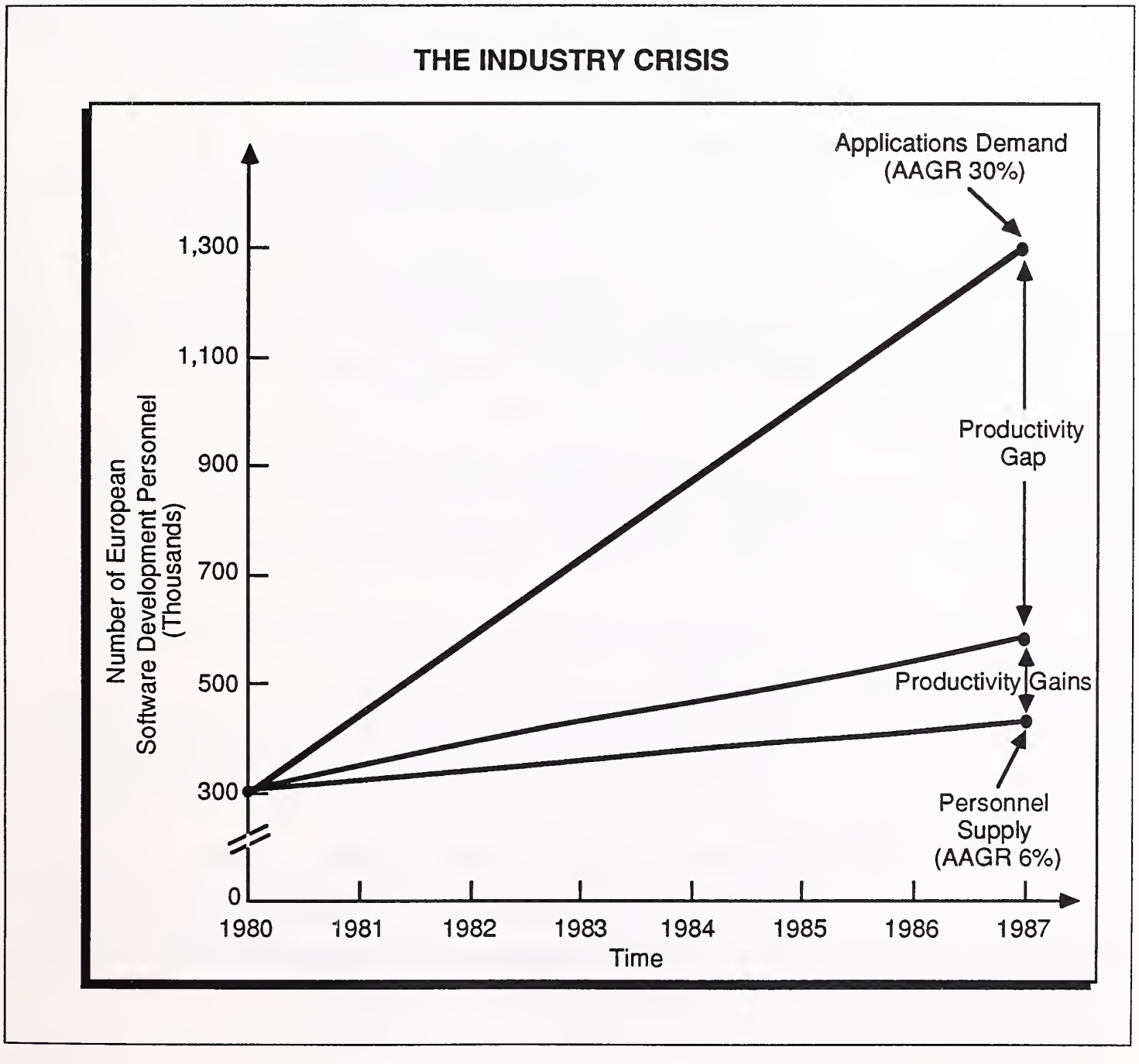
Exhibit II-2 illustrates the growing productivity gap. Productivity in software development has only increased in the last few years in the range of 4-8% versus 8% in other areas of industry. Hardware engineering productivity has improved by 30-35% during the same period. This means major crisis, especially in light of the fact that 80% of all systems functions are now implemented in software, compared to less that 20% in the mid 1950s.

Research also indicates that of the $100 billion plus spent annually worldwide on software development, as much as 60% is devoted to

maintaining existing systems. A further 20% is devoted to software projects that are abandoned without implementation, often after delivery.

Encouraging and supporting end users in developing their own applications is not an 'Aspro' solution to this problem. The industry has recognized that the solution lies in a disciplined engineering approach towards software development and a professional approach towards managing the software development process.

EXHIBIT II-2



C

The Catalyst to Change

INPUT's user research revealed two strategic concerns for data processing that are impacting the evolution towards CASE approaches to software development.

Firstly, users face the challenge of systems integration, i.e., building complex online systems which seek to link up disparate decentralized systems. Coupled with the growth of end-user computing, this is causing increasing backlogs in 45% of our research sample organisations.

Secondly, users are seeking productivity gains from all corporate assets, including capital investments, human resources, and information assets in order to meet the challenges of complexity and competitiveness in the business environment.

An effectively implemented CASE approach offers a solution to these problems at a macro level that is impacting data processing developers. The benefits from a CASE approach are illustrated in Exhibit II-3.

An effectively implemented CASE approach means investment in CASE tools, training, and effective broadbased management of information systems effectiveness, efficiency, and systems development productivity.

D

A Window of Opportunity

The momentum behind the effective implementation of CASE has been slowed by several factors:

- Cultural resistance to changing working practices.
- Lack of management committment to training and structured methods.
- Difficulties in justifying investment owing to the lack of standards for productivity measurement and the initial unsuitability of a CASE approach to relatively immature data processing environments.

As illustrated in Exhibit II-4, some innovative users have been disappointed with the inability of early CASE products to provide full lifecycle support for a range of methods across multiple hardware environments. In addition, there have been disappointments with the costs of implementation.

EXHIBIT II-3

THE CATALYST TO CHANGE -
A Clear Case for CASE

- Improved Morale and Job Satisfaction
- Improved Software Quality
- Simplified Programme Maintenance
- Reduced Development Lead Times
- Practically Supports a Structured Software Engineering Approach
- Allows System Development to Respond to Business Development
- Frees Developer to Focus on Creative Aspects of Software Development
- Improved Team Efficiency

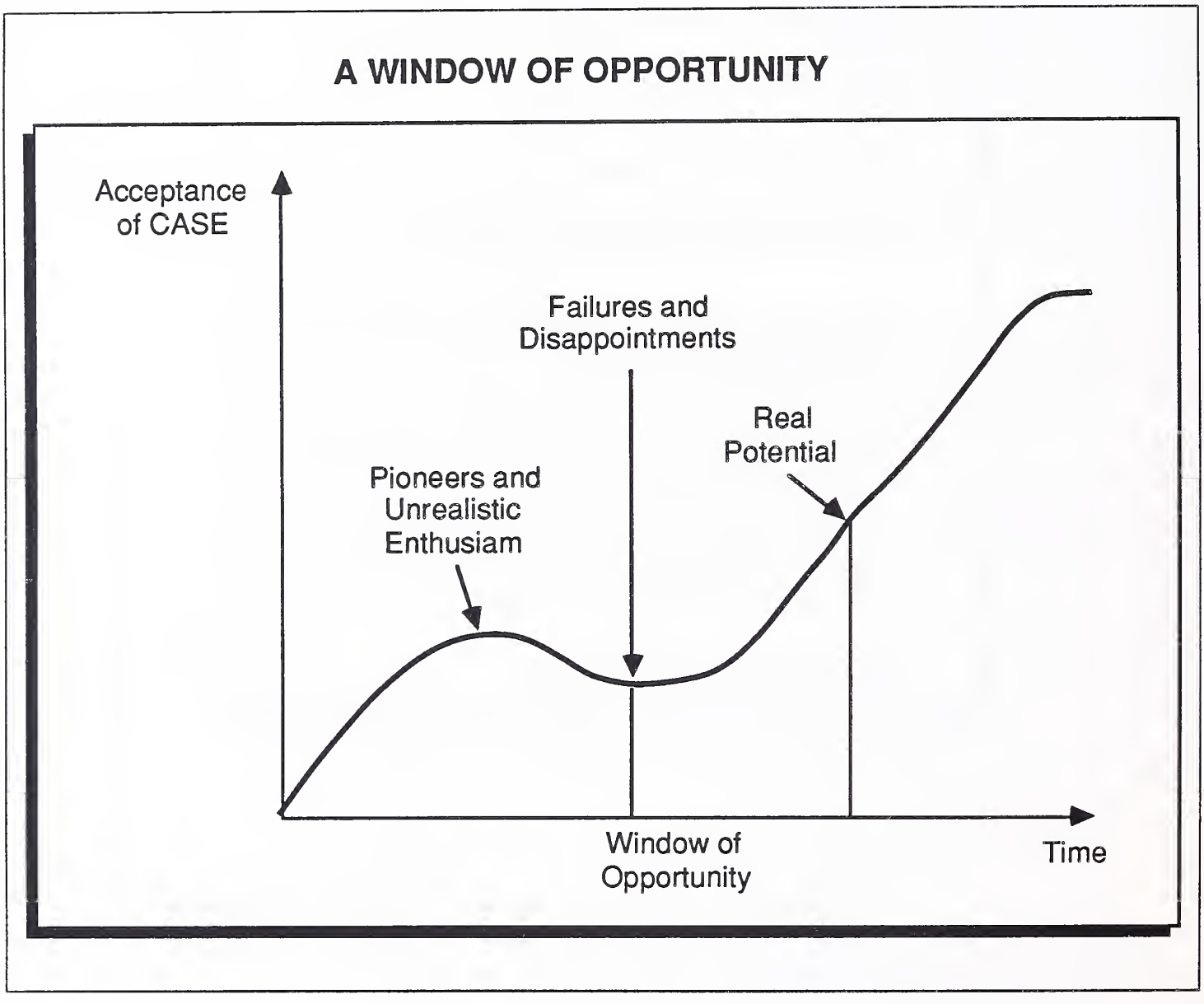
Nevertheless, INPUT sees a window of opportunity for vendors of advanced CASE products in the late 1980s. The market will evolve as CASE becomes the standard industry method of building and maintaining software.

The use of advanced CASE systems is growing rapidly from its roots in the defence, aerospace, and electronics sectors into the commercial sector.

Innovative users in the civil government, financial services, and manufacturing sectors have experienced significant gains in productivity after early losses on initial projects. The major benefits perceived, however,

have been in quality improvement.
As the market matures via education and awareness of CASE solutions, and CASE products and services become more sophisticated, INPUT

EXHIBIT II-4



E

Evolution Not Revolution

sees strong growth and profit potential for the future.
INPUT estimates that the Western European market for CASE software and services will grow from a base of $140 million in 1987 to $750 million by 1992 at an AAGR of 40%. See Exhibit II-5.

Despite the clear commercial benefits of the CASE approach, INPUT

sees change as evolutionary rather than revolutionary in this market. Our research revealed that many organisations are not yet ready in terms of data processing organisation methods and culture to adopt a CASE approach.

In recognition of the fact that the major issue in CASE inplementation is the people, and associated management process, rather than the technology, INPUT forecasts high growth especially for CASE-related training.

In addition, there are significant opportunities in CASE consultancy to ease the process of product and methodology evaluation, as well as the implementation of appropriate management techniques and methods for CASE.

CASE software products are evolving towards complete CASE environments as potentially inplementing CASE users realise that an integrated CASE approach is the only sensibly bona fide strategy for developing larger systems.

F

Strategic Directions for Vendors

In order to avoid the shake-out in the CASE tool market, vendors need to achieve critical market position in terms of a comprehensive product and implementation service offering. Strategic alliances and distribution agreements are a key element in this strategy.

Channel management strategy needs careful review in light of needs for comprehensive implementation support and vertical market specialisation to address the key CASE marketing and implementation issue of knowledge of a clients' business environment and culture.

As the CASE market is currently unstable and immature, vendors need to address the issue of market education via missionary marketing initiatives such as discussion workshops, seminars, and quality circles.

Product development needs to incorporate the concepts of integration, flexibility, and openness.

CASE implementation will become more effective if users can integrate existing programmes into new systems which are developed using CASE tools.

In addition, CASE tools need to be sufficiently authoritative to enforce structured methods but sufficiently flexible to allow for the evolution in techniques and open in their ability to incorporate new tools and methods

EXHIBIT II-5

EVOLUTION NOT REVOLUTION

WESTERN EUROPEAN CASE SOFTWARE AND SERVICES MARKET FORECAST, 1987-1992
(User Expenditure $ Millions)

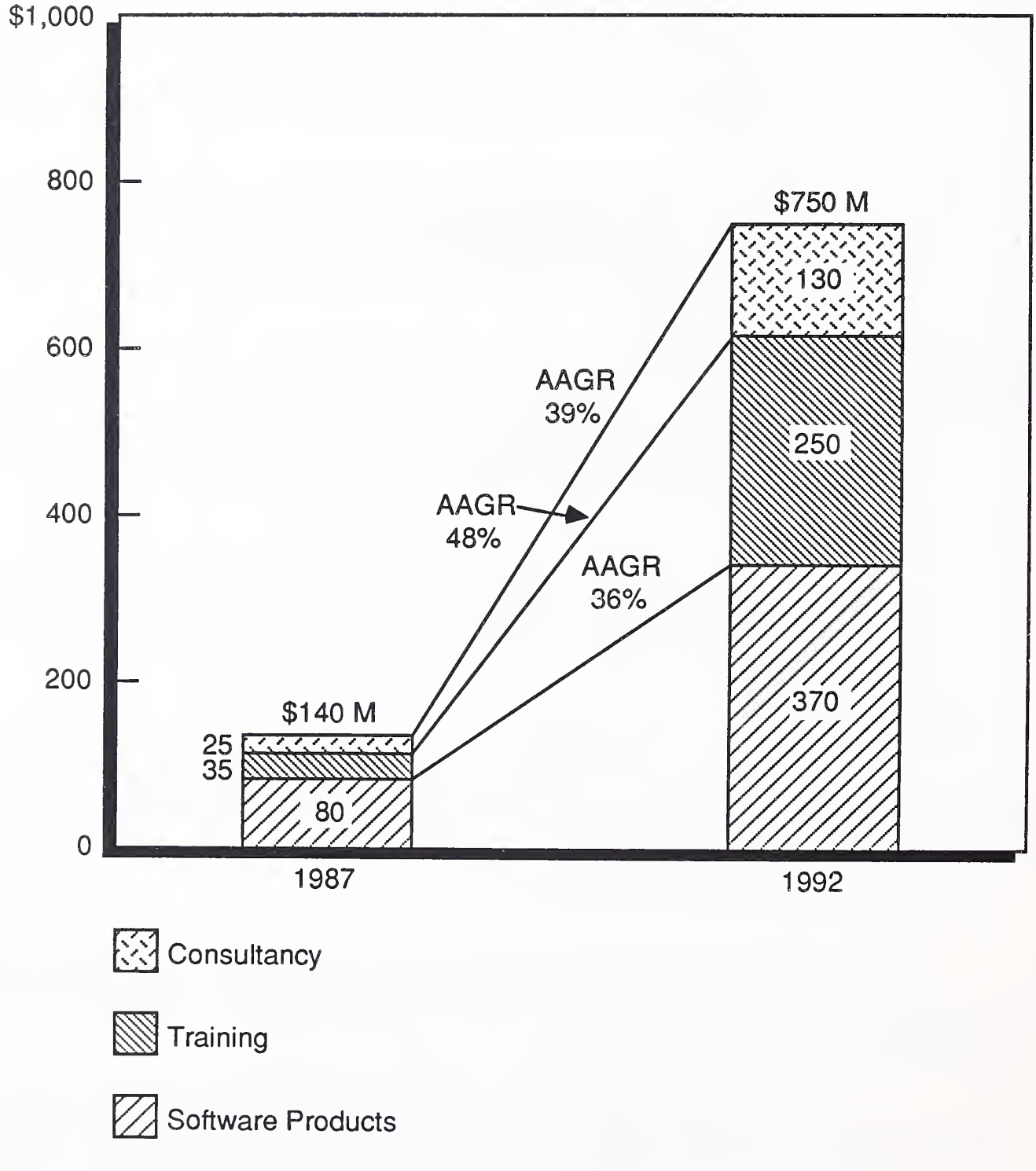


EXHIBIT II-6

STRATEGIC DIRECTIONS FOR VENDORS

- Achieve Critical Market Position in Terms of Product/Service/Distribution Offering
- Provide Comprehensive, and Preferably Independent, Implementation Support Service
- Educate the Market
- Carefully Review Channel Management Strategy
- Commit to the Principles of Openness, Flexibility, and Integration in Product Development

G

INPUT's Solution for Users

INPUT's key user recommendation is to adopt CASE with caution. That is, it needs to be implemented in the context of a mature professional management environment and productivity improvement plan.

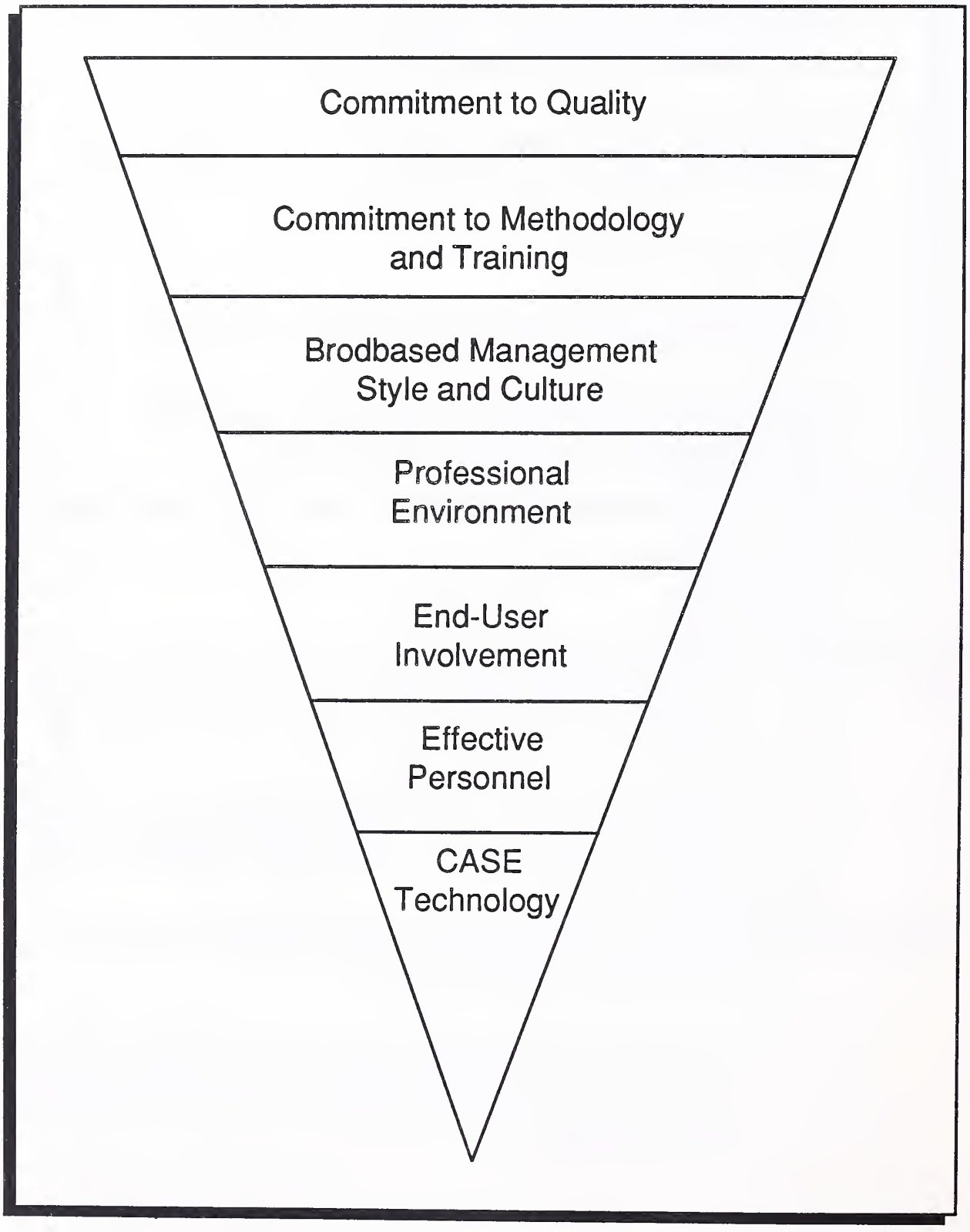
The rationale behind the plan is the recognition that the major factors impacting on software quality and productivity are firstly, management style and methods, secondly, methodologies employed, and thirdly, technology used.

Consequently, it is important to focus on the human resource management issues in systems development.

A professional environment is one where there is broadbased commitment to quality, methodology, training, and well-understood objectives. Control must not be overly rigid and everyone must be treated and behave like a professional.

EXHIBIT II-7

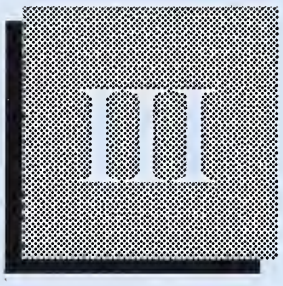
INPUT'S SOLUTION FOR USERS PRODUCTIVITY IMPROVEMENT PLAN



Careful research needs to be undertaken on the usage, attitudes towards, and requirements for CASE tools. This will then ease the process of standardization in terms of methods and tool support.

Users should budget for initial experimentation with CASE technology and it should be evaluated by a group isolated from mainstream development activity.

In addition, users should budget not only for specific training on advanced techniques but also for general on-going management and personnel skills training. The software engineer of the future is a creative business analyst, problem solver, and project manager.



Market Analysis and Forecast

Market Analysis and Forecast

A

Market Overview, Structure, and Evolution

1. Introduction - The Software Development Crisis

The current momentum behind CASE and the implementaiton of CASE technology has arisen out of a building crisis in the computer industry.

Software technology has simply not kept pace with the demand for new and increasingly complex systems.

Research indicates that 80% of all systems functions are now implemented in software, compared to less than 20% in the mid-1950s.

Research also indicates that of the $100 billion plus spent annually worldwide on software development, as much as 60% is devoted to maintaining existing systems. In addition, research also indicates that over half of the remaining expenditure is devoted to software projects that are abandoned without implementation, often after delivery.

The crisis in the computer industry is that advances in software engineering have lagged well behind developments in hardware engineering. Software development especially in the commercial area is slow and unreliable, largely due to the inefficient and unreliable methods, techniques, and management approaches adopted towards the development process.

The objective of CASE in its broadest definition is to solve this crisis by automating everything a human does to software, i.e., turning software development from an unstructured art into an engineering discipline.

To an extent the software industry is still labouring under a cottage industry mentality. Although hardware engineers have long been able to take advantage of CAE/CAD techniques and technologies which have aided quality and productivity in the hardware development lifecycle, software engineering techniques and supportive tools have until recently largely been igonored by companies' managements.

Software engineering is not a new concept in the industry and has developed from its birth at the Software Engineering Conference in Garmisch, West Germany in 1968.

Until recently, software engineering and CASE which is the automation of the engineering approach has been regarded by many organisations as "academic" and inappropriate to the process and management of software development.

However, CASE has now been endorsed by major suppliers and computer users alike and INPUT foresees that the implementation of CASE approaches and technology will have real commercial impact in the late 1980s onwards to provide a solution to the industry's crisis.

2. Market Structure and Development

With reference to Exhibit III-1 it is possible to discern a large variety of approaches and tools which have been developed that seek to address the problems of productivity and quality in applications development that can be labeled under the umbrella of applications development tools (ADTs).

Market development evolves as user needs become more complex and the development environment more challenging.

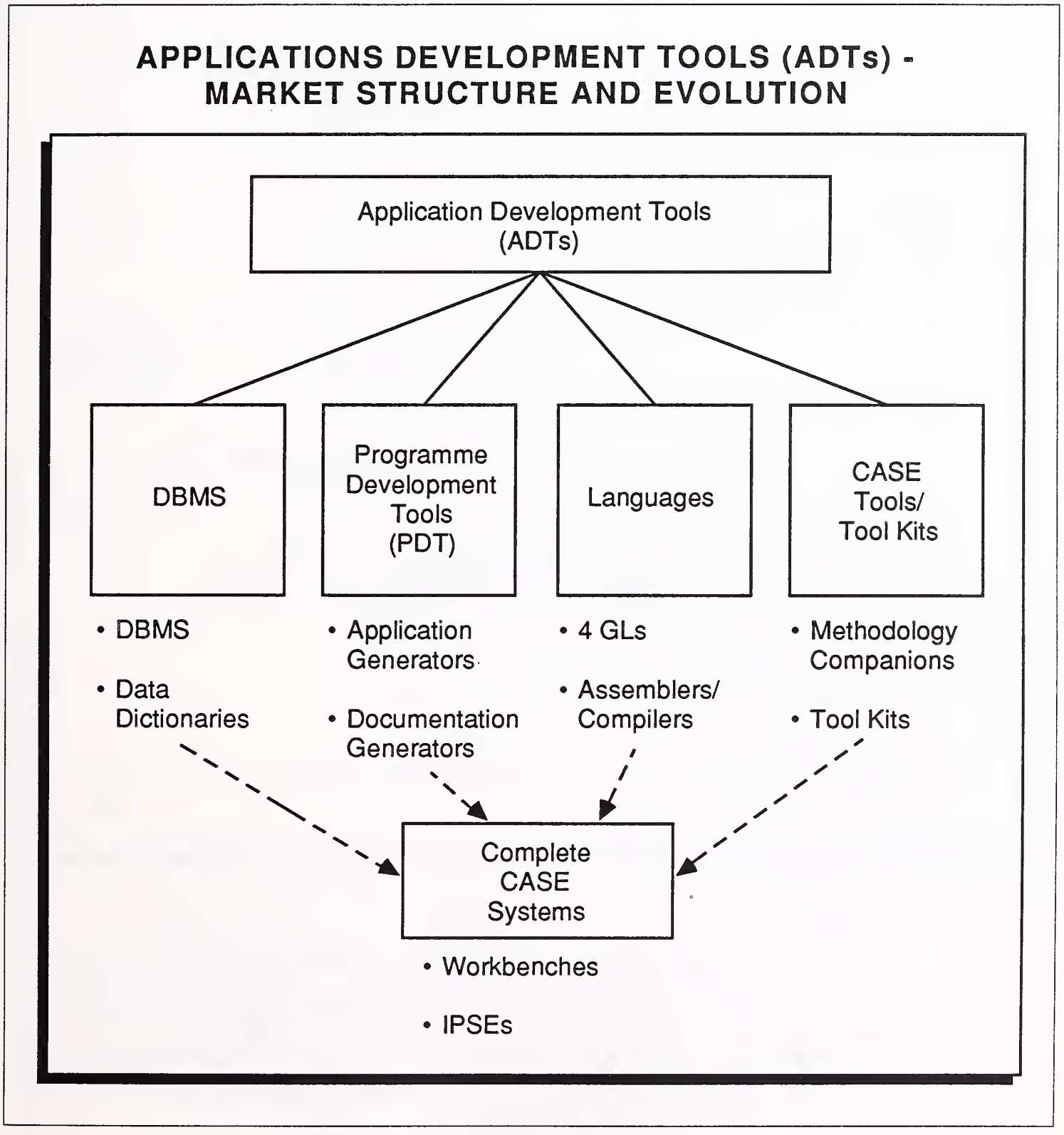
Exhibit III-2 illustrates the relative position of various approaches and tools used in software development in terms of a software development tool technology lifecycle.

RDBMS and 4GLs which had their roots in the 1960s are now widespread in their use within the corporate data processing environment.

CASE tools and tool kits are a relatively new innovation based on the old concepts of structured programming and structured analysis and design philosophies in the 1970s.

The most common CASE tools in use are those that automate the use of structured methods for analysis and design. In addition, automated

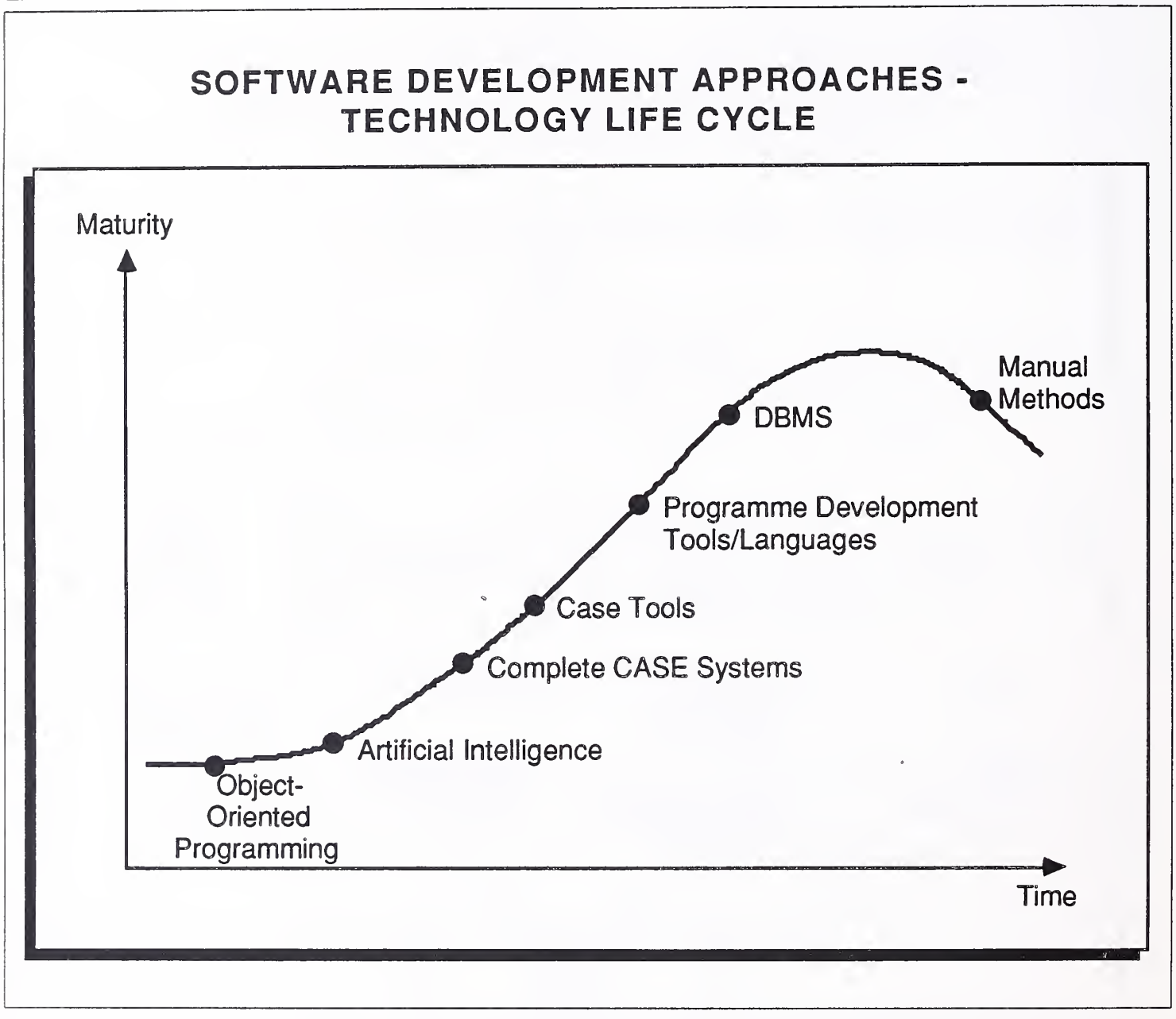
EXHIBIT III-1



project management tools are in relatively common use.

The major trend in the market for application development tools (ADTs), as illustrated in Exhibit III-1, is the convergence towards the use of COMPLETE CASE SYSTEMS.

EXHIBIT III-2



A COMPLETE CASE SYSTEM is essentially an integrated software technology that provides an automated, engineering discipline for software development, maintenance and project management. They include structured methodologies and integrated sets of tools that have a common user interface and automate all the phases of the software lifecycle in a common computer environment.

Essentially, these new COMPLETE CASE SYSTEMS fall into two broad categories: IPSEs (Integrated Project Support Environments) and full-cycle CASE WORKBENCHES.

Evolutionary trends in the development of CASE tools are reviewed more thoroughly in Chapter V of this report.

However, the general trend is towards integration and connectivity between ADTs and the convergence towards complete CASE systems that offer full life cycle support and flexibility for future development.

In terms of ADTs, complete CASE systems provide a total development environment of RDBMS, a central data dictionary or repository, 4GLs and two-dimensional graphical programming languages. Complete CASE systems allow the software engineers to build a graphical image of the derived application on his/her workstation and then automatically generate the code to produce it.

The characteristics of complete CASE systems are illustrated in a schematic given as Exhibit III-3.

EXHIBIT III-3

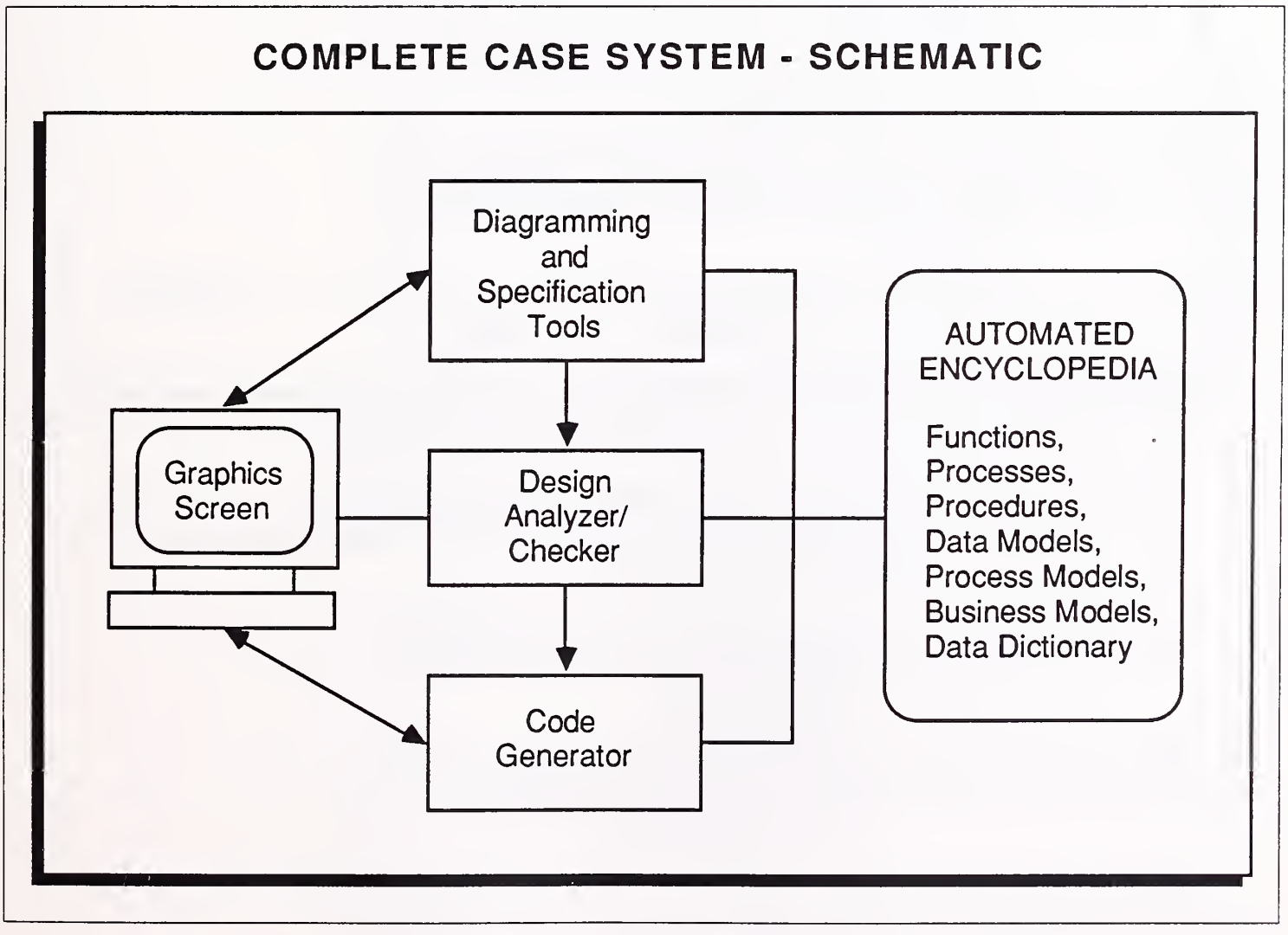
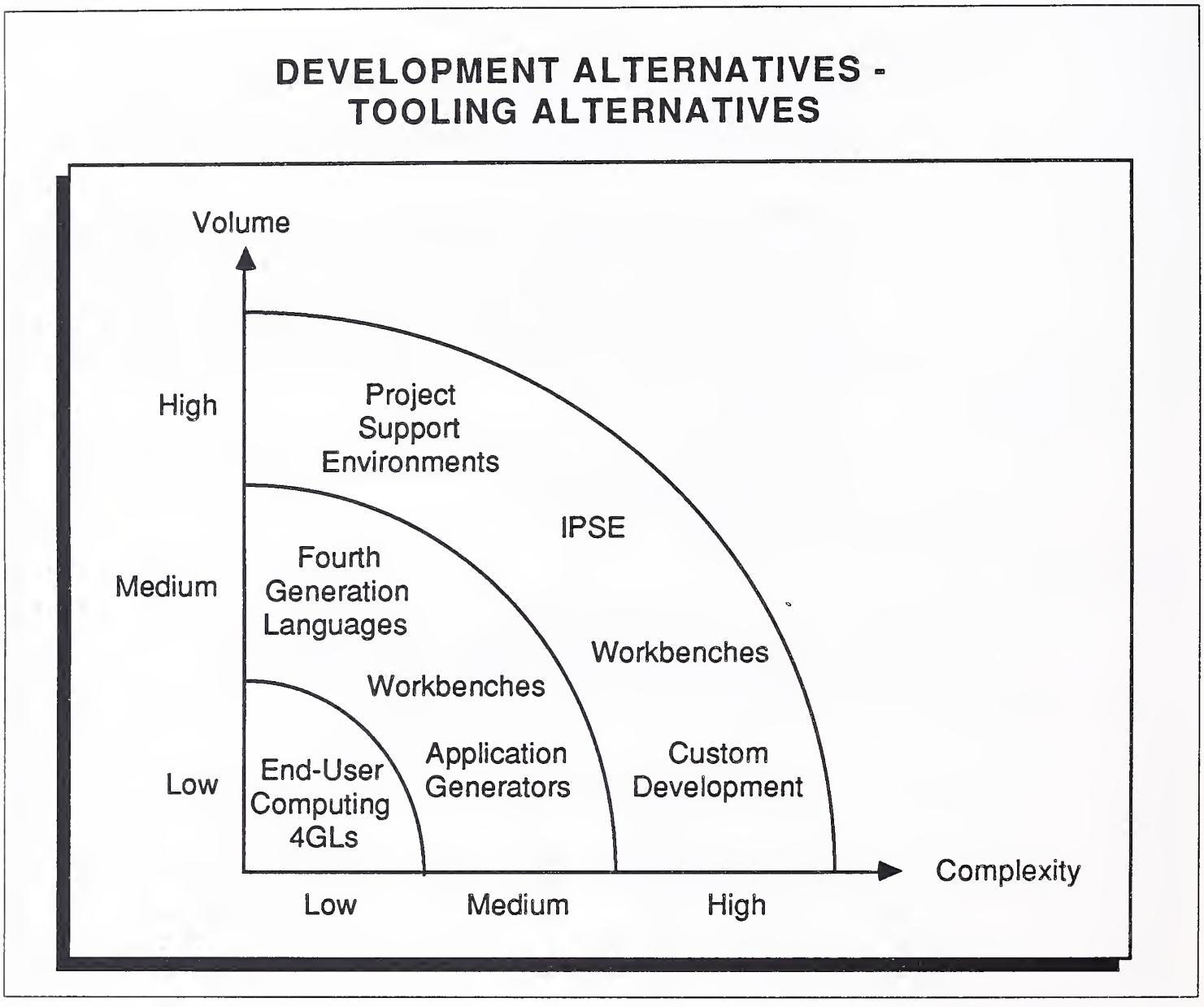


EXHIBIT III-4



The heart of the complete CASE system is an automated repository which holds all the information needed to create, modify and evolve software systems. That is:

- Information on problem to be solved.
- Problem domain.
- Emerging solution.
- Software process being used.
- Project resources and history.
- Organizational context.

The central repository then provides the basis for:

- CASE tools integration, including foreign tools.
- System specification consistency and integrity.
- System information sharing.
- Document standardization.
- System documentation generation.
- Code generation.
- Software reusability.
- Project management and control.

In terms of the evolution and future developemnt of ADTs, it is important to remember that the choice of the components of a development environment should be determined by the requirements of the system to be developed.

With reference to Exhibit III-4 it is possible to observe that a workbench approach is highly suitable to meeting the growing demand for integrated on-line transactions processing systems for which there is a growing demand in the commercial sector.

IPSEs are also suitable for this environment with large highly complex databases and high volumes. They are particularly suitable for large complex real-time projects and also large projects with dispersed project teams.

The 4GL approach is suitable for end-user computing and in the development centre for small or medium-sized projects.

However, again with reference to Exhibit III-4, it should be noted that these boundaries between development alternatives and tooling alternatives are increasingly blurring as hardware price/performance continues to improve and complete CASE systems become established in use.

Consequently, INPUT foresees that complete CASE systems (workbenches/IPSEs) are a highly viable approach towards developing medium-sized projects and large numbers of smaller projects.

In addition, as they easily facilitate the development of custom software, they are a highly viable alternative to the purchase of packaged software for many "standard" applications.

In addition to the choice of tooling approach to meet development requirements, there is also the choice of hardware architecture to be considered.

The market for CASE tools and especially complete CASE systems in terms of hardware architecture is being encouraged by connectivity and standardisation on "COTS" (commerical off-the-shelf) hardware.

Early CASE tools and toolkits were implemented on stand-alone micros which did not support the requirements of team-working and the integrated engineering approach. In addition, early versions of project support environments were implemented on non-standard UNIX-based hardware which involved innovative users in incremental hardware costs in implementation.

The complete CASE systems of the future will be implemented on:

- Networked micros.
- Networked micros linked to a target mainframe environment.
- Micros or workstations connected directly to the target mainframe.
- Micros or workstations connected to a minicomputer.

For smaller projects CASE workbenches implemented on networked micros are a viable alternative. For larger projects the IPSE approach implemented on networked workstations linked to a central mini/mainframe environment is the way forward.

B

Market Forecast

1. Forecast Definition

The market assessment and forecast growth that follow were developed from assessments of current and projected activities within the market definition illustrated in Exhibit III-5.

Although there are many grey areas of overlap between the boundaries seperating various categories of application development tools, INPUT has focused its analysis on products and services which come under the umbrella of CASE.

Consequently the forecast covers basic CASE tools and toolkits and advanced CASE tools, i.e., complete CASE systems (IPSEs and workbenches).

It should be remembered that as other segments of the ADT market evolve towards complete CASE systems (for example, data dictionaries linked to toolkits and applications generators linked to analysts' workbenches) these products are then measured within INPUT's definition of the CASE market.

EXHIBIT III-5

CASE SOFTWARE AND SERVICES FORECAST DEFINITION

TYPE OF PRODUCT / MODE OF DELIVERY	SOFTWARE PRODUCTS	PROFESSIONAL SERVICES - Consultancy - Education and Training	STANDARD TURNKEY SYSTEMS
ADVANCED CASE TOOLS	INTEGRATED PROJECT SUPPORT ENVIRONMENTS (IPSE'S)		
	WORKBENCHES - ANALYST/PROGRAMMER INTEGRATED ENVIRONMENTS		
BASIC CASE TOOLS	INDIVIDUAL TOOLS OR WORKBENCHES SUPPORTING A LIMITED RANGE OF PHASES OF THE DEVELOPMENT LIFE CYCLE i.e.: • Project, Quality and Implementation Management • Design and Analysis Tools • Specification Tools • Prototyping Tools/Testing Tools • Configuration Management Tools • Maintenance (Retrofit) Tools • A.I. Tools		
OTHER APPLICATION DEVELOPMENT TOOLS (ADTS)	• DBMS • Application Generators • Languages (e.g. 4GLs) • Data Dictionaries - Screen Painters • Compilers - Assemblers • Retrieval Systems • Static and Dynamic Analyzers		

Boundary of INPUT's Market Definition for Specialist CASE tools, products and services.

In recognition of the fact that CASE implementation entails significant change in the working environment and working practices of system developers and managers, INPUT has included revenues from CASE training and consultancy within the forecast definition.

The forecast covers the period 1987 to 1992 (including 1986 actuals) and accesses end-user expenditures. Forecasts are made in local currency and converted into U.S. dollars for aggregation and comparative purposes.

Owing to the volatility of international exchange rates anticipated for the forecast period, the U.S. dollar conversion rates used have been taken as the average rate for 1986.

Exhibit III-6 sets out the average U.S. dollar exchange rates for 1986 and Exhibit III-7 average U.S. dollar exchange rates for 1987 for a range of Western European currencies.

In addition, the forecasts have been expressed in actual monetary terms.

For your interest, the latest inflation rates in Western European countries and inflationary trends over the last twelve months have been illustrated in Exhibit III-6 and Exhibit III-7.

The statistics reveal an underlying trend for inflation rates to be rising in European economies owing to demand pull factors which relate to high levels of money supply (M3) growth and levels of salary/wage settlements.

EXHIBIT III-6

COMPARATIVE ECONOMIC STATISTICS
1986

	France	U.K.	West Germany	Italy	Holland	Belgium	Sweden	Denmark
Gross Domestic Product ($ Billions)	589	519	719	418	143	91	105	67
Growth 1986 GDP (Percent)	+2.2	+2.5	+2.6	+2.8	+2.2	+2.5	+1.8	+2.4
Size of Information Services Business ($ Billions)	6.67	4.35	4.78	2.82	1.39	0.84	0.91	0.67
Percent of GDP Information Services	1.14	0.83	0.67	0.67	0.97	0.92	0.86	0.99
Percent of GDP Market Services	45	42	41	40	47	49	36	43
U.S. Dollar Exchange Rates - Average Calendar 1986	6.86	0.68	2.14	1,473.5	2.43	45.3	7.08	7.76
1986 Inflation Rate (Percent)	+2.5	+3.4	-0.2	+6.0	+0.3	+1.4	+4.0	+3.8

EXHIBIT III-6A

COMPARATIVE ECONOMIC STATISTICS 1986 (CONT.)

	Norway	Finland	Switzer-land	Spain	Por-tugal	Ireland	Austria	Total	USA
Gross Domestic Product ($ Billions)	60	71	108	197	24	21	75	3,207	3,970
Growth 1986 GDP (Percent)	+3.0	+1.8	+2.0	+2.8	+3.0	+2.5	+2.3	+2.5	+2.3
Size of Information Services Business ($ Billions)	0.59	0.51	1.00	0.60	0.06	0.14	0.37	25.71	54.60
Percent of GDP Information Services	0.98	0.71	0.93	0.30	0.25	0.67	0.50	0.80	1.38
Percent of GDP Market Services	40	37	45	43	41	48	43	42	43
U.S. Dollar Exchange Rates - Average Calendar 1986	7.40	4.85	1.81	141.0	171.0	0.71	15.1	-	-
1986 Inflation Rate (Percent)	+6.9	+2.6	+1.0	+8.9	+12.0	+4.0	+1.8	+3.3	+3.0

EXHIBIT III-7

COMPARATIVE ECONOMIC STATISTICS 1987

	France	U.K.	West Germany	Italy	Holland	Belgium	Sweden	Denmark
Gross Domestic Product ($ Billions)	602	532	738	430	146	93	107	69
Growth 1987 GDP (Percent)	+1.3	+3.0	+1.5	+2.3	+1.7	+2.3	+2.0	+1.8
Size of Information Services Business ($ Billions)	7.98	5.29	5.71	3.56	1.65	1.00	1.07	0.79
Percent of GDP Information Services	1.32	1.0	0.77	0.83	1.13	1.07	1.00	1.14
U.S. Dollar Exchange Rates - Average Calendar 1987	6.12	0.63	1.84	1,316.4	2.07	38.4	6.44	6.96
1987 Inflation Rate (Percent)	+3.5	+4.4	+1.0	+6.2	+2.0	+1.7	+5.6	+4.5

EXHIBIT III-7A

COMPARATIVE ECONOMIC STATISTICS 1987 (CONT.)

	Norway	Finland	Switzer-land	Spain	Por-tugal	Ireland	Austria	Total	USA
Gross Domestic Product ($ Billions)	62	72	110	203	25	21	76	3,286	4,062
Growth 1986 GDP (Percent)	+1.5	+2.0	+2.0	+3.0	+3.6	+2.5	+2.4	+2.6	+2.5
Size of Information Services Business ($ Billions)	0.69	0.60	1.22	0.75	0.07	0.16	31.02	31.02	64.10
Percent of GDP Information Services	1.11	0.83	1.11	0.37	0.28	0.76	0.94	0.94	1.58
U.S. Dollar Exchange Rates - Average Calendar 1987	6.85	4.49	1.53	126.9	154.2	0.69	-	-	-
1987 Inflation Rate (Percent)	+7.3	+4.0	+1.5	+6.5	+10.5	+3.5	+3.9	+3.9	+3.8

2. Market Pressures Influence Size and Growth

The key overall market forces that are impacting the development of the CASE market are illustrated in Exhibit III-8.

The major driving force for CASE is essentially the crisis in the computer industry which has been documented in the introductory section to this chapter.

CASE offers a solution to the problems of excessive wastage of expensive resources by maintaining existing systems and developing systems that do not meet business requirements.

EXHIBIT III-8

MARKET PRESSURES INFLUENCING SIZE AND GROWTH

DRIVERS

- Increasing Gap between Applications Demand and Effective Skills Supply.
- Increasing Size and Complexity of Development Projects.
- Increasing Need of Information Systems and IS Developer to Respond to Changes in a Complex and Competitive Business Environment.
- Increasing Acceptance of Structured Methods.

INHIBITORS

- Cultural Resistance to Changing Working Practices.
- Lack of Management Committment to Training and Structured Methods.
- Risky Technology—Long Term Investment
- Lack of Skilled CASE Implementors.
- Lack of Standards for Productivity Management.

a. Systems Integration - A Major Driving Force

The other major driving force is the increasing size and complexity of development projects.

Advanced CASE products such as IPSEs are more than just important for the future of software development. For large systems they are essential to ensure quality and avoid cost and time overruns.

The increasing size and complexity of development projects is best exemplified by the growing demand in Western Europe for systems integration and the award of SI contracts to external service companies for their development.

Trends towards SI essentially are creating a derived demand market for suppliers of advanced CASE products. As one vendor suggested ". . . in the future the customer will increasingly ask the question . . . was the system developed using an IPSE. . .".

Systems integration (SI) can be broadly defined as ". . . the provision of a total solution to a multi-disciplinary information systems requirement . . .".

SI contracts have substantial networking and project management components. In addition, they have significant uniqueness in terms of applications supported, that is, a significant custom software component. Also, there is often substantial complexity in terms of hardware/system interfaces.

INPUT's latest forecasts on the Western European SI market are illustrated for your benefit in Exhibits III-9 to III-11.

It can be seen that there is substantial growth in the large SI contract segment (i.e., contracts greater than $5 million) especially in the commercial sector.

Readers should note that the forecasts of government SI include both civil and defence contracts which also include NATO and space.

In addition, all the SI forecasts include the following project components:

- Hardware
  - Computing and Communications (approx. 35% of total contract value)
- Software packages
- Custom software development
- Consulting
- Project management
- Education and training
- Operation and maintenance (includes facilities management)

EXHIBIT III-9

WESTERN EUROPEAN SYSTEMS INTEGRATION MARKET FORECAST, 1987-
European User Expenditure ($ Billions)

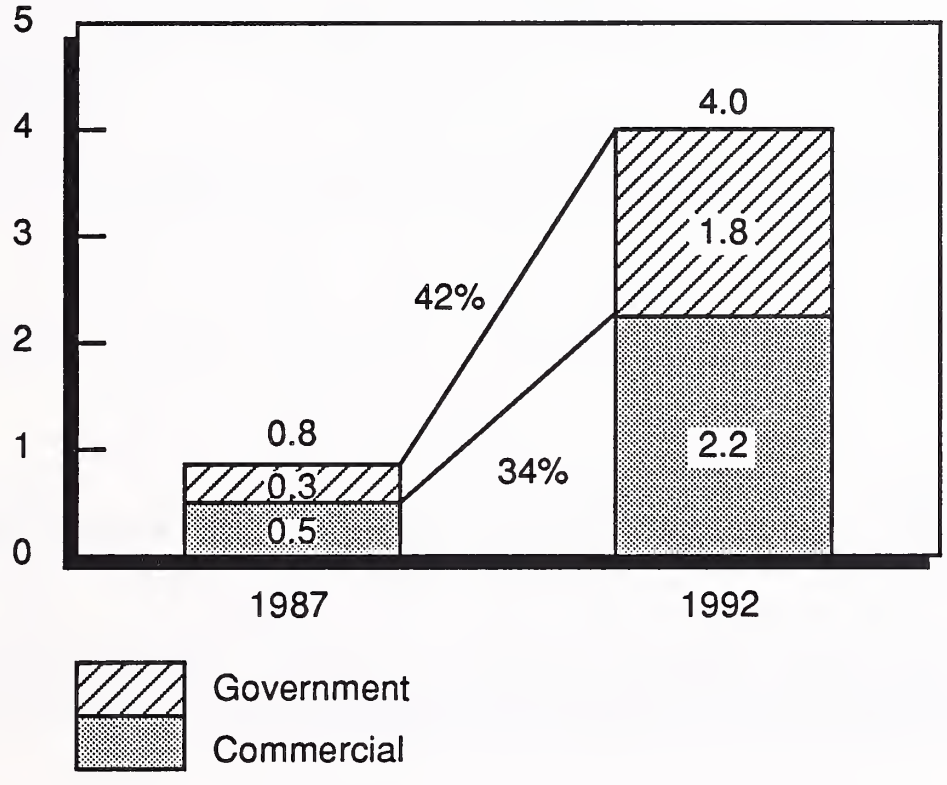


EXHIBIT III-10

COMMERCIAL SYSTEMS INTEGRATION MARKET FORECAST, 1987-1992 (Total Western Europe)

CONTRACT TYPE	MARKET FORECAST ($ Millions)					
	1986	1987	1986-1987 AAGR (Percent)	1989	1986-1987 AAGR (Percent)	1992
Contract Value Greater Than $5 Million	100	160	49	330	40	900
Contract Value Less Than $5 Million	230	345	38	600	29	1,300
Total Western Europe	330	505	41	930	33	2,200

EXHIBIT III-11

GOVERNMENT SYSTEMS INTEGRATION MARKET FORECAST, 1987-1992 (Total Western Europe)

CONTRACT TYPE	MARKET FORECAST ($ Millions)					
	1986	1987	1986-1987 AAGR (Percent)	1989	1986-1987 AAGR (Percent)	1992
Contract Value Greater Than $5 Million	175	265	47	555	43	1,160
Contract Value Less Than $5 Million	25	35	38	65	32	150
Total Western Europe	200	300	46	620	42	1,760

b. Examples of SI Contracts Awarded/To Be Awarded in 1987

The largest European SI contract which is expected to be awarded to IBM in 1987 is for the AMADEUS system. The contract value is estimated at a total of $100 million.

The AMADEUS system will provide a pan-European travel reservation system which will allow a consortium of European airlines and travel operators to effectively compete against the Americans.

The American airlines via their SABRE system are already encroaching on European turf and the new system will facilitate one-stop shopping for travel reservations.
The airlines in the pan-European consortium include SAS, Air France, Lufthansa, KLM, AI Italia, Sibena and British Airways.

Also in the commercial sector, Grumman Data Systems has been awarded a $40 million SI contract with Rolls Royce in the U.K. for development of engine testing systems.

In the public utilities sector the Belgian National Electricity Authority has awarded a $20 million contract for the development of a Ranger Energy Management System. Ferranti is the prime contractor and the two major subcontractors are the Belgian company TRASYS, who will undertake software development, and TRACTEBEL, who will be responsible for project management from the Belgian end.

In the U.K., the C.E.G.B. (Central Electricity Generating Board) is expected to award a $140 million contract for an energy management system. Control Data Corporation and Ferranti are currently bidding for the contract.

In addition, the Metropolitan Police (Scotland Yard) have awarded a $30 million contract to develop the CRIS project. Systems Designers, a U.K.-based software house, are the prime contractors.

In the defense sector, ICL and Computer Sciences Corporation (CSC) have been jointly awarded a $60 million contract to develop the UKAIR networking project for the R.A.F. This is a good example of a joint venture approach towards SI.

In addition, Digital has been awarded a $20 million contract to develop a communications network for the U.K. M.O.D. The principal sub-contractor is Atlantic Network Systems.

Although these above examples are by no means exhaustive in terms of the total Western European SI market, they provide prima facie evidence of not only trends in market size and growth but also of the major players and approaches towards market development.

For further background of trends and approaches towards addressing SI opportunities, readers are recommended to consult INPUT's report entitled *European Professional Services Markets - Trends and Opportunities, 1987-1992*, which focusses on SI.

c. New Methods of Working - A Major Inhibitor

The key issue in CASE implementation is that it entails a significant change in existing methods of working. CASE implementation entails a major management commitment to ongoing training and support programmes. In addition, CASE needs to be regarded as a long-term investment with a long-term payback in terms of quality and productivity.

INPUT's research indicated that the introduction of CASE tools has struck a chord of dissonance in some innovative organisations as analysts and designers have tried to fit new tools into old methods of working.

CASE implementation has also been slowed by user dissatisfaction with products that do not adequately support all phases of the development lifecycle.

Many early CASE tools (e.g., analysts' toolkits) are used as fancy drawing tools only and the full capabilities of advanced products such as Softlab's Maestro are underutilised, e.g., used as a 4GL only.

INPUT's research indicated that many organisations were simply not ready for the implementation of CASE, especially those who were not sensitive to the implementation of methodology or only working on relatively small projects.

The CASE approach is essentially a long-term investment for organisations and not a short-term fix. Consequently, the justification of expenditure is seen as risky by many organisations.

Implementing organisations have found that productivity actually drops with the first project using CASE tools. Consequently, the gain is longer term and comes when the central repository contains data systems of related information in the company that the CASE product can build on.

In addition, the biggest productivity gains have been made by those organisations who already have a methodology in place and a commitment from management to support it.

To get the full benefits from CASE a company also needs to have a mature and capable management with a sound reporting structure in place that is able to trust each of its elements. In other words, a company must have a system for management, i.e., a mature professional working environment, as well as a system for design.

Although these factors may well act as a barrier to rapid innovation of CASE technology, innovative users of advanced CASE products have experienced substantial benefits in terms of:

- Development productivity.
- Quality.
- Software portability
- Maintenance and change control.

A five to ten percent per annum steady improvement in productivity justifies the investment in CASE products and supporting services alone. INPUT foresees that this message will spread rapidly in the commercial sector.

A futher inhibiting factor which relates to organisations' inability to justify the investment in CASE is the lack of effective methods for measuring productivity and standards for software productivity metrics.

However, the sooner organisations come to realise that the important measure of software producitvity is system effectiveness and quality, the sooner the industry will adopt the CASE approach.

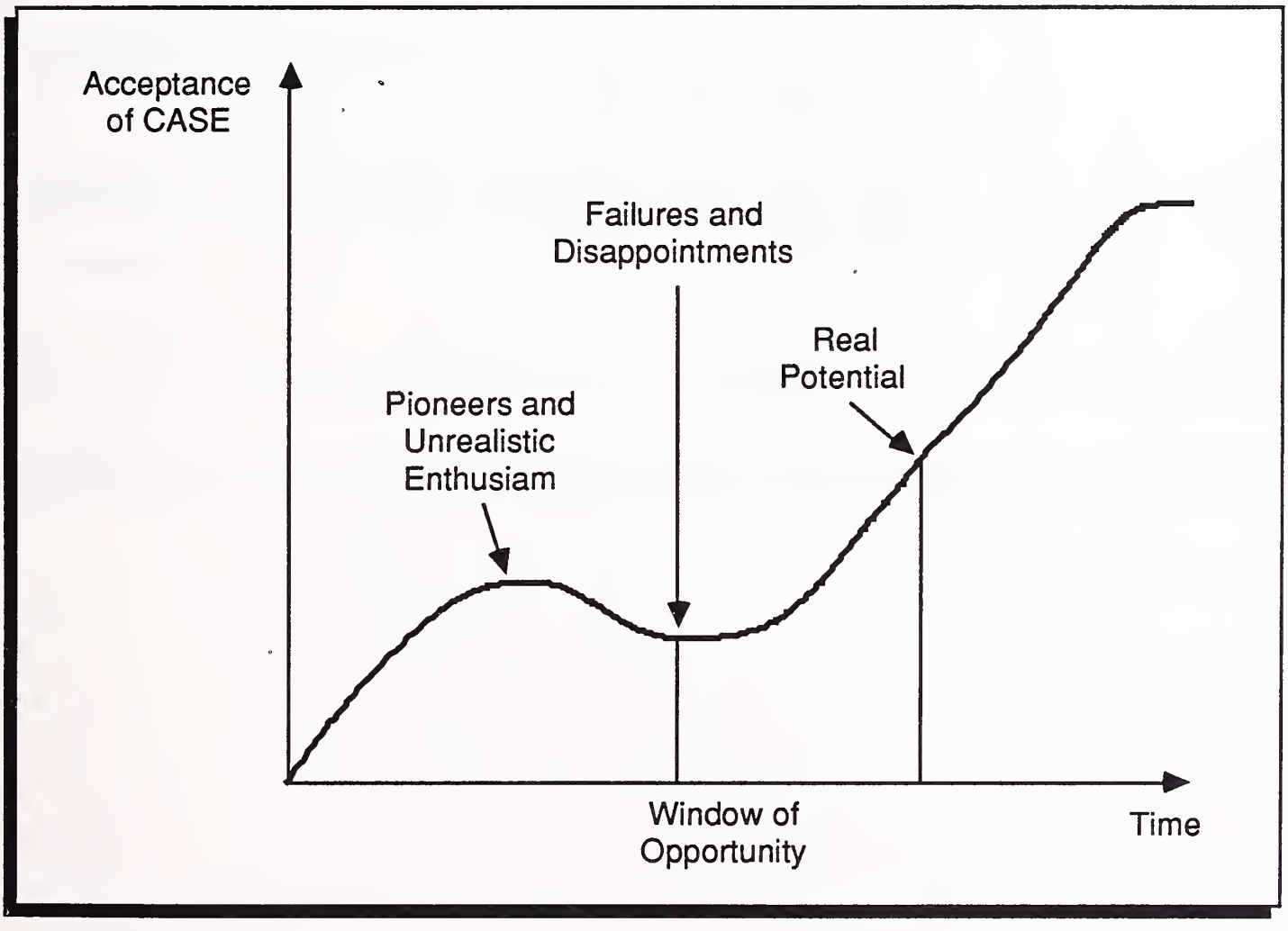
As one innovative CASE user commented ". . . we have acquired CASE products to improve system quality. . . the time factor is a secondary consideration . . .".

Innovative users in the commercial sector have tended to experiment with CASE generally on smaller, peripheral systems. However, once having learned the technology and methods of working, it is anticipated that CASE approaches will be adopted towards developing core operational systems - the next step in the industry's evolution.

Exhibit III-12 illustrates the life cycle of CASE innovation. Innovative users in the early 1980s have been disappointed with lack of CASE tool connectivity and have experienced resistance on behalf of systems developers to change methods of working.

EXHIBIT III-12

CASE
NEW TOOLS, OLD APPROACHES



However, INPUT uses a window of opportunity for vendors of advanced CASE products in the late 1980s. Resistance to change is gradually eroding due to market education and awareness. Products are more advanced and supporting services more sophisticated.

The evolution towards CASE has already gathered momentum by 1987 and INPUT foresees real growth and profit potential for the future.

3. Market Size and Growth by Country and Delivery Mode

INPUT forecasts that the Western European market for CASE software and services (as defined) will grow from a base of $85 million in 1986 to $140 million in 1987, reaching $270 million by 1989 at an annual average growth rate of 47%. The five-year projection shows a market size of $750 million by 1992, which represents an annual average growth rate of 41% for the period 1989 to 1992.

The overall factors that are driving and inhibiting this growth have been illustrated in Exhibit III-9.

INPUT's forecasts for the Western European market shown by country market and delivery mode are illustrated in Exhibits III-13 and III-14, respectively.

EXHIBIT III-13

WESTERN EUROPEAN CASE SOFTWARE AND SERVICES MARKET FORECAST, 1987-1992 (Analysed by Country Market)

COUNTRY MARKET	MARKET FORECAST ($ Millions)					
	1986	1987	1986-1987 AAGR (Percent)	1989	1986-1987 AAGR (Percent)	1992
France	15	26	54	55	43	160
United Kingdom	15	26	49	50	39	135
West Germany	25	37	38	65	35	160
Italy	8	13	46	25	44	75
Benelux	6	10	49	20	40	55
Scandinavia	8	12	46	25	44	75
Rest of Europe	8	16	55	30	44	90
Total Western Europe	85	140	47	270	41	750

EXHIBIT III-14

WESTERN EUROPEAN CASE SOFTWARE AND SERVICES MARKET FORECAST, 1987-1992 (By Delivery Mode)

DELIVERY MODE	MARKET FORECAST ($ Millions)					
	1986	1987	1986-1987 AAGR (Percent)	1989	1986-1987 AAGR (Percent)	1992
Software Products	50	80	40	140	38	370
Professional Services						
Training	20	35	59	80	46	250
Consultancy	15	25	49	50	38	130
Total	85	140	47	270	41	750

C

Trends, Issues and Opportunities in Key Market Segments

1. Overview

In general, CASE software and services has been (and will be) most effectively implemented by organisations that are already well advanced in terms of their use of standard methodologies and also in terms of their management organisation. That is, they have an established, mature professional management environment for system development.

Organisations which are less mature in terms of data processing organisation are not yet prepared for the culture shock of CASE; however, this position will evolve over time. Immature data processing organisations have tended to opt for end user tools, e.g., 4GLs, and many still continue with traditional techniques for data processing development.

2. Country Market Commentary

a. U.S.A.

The development of CASE within the U.S.A. is discussed in more detail in Chapter V of this report. However, it should be emphasised that the size, scope, and maturity of the market in the U.S. faces significant opportunities for vendors of CASE software and services.

North American organisations are generally more willing to adopt new technology and experiment with new methods of working.

Consequently, from a marketing perspective, the sales cycle can be shorter than in Western Europe as less time is spent in detailed product evaluation before commitment to purchase.

In addition, North American organisations are prepared to commit resources to effective implementation. U.S. companies spend on average 3% of turnover on employee education and training, i.e., at least 50% more than the majority of European organisations and 500% more than most U.K.-based organisations.

b. West Germany

West Germany is currently the largest European market for CASE software and services. This is largely due to a cultural preference for engineering discipline and commitment to structured methods and training. The German market has also been fostered by government support, especially by the finance ministry's decision to allow accelerated depreciation of software tools.

However, it should be noted that vendors seeking to penetrate the West German market will be faced with high bidding costs because of the German preference for undertaking detailed evalutaion of products prior to purchase.

c. Italy

The Italian market for CASE software and services is currently small and immature. However, with reference to Exhibit III-13, INPUT foresees strong growth potential in the late 1980s onwards.

Generally speaking, the Italians are very creative in the field of software development and there is already high use of CASE technology in the defense contracting community, e.g., CELINA.

In the commercial area, however, Italian data processing organisation is less sophisticated than in Northern European countries, except in the larger multinational companies, e.g., Fiat and Olivetti.

Consequently, many organisations have adopted a 4GL approach both for end-users and development staff as a cost-effective approach towards smaller projects.

Vendors seeking to address opportunities in Italy must be aware of the need to work with local players, either in joint venture partnership or as distributors, because of the nationalistic culture.

Another barrier to CASE market development in Italy is lack of commitment to training and effective implementaiton by Italian organisations. Consequently, vendors need to be able to offer a comprehensive umbrella of support services when addressing this potential opportunity.

d. United Kingdom

The U.K. market for CASE is currently immature but there is already significant momentum behind its innovation in the commercial sector.

CASE is increasingly being recognised as the only effective solution to the problems of decentralized data processing development, i.e., lack of standards, lack of quality control, ineffective communications, ineffective project management.

There are strong opportunities for CASE software and services vendors among organisations that are developing high volume, transaction-processing applications, such as the banking and finance sectors, the insurance sector, central government and also manufacturing (e.g., process control).

Until recently, CASE could be described as a "Cognoscenti Market" in the U.K.; however, growth is being encouraged by government initiatives and vendors missionary marketing activities. There are already signs of significant implementation of CASE outside the traditional defense contracting arena.

e. France

Given the large overall size of the software and services marketplace in France, the size of the CASE market is very small and immature.

To an extent, the lack of adoption of CASE technology by French user organisations has fueled the demand for custom developed systems subcontracted to the large SSII organisations.

Driving forces behind the adoption of CASE in France are:

- Decentralized data processing development.
- Cultural preference for structured analytical techniques and methods.

A major inhibitor on the development of the French CASE market to date has been the preference for unique design methods, (for example, MERISE), which are difficult to automate.

To date, French CASE product developers have tended to focus their attention around the Esprit programme, for example, the Development of Emeraude.

However, it is anticipated that CASE will be increasingly adopted by the manufacturing community and also financial services organisations.

f. Scandinavia

The Scandinavian market for CASE products and services is currently very small and immature. To an extent this is a reflection of immaturity in data processing organisation and, more importantly, a lack of effective government endorsement of CASE.

Nevertheless, the market has strong potential for future growth. Generally, the Scandinavians are:

- Flexible to changes in technology.
- Increasingly using methodologies in analysis and design.
- Prepared strongly to commit to effective implementation.

Vendors seeking to address pockets of market opportunity (e.g., Finland) should be aware of nationalistic preferences to contract with local players. Consequently, strategic partnerships are the most appropriate approach to market entry.

g. Benelux

For the benefit of clients seeking to evaluate this potential marketplace, the Benelux group of countries can be roughly subdivided between Belgium (one-third) and the Netherlands (two-thirds).

The Dutch market is generally least conservative in terms of innovation of new technology and has high growth potential.

However, in terms of the CASE marketplace, Belgium has the largest initial development potential owing to the size and strength of the financial services community. In addition, Belgium is the European headquarters of many multinational organisations.

h. Rest of Europe

The major country markets falling within this grouping are Switzerland, Spain, Austria, Portugal and Ireland.

In terms of CASE software and services opportunities, INPUT's research revealed opportunities in Switzerland and Spain.

In Switzerland, overall opportunities are small and specific, for example, the financial services community. However, they represent a good short-term business development opportunity.

The Spanish market has larger potential in both the short and especially long term. Deregulation in the Spanish financial services community is creating opportunities for many European vendors.

Again, strategic partnership is recommended as the best method of market entry.

3. Market Sector Commentary

Exhibit III-15 illustrates INPUT's assessment of existing and potential market opportunities for advanced CASE products and services in terms of vertical markets.

EXHIBIT III-15

CASE PRODUCTS AND SERVICES "HOT VERTICALS"

END USER MARKETPLACE

EXISTING MARKETS
- Defence
- Banking
- Aerospace
- Electronics

HIGH GROWTH POTENTIAL
- Civil Government
- Insurance
- Telecommunications
- Pharmaceuticals
- Manufacturing - Process and Line Assembly
- Transportation

OEM MARKETPLACE
- Systems Integrators
- Hardware Manufacturers
- Software Houses
- Professional Services Companies

Innovative users have been organisations in the defense, electronics, and aerospace contracting communities. The need to deliver quality embedded software, on-time and within budget within the constraints of fixed price contracting has been a major force driving the adoption of CASE technology.

The banking community has also been a major innovator owing to high security risks and the necessity to use formal methodology especially for the development of safety critical kernels.

INPUT forsees high potential in other commercial sectors, especially organisations developing high-volume, on-line transactions processing systems.

The OEM marketplace is also a major opportunity, i.e., hardware manufacturers and software/systems houses, especially those organisations that are positioning themselves as systems integrators.

The OEM marketplace also offers significant opportunities for CASE product vendors in terms of distribution arrangements; for example, Cullinet distributes LBMS's products and services worldwide and Philips distributes Softlab's products in Europe.

4. Delivery Mode Commentary

a. Professional Services

With reference to Exhibit III-14 it is possible to observe that there are significant opportunities for vendors offering profesional services, either independently or in conjunction with the marketing of CASE software products.

Opportunities are available owing to the fact that the key implementation issues (with respect to CASE) are the difficulties and inertia attached to changing working practices. Generally, the costs and difficulties involved in organisational change attached to implementing information technology is much greater than the investment in technology.

i. Training

INPUT estimates that the market for CASE-related training will grow from a base of $35 million in 1987 to $250 million by 1992 at an AAGR of 48%.

INPUT's user research indicated that organisations across Europe were placing increasing emphasis on training, both of data processing and end-user staff.

Exhibit III-16 highlights some key issues, trends and opportunities with respect to CASE-related training.

Although CASE training is a major opportunity and growth area, the key issue that is inhibiting growth is the shortage of skilled trainers, i.e., people with technical knowledge and interpersonnel skills.

To a certain extent this problem is heightened by prevailing attitudes in the industry that regard training skills as being tangential to career and

EXHIBIT III-16

personal development. In addition, the market is inhibited by attitudes in many organisations that training is a drain on productive time and an inessential activity rather than a valuable long-term investment. However, these attitudes are gradually being eroded as organisations increasingly perceive the importance of creating quality environments for quality staff. Training is an essential ingredient in the personnel management solution to the problems of excessive staff turnover and high recruitment costs.

Government support is a major factor that is slowing the development of the CASE training market.

In France, the law obliges companies to spend 0.5% of their wage bill on 'formation initiale' and 1.1% on 'formation continue'. Any of this money not spent on training is forfeited to the treasury. French companies typically do more than the law requires and spend, on average, 2.1% of turnover on training. West German companies also place high emphasis on training.

In the U.K., the government has been slow to effectively back training initiatives and U.K. companies only spend, on average, 0.2% of turnover on training.

However, there are signs of changing attitudes and important initiatives, for example, the NCC's Tool School which has DTI sponsorship. In addition, there are an increasing number of organisations, i.e., software houses, professional services companies and independent training companies, that are offering courses on structured methods and organisation as well as specialist CASE courses, e.g., quality assurance management.

To date the most popular software tools, and hence more popular training courses, are those which are easy to use and therefore easy to learn, for example, project management. However, there is a momentum and commitment towards training in methodology and a re-orientation towards thinking in engineering disciplines.

In terms of opportunites for CASE training vendors, a major trend in the market is the move away from public to specialized, on-site exclusive company courses.

Custom-built courses are higher margin, quality can be more closely controlled, and effective product differentiation is more easily established via meeting precise customer needs and monitoring progress with follow-up reviews.

This is especially true with respect to CASE training in that CASE has to be effectively implemented within the framework of an organisation's unique style, organisation, and culture.

A further major opportunity is in the area of management/personnel skills training. As the traditional barrier between analysts and programmers gradually erodes and systems developers become systems builders and business analysts, management skills will become a key priority for personal development.

A further key priority for CASE training vendors is to incorporate CBT elements into ILT courses. Vendors must be conscious of the low barriers to entry into this marketplace and CBT offers just one of many opportunities to differentiate via quality courseware and quality service.

ii. Consultancy

INPUT estimates that the market for CASE-related consultancy will grow from a base of $25 million in 1987 to $130 million by 1992 at an AAGR of 39%.

Although, like training, consultancy is a major opportunity, its growth is inhibited by supply-side resource contraints.

Consultancy should be an essential element of a CASE product vendor's offering. Many vendors' products and services currently support the production or the management of software but not both. The essential opportunity to address is effective CASE implementation.

The rationale behind this argument is that the major factors impacting on software quality and productivity in order of importance and priority are:

- Management style and methods e.g., change control, quality assurance, team organisation, cost control, project management.
- Methodologies employed.
- Tools used.

Owing to the fact that many organisations are not standardised in terms of their use of structured techniques, tools, management reporting and review procedures etc., there are ample opportunities for independent consultancies to give advice on:

- Structured techniques and methodologies.
- CASE technology evaluation and selection.
- Project management techniques.
- Data processing development organisation and procedures
- Quality assurance management.

Exhibit III-17 highlights some key issues and trends with respect to CASE-related consultancy.

The major opportunity to address is user confusion with the plethora of products and services available on the CASE market.

The major challenge is to help organisations ease the problem of effective implementation. CASE has been seen as a risky technology by many organisations and the need for independent advice and inplementation assistance is critical.

EXHIBIT III-17

CASE CONSULTANCY KEY TRENDS AND ISSUES

- The Challenge of Effective Implementation
- The Need for Standardization
- The Overriding Importance of Changing Management Style and Methods
- The Need for "True" Independence
- User Confusion—Plethora of Solutions

INPUT suggests that independent CASE consultancy cannot be effectively operated by vendors of individual products, although it is a central plank of their pre- and post-sales support service.

INPUT suggests that vendors should consider joint ventures and marketing agreements with major professional services firms in order to address this opportunity.

Exhibits III-18 and III-19 are schematics which illustrate a software development methodology and the basic operation of a project control system, respectively.

The consultancy challenge is to ensure that these elements are effectively implemented, i.e., the appropriate management procedures and practices are in place to ensure their success from the viewpoint of all parties concerned (software developers, company management, and system users).

As one vendor suggested, the crisis in the software industry has been created by the organisational problem of ". . . management by those who don't understand what they manage. . . and understanding by those who can't control what they do. . .".

EXHIBIT III-18

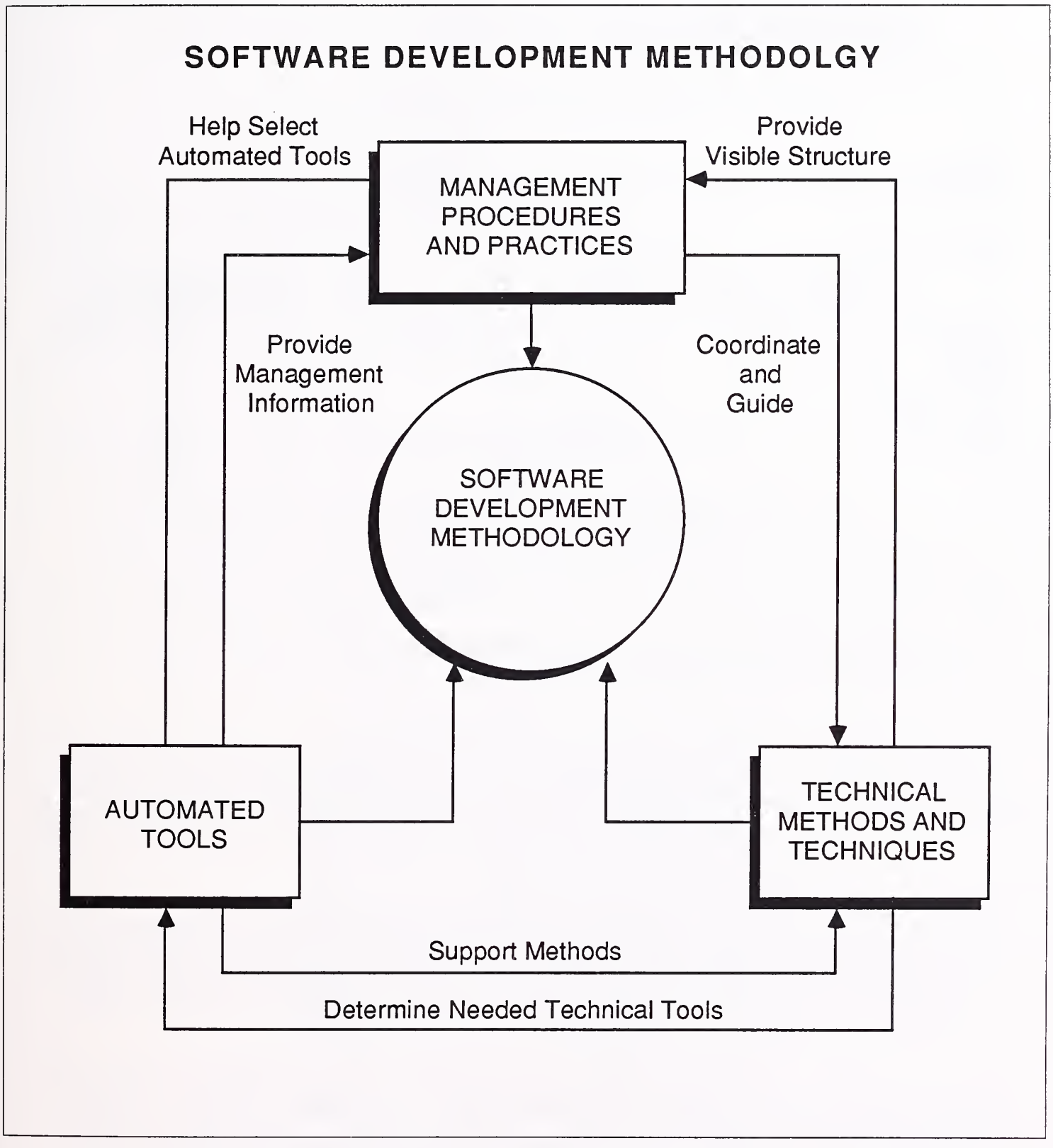
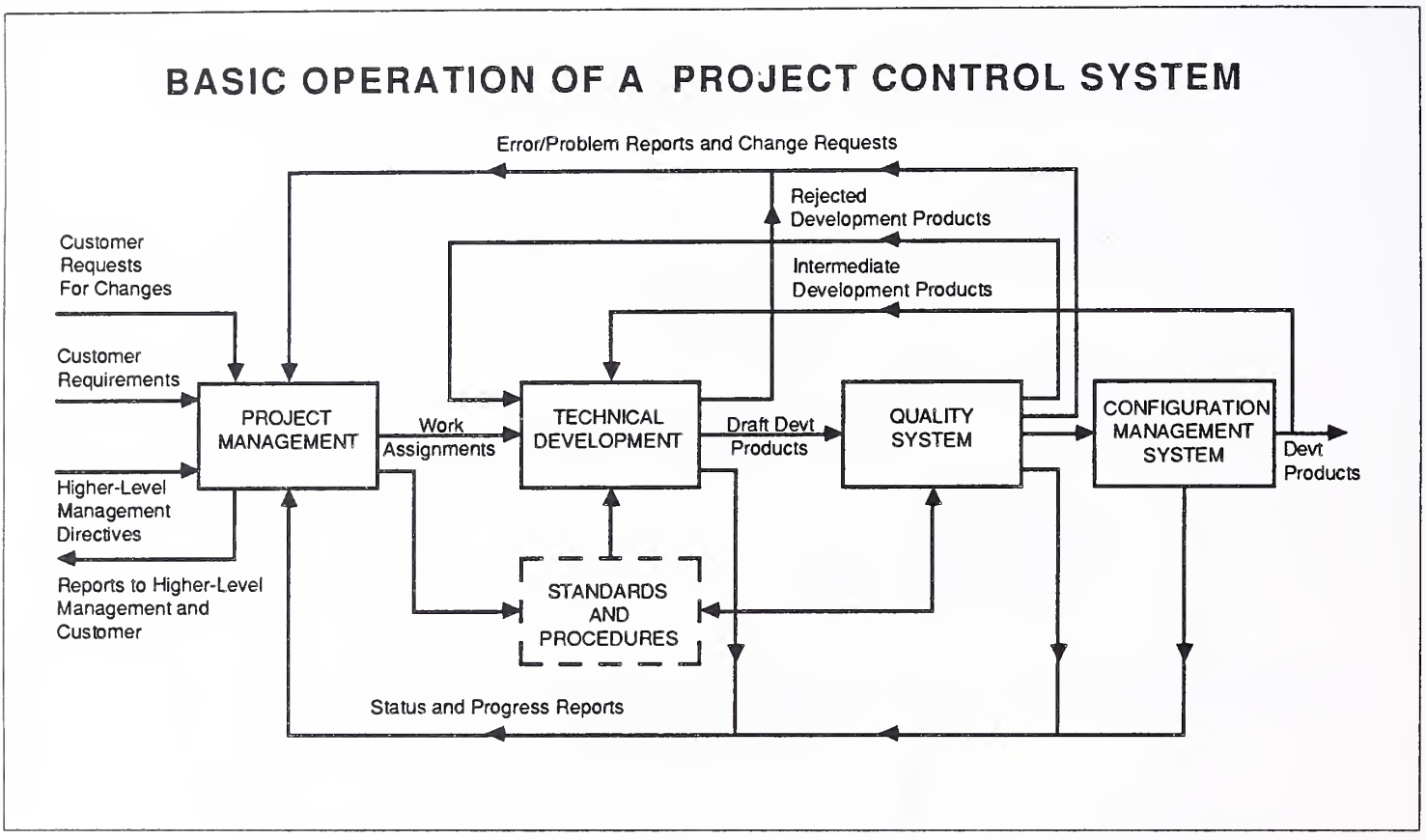


EXHIBIT III-19



b. Software Products

INPUT estimates that the Western European market for CASE software products (as defined) will grow from a base of $80 million in 1987 to $370 million by 1992 at an AAGR of around 36%. Exhibit III-20 highlights some key trends, issues and opportunities with respect to market evolution and development.

The development and evolution of CASE software products is discussed in more detail in Chapter V of this report. However, the major trend is the convergence towards meeting user demands for complete CASE systems.

INPUT's research indicates that the market for individual CASE tools and toolkits will shake-out. There are a plethora of products offering limited life cycle support which have been launched onto the market without precise clarification of ideas and management investment. Many of these tools are unconnectable in both functional and architectural terms.

EXHIBIT III-20

CASE SOFTWARE PRODUCTS TRENDS, ISSUES AND OPPORTUNITIES

TRENDS
- Convergence towards Complete CASE Systems
- Market Shake-Out
- Connectivity and Portability

ISSUES
- Standards

OPPORTUNITIES
- A.I. Linkages
- Maintenance Tools
- Parallel Programming Tools
- Integrated Analysts'/Programmers' Workbenches
- IPSE's

Successful CASE software product offerings provide a flexible environment for the support of all software development activities (including management), they also need to be portable and offer connectivity across a range of general commercial hardware architectures.

CASE software should provide effective 'office automation' for all an organisation's development techniques, methods, and tasks including those that have been developed internally; it should be sufficiently authoritative to encourage and support standardisation, but sufficiently flexible to incorporate a flexible approach towards the evolution of techniques, methods and tool support.

Connectivity is being enhanced by the development of fully-integrated products that can operate within a multi-vendor hardware environment of networked high-performance workstations linked to a central mini/mainframe environment.

Portability and data interoperability has been enhanced by the development of common standards and their endorsement by CASE software product developers.

The European Esprit programme has resulted in the development of the PCTE (Portable Common Tool Environment) standard.

A PCTE-based prototype system, Emeraude, is currently being implemented for the development of industrial applications in France. PCTE is not in itself a system but a specification for tools support interfaces and has already been approved by the European Space Agency (ESA) and other government bodies. PCTE also provides a common focus for European commercial tool developers and a stable environment for workbench/IPSE development.

The future requirement is to develop a common international standard which harmonises with the U.S. CAIS (Common APSE Interface Set) standard with the European standard. CAIS-1 became DOD Standard 1838 in 1986 and CAIS-2 is currently under development by Softech. The CASE tool industry needs to harmonise development and avoid the wasteful "Tower of Babel" syndrome.

Although there are limited future market opportunities for vendors of individual software tools generally, INPUT foresees significant opportunities for tools in the following areas:

- Artificial intelligence (A.I.) tools for the development of Expert Systems.
- Maintenance tools.
- Configuration management tools.
- Parallel programming tools and methods.

A.I. tools facilitate rapid prototyping of emerging commercial applications for expert systems, for example, process and network control.

There are already a number of proven products on the market which are now available on "COTS" (Commercial Off-the-Shelf) hardware, e.g., DEC VAX.

Leading products are ART from the Inference Corporation which is distributed in Europe by Ferranti. Others include KEE (Knowledge Engineering Environment) from Intell Corp.

The integration of CASE and knowledge-based systems (KBS) is a central plank of current research activity under the Esprit, Alvey and Eureka programmes when creating the next (third) generation of IPSEs. This third generation will incorporate a knowledge base incorporating the skills of experienced designers integrated with intelligent knowledge-based systems (KBS) tools. These IPSEs will support the different styles of system development and prototyping that are now being researched by the world's KBS community.

The next generation of IPSEs will also allow for the joint development of software and hardware. That is, they will include tools for computer-aided design of VLSI. They will form the basis for the future Information System Factories (ISFs) the ultimate software engineering concept.

For your benefit, the current range of activities encompassed by a (second generation) IPSE are illustrated in Exhibit III-21.

Essentially, an IPSE consists of a flexibility integrated set of software tools mounted on top of a database and interfaced to a variety of users via a common interface package. IPSEs facilitate information/data sharing and resource control across projects. They facilitate effective control of dispersed project teams on large projects or the coordinated control of large numbers of smaller projects. IPSEs are flexible and integratable with multiple methods of working. In addition, they are customisable and flexible in their tool scope, i.e. in-house and externally sourced.

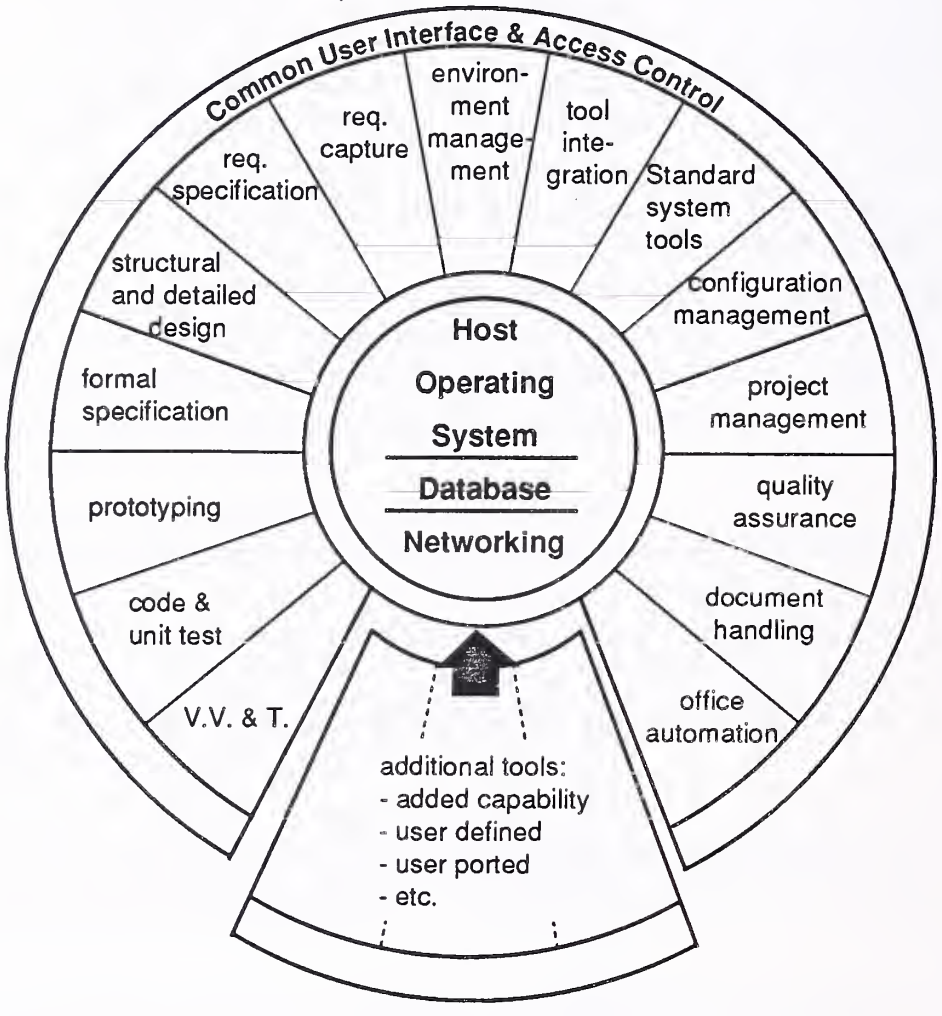
More recent IPSEs based on the PCTE project also provide access to distributed databases as well a central database from any workstation held anywhere in a network. Consequently, it is possible to conclude that IPSEs are the long-term opportunity as far as CASE software products are concerned.

Other short-term opportunities include the devleopment and use of maintenance tools. A major inhibitor to the adoption of CASE approaches is the lack of ability of user organisations to incorporate their existing programmes into new systems which are developed using CASE tools. "Structured retrofit" tools which address the problems of escalating maintenance costs will continue to flourish in the market.

Other longer-term opportunities for CASE product vendors include the development of tools and methods which accommodate concurrency. The software industry needs to keep pace with hardware developments and developments in parallel architecture; for example, the Japanese Fifth Generation Project, should be no exception.

EXHIBIT III-21

ACTIVITIES ENCOMPASSED BY AN IPSE



A very major and shorter-term opportunity in terms of CASE products lies in the integration of analysts and programmers workbenches which offer a cost-effective solution to organisations' developing small/medium-sized systems.

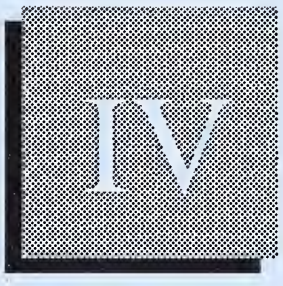
While the IPSE concept originated from the need for ultra high system reliability and performance in complex realtime projects (e.g., defense and aerospace); workbenches have evolved from the traditional data processing environment where shortage of programmer resources and associated productivity problems are the critical issue.

IPSE and workbench developments are essentially complimentary as workbenches evolve to cover all life cycle phases and development activities.

For medium-sized systems developers the advantage of the workbench approach is that it is incremental and integratable with existing investments in databases, languages and hardware.

In terms of the future, however, flexibility is the key to CASE software product success. Flexible IPSEs are the way forward for large project development. Flexible workbenches will increasingly play a role in the development of smaller commercial systems.

The next chapter provides a comprehensive analysis of the user environment with respect to CASE. It includes an analysis of INPUT's market research and highlights trends, issues, and opportunities in market development.



User Environment

User Environment

This chapter provides a summary and analysis of INPUT's user research and highlights trends and issues impacting end user organisations with regard to software development. It also highlights areas of business development opportunity for existing and potential vendors of specialist CASE (Computer Aided Software Engineering) tools and application development products.

As part of INPUT's Information Services Programme research, a sample of 210 senior data processing professionals were interviewed by telephone. An analysis of the research sample is given in Appendix B. The questionnaire used as the basis of these interviews is given as Appendix C.

The purpose of this survey was to access levels of awareness, usage, and attitude towards the development of information services and to highlight changes and trends across the population.

With respect of systems development productivity the major areas covered were:

- Trends in the size of applications backlogs and perceptions of factors that are impacting on those trends.

- Levels of usage, planned utilisation, and attitudes towards CASE tools, systems development methodologies, and other applications development tools.

- Perceptions of benefits and drawbacks associated with the use of CASE tools and methodologies.

- Estimates of time saved on development projects by using productivity tools and techniques.
- Perceptions of effectiveness of various potential approaches towards improving the productivity of data processing development staff.
- Perceptions of effectiveness of various potential approaches towards the support of end-user development productivity.

It must be noted that INPUT is cognizant that the user environment for CASE tools is not restricted to organisations whose primary business focus is outside the computer industry. Issues and trends impacting the implementation of CASE amongst information technology suppliers have been discussed elsewhere in this report. However, it is interesting to note that many of our research findings among non-IT businesses are also applicable to the user environment amongst IT suppliers.

A Macro Issues—User Environment

Based on INPUT's ongoing research programme into the user environment it is possible to isolate three major strategic concerns for data processing:

- The challenge of integration.
- The search for productivity.
- The management of people.

1. The Challenge of Integration

A major on-going trend has been the decentralisation of the data processing function within organisations and a shift in systems and support from the centralised information services centre to the point of work (POW).

Systems are increasing in their organisational pervasiveness and spreading into production, distribution and marketing. This poses the challenge of effective corporate communications as users seek to link up disparate systems in operating divisions, subsidiaries, customers and suppliers. In addition, the growth of end-user computing is leading to seemingly insatiable demand for online information support systems.

The challenge of integration is therefore the effective management of technological complexity. Users increasingly need to integrate hardware of differing manufacturers with data bases of different structures, with software products of differing vendors into a single, cohesive, coherent

system that supports the ongoing operational management of a corporation.

2. The Search for Productivity

Changes in the complexity and competitiveness of the business environment is leading to demand at the highest level of organisations for a more productive utilisation of current corporate assets, including capital investments, human resources and information assets.

Essentially, this poses three major problems for data processing: firstly, meeting deadlines for applications developments; secondly, meeting demands for new applications; thirdly, meeting demands for system quality.

With overall demand for new applications growing by approximately 45% per year and the number of skilled staff growing by only approximately 5% per annum, there are essentially three potential solutions:

- CASE Tools.
- Training.
- Effective broadbased management of IS productivity and performance.

Currently, there are very few adequate measurement systems for IS productivity and despite all the tools and application development products that have been put on the marketplace the backlog of unimplemented applications is continuing to grow.

The real solution is to focus on quality. Essentially, this means working with the people and focussing on the overall efficiency of applications to support corporate information requirements, rather than focussing on the limited objective of speeding the development of application code.

3. The Management of People

Changes in the social environment lead to a general perception of gradual decrease in the quality and, above all, dependability of the workforce to cope with an increasingly complex environment.

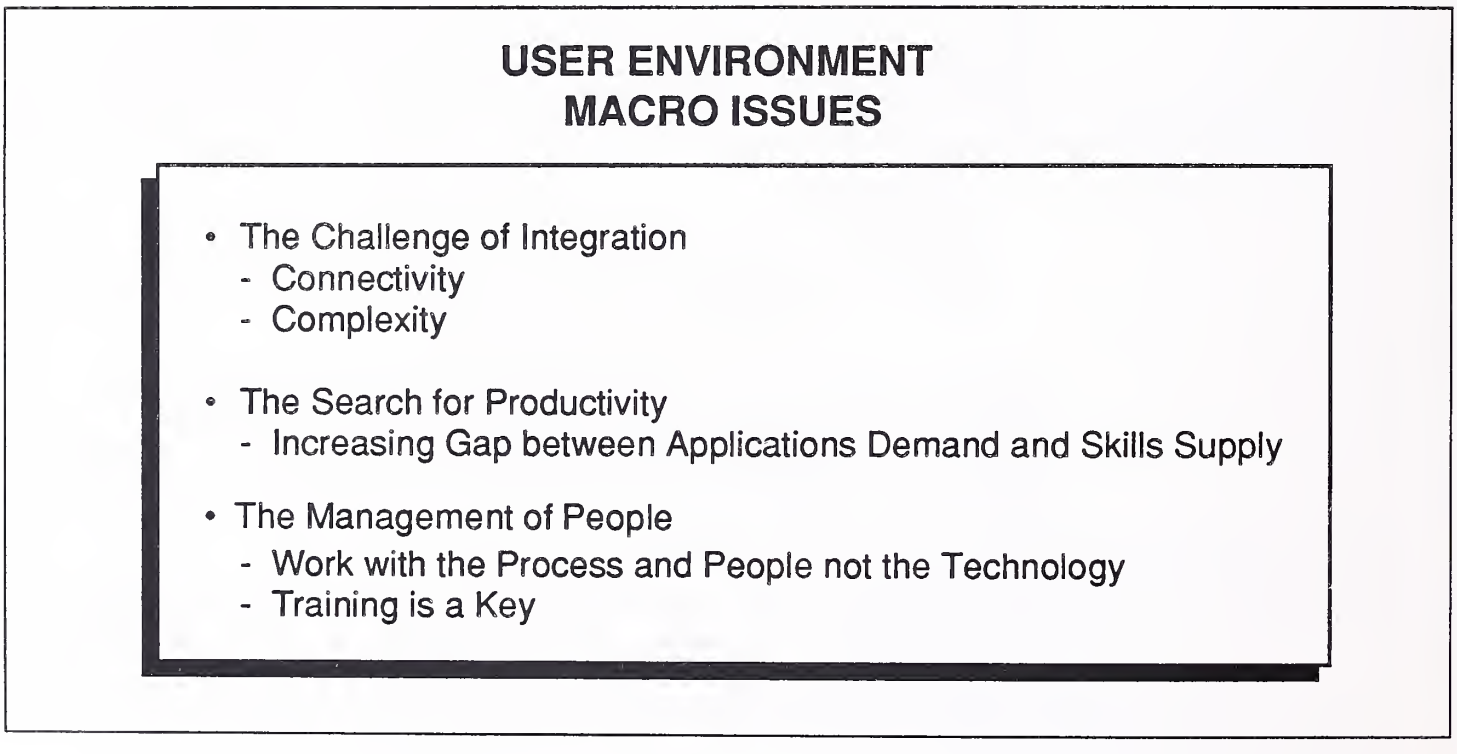
A key concern and priority is the training not only of data processing staff but also of end-users. In addition, there is a growing need for systems that

stress education and training as part of their service offering in order to adequately support management information requirements.

The raison d'etre behind this evaluation of user priorities is the gradual recognition that the costs and difficulties involved in organisational change attached to implementing information technology is much greater than the investment in technology.

With respect to software development the key to success lies with the implementation of effective procedures and policies for developing motivating and managing personnel. Many organisations are now realising that investment in tools is not an 'Aspro solution'.
INPUT's view of the macro issues impacting the user environment are given in Exhibit IV-1.

EXHIBIT IV-1



B

Trends in Applications Backlog

The need for the CASE approach towards software development is highlighted by INPUT's findings with respect to changes in the applications backlog between 1986 and 1987.

Exhibit IV-2 and IV-3 illustrate INPUT's findings analyzed by industry sector, country market and establishment size.

Overall, the research revealed that 86% of organisations had failed to reduce the backlog of applications. This trend is particularly marked amongst organisations in government, insurance, and manufacturing.

EXHIBIT IV-2

CHANGES IN THE APPLICATIONS BACKLOG 1986-1987 (Analysed by Industry Sector)

INDUSTRY SECTOR	SAMPLE SIZE	PERCENTAGE SPLIT		
		INCREASED	DECREASED	REMAINED SAME
Process Manufacturing	25	[60]	16	24
Discrete Manufacturing	28	[57]	07	36
Retail Distribution	19	26	[21]	53
Finance and Banking	38	45	18	37
Insurance	18	[50]	06	44
Health Care	14	29	14	57
Government	26	[46]	15	39
Transport	22	32	18	50
Utilities	20	40	05	55
All Sectors	210	44	14	42

EXHIBIT IV-3

CHANGES IN THE APPLICATIONS BACKLOG 1986-1987 (Analysed by Country Market and by Established Size)

CATEGORY OF ANALYSIS	SAMPLE SIZE	PERCENTAGE SPLIT		
		INCREASED	DECREASED	REMAINED SAME
ANALYSIS BY COUNTRY				
West Germany	40	43	14	43
United Kingdom	40	43	12	45
France	30	43	07	50
Scandinavia	40	[53]	15	32
Italy	30	40	10	50
Benelux	30	43	[23]	34
All Countries	210	44	14	42
ANALYSIS BY ESTABLISHMENT SIZE				
Greater than 1000 Employees	127	40	17	43
Between 500 and 1000 Employees	83	[51]	10	40
All Establishments	210	44	14	42

The perennial problems of the software community to satisfy growing end-user pressure for new applications and solve the problems of delivering high-quality systems, on-time and within budget is common throughout Western Europe. The research reveals that it is particularly marked in France and Italy, and also, interestingly enough, in medium-sized establishments.

It is interesting to note the comment of one vendor respondent (Director - U.S. software house): "...the backlog of applications now waiting to be developed is greater than the entire number of applications that have been generated in the last twenty years of computing. . ."

C

Users' Perceptions of Major Constraints on Their Ability to Reduce the Applications Backlog

Exhibit IV-4 illustrates a synthesis of the comments most frequently mentioned by users as inhibitors on their ability to reduce the backlog.

INPUT's analysis confirms general perceptions in the industry about the growing gap between skills supply and applications demand.

EXHIBIT IV-4

USERS' PERCEPTIONS OF MAJOR CONSTRAINTS ON ABILITY TO REDUCE THE APPLICATIONS BACKLOG
(Total Western Europe - Synthesis of Most Frequently Mentioned Comments)

RANK	COMMENT	NO. OF MENTIONS
1	Increased Number of Applications	27
2	Shortage of Qualified Staff	25
3	Growing Demand for Management Information	21
4	Changes in the Marketplace	10
5	Resource Shortages/Budget Restrictions	8
6	Reorganization/Decentralization of Data Processing Department	6
7	Difficulties in Implementing CASE Products	4
8	Difficulties in Maintaining Existing Programmes	4

N.B. Number of Respondents = 105

Shortage of qualified staff was mentioned as a key inhibitor and each site interviewed typically had at least five technical staff vacancies.

A further major inhibitor and problem for data processing developers is the rapid increase in demand for on-line information support systems which reflect the trend towards end-user computing and the growing strategic importance of IS to corporate success.

Another inhibitor frequently mentioned was the increasing accountability of data processing to respond to changes in the business environment. IS are becoming a competitive weapon in many organisations and our research reveals that data processing departments do not have sufficient resources and flexibility to adjust.

In many of the establishments interviewed by INPUT, expensive development resources were being tied up in the maintenance of existing systems (especially IBM sites) that had been developed using "seat of the pants" analysis and design techniques and hand coded using traditional technologies (e.g., COBOL). Documentation of these old, but strategically important systems, is also poor - a factor that exacerbates the problem even further.

In the current data processing environment as much as 70% of the cost associated with applications software is in its maintenance. A key opportunity for proponents of the CASE approach to address.

Several users adopting a CASE solution pointed to significant difficulties with the practical implementation of the new breed of intelligent software tools.

CASE implementation effectively means fundamentally changing the way people work; implementors of software automation have found (like factory automation) that there is a considerable amount of conservatism and resistance to change in the data processing environment.

Users pointed to the high costs of training which, in the short term, is greater than the productivity and quality benefits of the software.

The "skills gap" in terms of availability of trained software engineers experienced in implementing systems using CASE techniques is so large that many organisations have resorted to traditional methods, i.e., what they know and understand. Data processing managers are notoriously conservative when it comes to changing working methods.

D

Users' Perceptions of Major Factors Enabling Reduction of Applications Backlogs

Exhibit IV-5 illustrates a synthesis of the comments most frequently mentioned by users' as drives on their ability to reduce the backlog.

INPUT's analysis reveals that users see the utilisation of automated software technologies as a central plank in their battle with the backlog and undoubtedly a limited number of organisations have reaped benefits from the usage of development tools.

EXHIBIT IV-5

USERS' PERCEPTIONS OF MAJOR FACTORS ENABLING REDUCTION OF APPLICATIONS BACKLOG
(Total Western Europe - Synthesis of Most Frequently Mentioned Comments)

RANK	COMMENT	NO. OF MENTIONS
1=	Increased Usage of Externally Sourced Packaged Software	6
1=	Increased Usage of Software Development Tools and Sophisticated Methods	6
3	Careful Management Controls	4
4	Use of Fourth Generation Language	3
5=	Fruition of IT Strategy/Automation Plan	2
5=	Increased Numbers of Skilled Staff	2
7	Improved Standards/Discipline	2

N.B. Number of Respondents = 26

The results, however, shed some light on the relative importance of tools in the productivity equation. Effective broadbased management is a further key factor for success, i.e.:

- Linking the IS strategy with corporate strategy.
- Careful implementation of management controls on development.
- General improvement of quality in terms of standards for development team practice and skill levels.

Users' comments regarding the importance of fourth generation languages are a reflection of the market status in terms of the introduction of development tools.

Many users of tools have chosen to adopt an incremental approach towards automating the development process. Conservatism, inertia, and initial bad experiences with CASE approaches have led users to seek incremental improvements via the utilisation of 4GLs, DBMS and code generation/structuring tools which are being increasingly front-ended by screen handles and document generators. The integrated CASE approach, it would appear, is still in its infancy.

E

Usage and Attitude towards Integrated Project Support Environments (IPSEs)

INPUT's research revealed that the status of the market for integrated CASE products that offer full life cycle support is currently at an early stage of development.

Only three of the 210 establishments interviewed by INPUT were using what INPUT would consider to be an IPSE (for example, the BIS-IPSE, GENOS from GEC Software, ISTAR from Imperial Software or Maestro from Softlab).

The research findings revealed general confusion about what constituted an IPSE. For example, users with a stable of individual tools were inclined to regard themselves as IPSE users.

Of more telling importance is the fact that the research revealed a very deep lack of awareness of what an IPSE was. This clearly points to the need for management education and missionary marketing on the part of vendors seeking to actively address the potential market amongst users developing general commercial applications.

Surprisingly, levels of awareness of IPSE products were not higher in the U.K. than in the rest of Europe, where the defence, electronics, and

banking communities have demanded a sophisticated CASE approach towards software development and funded product development.

Despite these rather negative research findings INPUT sees a strong future for IPSE products throughout Western Europe. Software projects are increasing in size and complexity and the challenge of integration is becoming increasingly difficult. Larger organisations are reaping the benefits of fully-integratged CASE tools and INPUT believes that the IPSE approach could well become a standard for the 1990s.

An evaluation of IPSE products available on the market will be given in Chapter V.

F

Structured Methodologies—Market Penetration and Opportunities

Exhibits IV-6 and IV-7 illustrate INPUT's research findings with respect to market penetration and planned usage of structured methodologies.

It is interesting to note that planned usage is greatest in the public sector. Governments have placed high emphasis on standardisation and the benefits of common methodology to ease maintenance problems and improve quality.

1. West Germany

The research into the German market reveals a generally high awareness of the CASE approach and the benefits of methodologies and supportive tools.

However, the usage and planned usage of proprietary methodologies is only 35% of establishments.

Many establishments place a high emphasis on their own internally developed methodologies which they found to be more appropriate to meeting their specific requirements.

In addition, a number of organisations interviewed had already evaluated proprietary methodologies and found them to be inefficient.

2. United Kingdom

Levels of usage and awareness of structured methodologies in the U.K is the lowest in Western Europe. However, the U.K. offers the greatest potential (along with Italy) for future growth.

EXHIBIT IV-6

STRUCTURED METHODOLOGIES MARKET PENETRATION AND OPPORTUNITIES (Analysed by Industry Sector)

INDUSTRY SECTOR	SAMPLE SIZE	PERCENTAGE SPLIT		
		USE	PLAN TO USE	DO NOT USE
Process Manufacturing	25	16	[20]	64
Discrete Manufacturing	28	39	07	54
Retail Distribution	19	37	[21]	42
Finance and Banking	38	42	11	47
Insurance	18	39	11	50
Health Care	14	43	07	50
Government	26	42	[19]	39
Transport	22	36	00	64
Utilities	20	30	[25]	45
All Sectors	210	35	13	52

EXHIBIT IV-7

STRUCTURED METHODOLOGIES MARKET PENETRATION AND OPPORTUNITIES (Analysed by Country Market)

COUNTRY	SAMPLE SIZE	PERCENTAGE SPLIT		
		USE	PLAN TO USE	DO NOT USE
West Germany	40	28	07	65
United Kingdom	40	[10]	[18]	72
France	30	47	10	43
Scandinavia	40	50	13	37
Italy	30	37	[17]	46
Benelux	30	54	13	33
All Countries	210	35	13	52

In the government sector, the use of SSADM is increasing and being actively promoted by the Department of Trade and Industry in the commercial sector in tandem with its Software Engineering Initiative. In the defence contracting community, MASCOT is now a common standard.

However, many companies feel that methodology is only relevant to very large developments and heavy investment in training will doubtless curb the speed the innovation.

3. France

Levels of usage and awareness of the benefits of structured methodologies is high. French companies tend to place higher emphasis on the need for common methodology in order to improve quality standards and ease maintenance, rather than on CASE tools.

French users place heavy reliance on their in-house developed methodology and the French standard methodology, MERISE.

4. Scandinavia

Like France, the use of structured methodologies is high with a strong preference for in-house developed methods and domestically developed products and techniques (e.g., SAK and RAS in Sweden).

5. Italy

Italian organisations revealed a high awareness of the need for standardisation and common structured methodology.

In addition, there is high usage of imported proprietary methodologies, (for example, Jackson, Warner, and Information Engineering), rather than domestically developed methodologies such as ODATI.

However, the development of data processing is at a relatively innovative stage in Italy and many organisations felt that a 4GL approach was adequate for their current stage of data processing development, i.e., developing relatively small applications.

In addition, several organisations felt that structured methods are too rigid and impair the creativity required for software development.

6. Benelux

Usage of structured methodologies was high; however, most organisations were content to use in-house developed methods and those linked to the hardware manufacturers proprietary technology, for example, IBM's SDM.

7. General Market Commentary

Exhibit IV-8 summarises the most frequently mentioned comments made by users for not using proprietary structured methodologies.

In general, INPUT's research revealed that many data processing departments would like to use a proprietary methodology but do not have the time to retrofit the large complex systems that they must maintain to the new externally caused methodology.

EXHIBIT IV-8

USERS' REASONS FOR NOT USING PROPRIETARY METHODOLOGIES

(Total Western Europe - Synthesis of Most Frequently Mentioned Comments)

RANK	COMMENT
1	Own Standard Method Adequate
2	Staff and Internal Organisation not Prepared
3	Too Rigid
4	Proprietary Methodologies do not Address our Specific Requirements
5	Inefficient
6	Only Need for Large Developements

CASE products that have been successful in the marketplace (such as Maestro) have a flexible open architecture that is not wedded to a particular design methodology.

Consequently, users can ease the difficutlties in CASE implementation by integrating automation with their own developed methods, mainly for programming and more recently for analysis.

In addition, INPUT's research reveals a dissatisfaction with the comprehensiveness and specificity of proprietary methodologies for systems analysis. Users have discovered that the methods available on the market such as LBMS, JSD, and Yourdon are very useful as a means of describing a system and defining what deliverables are required but do not describe how they are to be achieved.

Proprietary methodologies and indeed tools supporting methodologies provide a way of producing data flow diagrams and data models and associated documentation automatically in tandem with the all-important consistency checking. Consequently, although they do not offer a total

panacea to the problems of development, they free up valuable time for analysts to focus on quality, i.e., meeting user requirements and ensuring system efficiency in line with the evolution of the company's business requirements.

The choice of methodology for a user organisation needs to be flexible, dependent on company culture, ease of access to users and the type of project, that is, whether the aim is a detailed, vigorous analysis of a system demanding a "bottom-up" approach, or a strategy study where "top-down" analysis is appropriate.

Although vendors' claims for the benefits of structured methods are grossly exaggerated with respect to time and cost saving, there are some important benefits to be realised in comparison with the all too common "seat of the pants" approach to analysis and design.

Structured methods enable:

- The integration of logical design and physical reality.
- Logical analysis to be undertaken prior to physical design.
- Standards to be set for quality.
- Enhanced role of user in development cycle.
- Physical deliverables to perform at an agreed level.

In summary, a system developed using a structured methodology has a longer life and the major pay-off is not in the early project stages but in the maintenance phase.

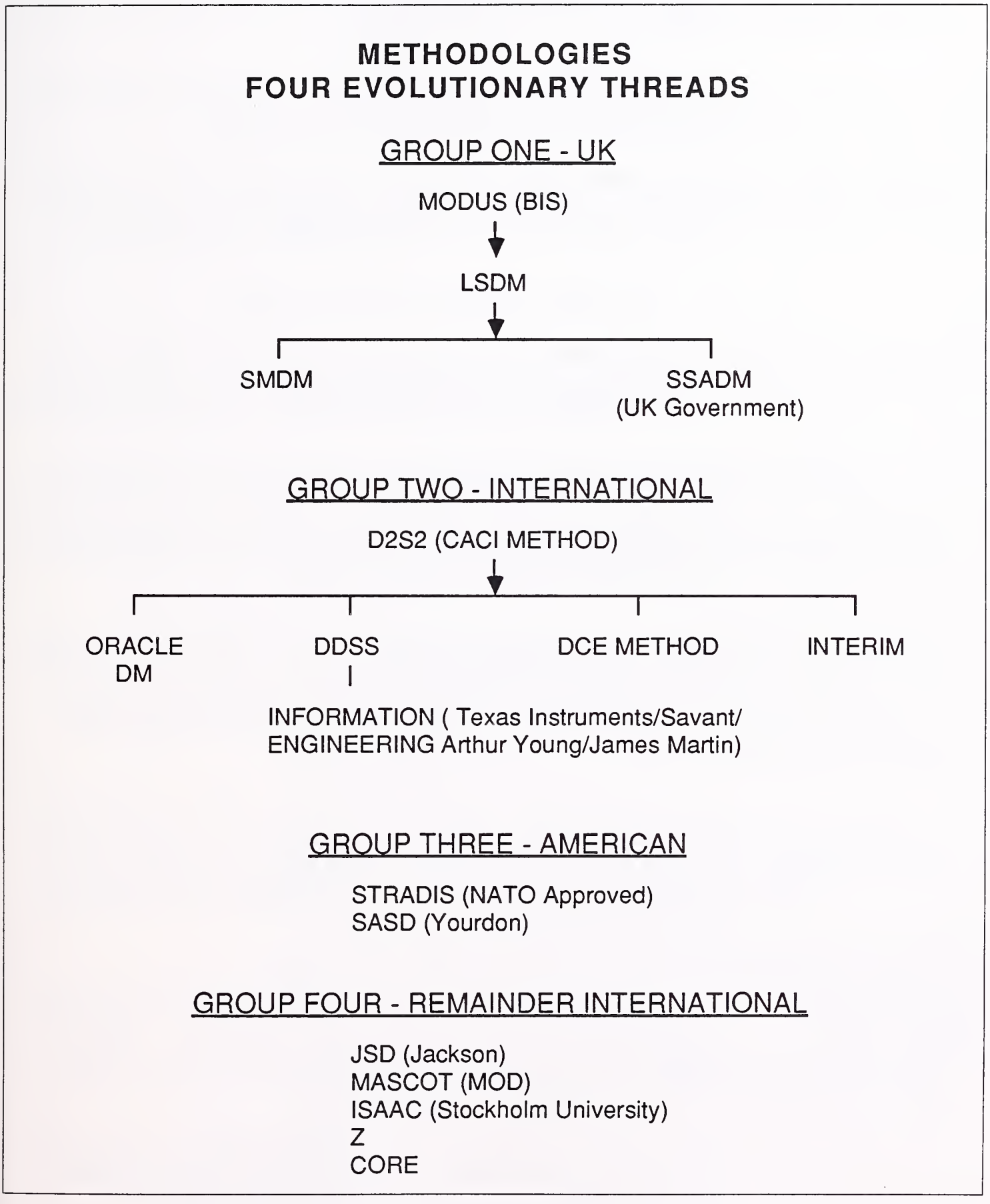
8. Summary Review of Methodologies

Essentially there are four types of methodologies used in IT:

- Systems analysis and design methods e.g., LSDM, SSADM, JSD and YOURDON.
- Strategic planning methods e.g., LEAP and BSP.
- Project control methods e.g. PROMPT and LIFESPAN.
- Office automation methods e.g., COMPACT.

With respect to systems analysis and design there are essentially four evolutionary threads or "stables" as illustrated in Exhibit IV-9.

EXHIBIT IV-9



INPUT's research revealed that the two market leaders are Information Engineering (James Martin) and LSDM. Both methods are excellent for centralised batch and on-line systems.

In addition, Jackson, Yourdon and Macot are all good for realtime systems with a heavy concentration on data processing.

Consequently, from a user perspective it is necessary to be flexible in one's choice of methodology as all methods do not support all environments, i.e., they only give coverage for 50% of computer systems.

9. The Future - Tools and Methodologies

The major opportunity for the future is with advanced CASE tools, i.e., analyst/programmer workbenches that support well-established methodologies, including in-house methodologies in an open design architecture.

There is also an opportunity to develop tools that allow the end user to have an increasing role in logical analysis and follow structured methods for process-oriented logic. User-friendly software is needed for end users (i.e., on PCs) which is menu driven and incorporates sophisticated graphics and tutorial dialogues.

Project management and systems development methodologies and techniques are now undergoing a rebirth after their automation on the ubiquitous PC and networked PCs. For example, Metier's Artemis Project product is one of the hottest worldwide sellers in the micro software market.

A further trend for the future is the link between CASE tools and knowledge-based systems engineering (i.e. A.I.). There is an opportunity for CASE workbenches to be linked to knowledge-based databases which encapsulate user experience and knowledge together with the best software engineering management practices.

G

Users' Attitudes towards CASE

As one vendor commented ". . . the current status of the computer software industry is like the aerospace industry in the 1950s . . . there is a need for a revolution in the basic approach towards systems development. . .".

This comment is a reflection of five traditional problems associated with the development of information systems that is continuing to fuel the need for a CASE approach towards software development.

- Development lead times are excessive.
- Information systems ineffectively support and constrain business activities.
- Systems are unresilient to change.
- High levels of technical skill are required for development.
- Existing systems require substantial maintenance.

Clearly, there is a requirement and need for change, however INPUT's research reveals that this will be a slow evolutionary and not a revolutionary process.

1. The Case Against CASE

Exhibits IV-10 and IV-11 summarise the most frequently mentioned comments made by users with respect to their utilisation of CASE techniques and problems associated with their use.

The key message appearing from the analysis is justification.

Many organisations felt that advanced CASE tools such as IPSEs and workbenches were too sophisticated for their current stage of data processing development.

The IPSE approach is appropriate for projects which involve large dispersed teams working simultaneously, often in different locations. Data sharing is integrated and security is good. In addition, the tool scope is very wide in an open architecture environment. However, there are currently a limited number of international projects for which this solution is applicable.

In addition, it is difficult for organisations to justify the high initial investment costs, especially when productivity is not adequately measured or most commonly is not attempted to be measured.

INPUT's research reveals that many organisations are content to duck the real issue in CASE, that is, changing working practices and methods towards an engineering approach.

Users have addressed the problem by adopting an incremental approach via the automation of code generation and the use of low-cost tools for analysis and design on standalone PCs.

EXHIBIT IV-10

USERS' REASONS FOR NOT USING CASE TOOLS AND TECHNIQUES (Total Western Europe – Synthesis of Most Frequently Mentioned/Valuable Comments)

RANK	COMMENT
1	Too Adavanced for Current Stage of DP Development
2	In-House Tools and Traditional Approach Adequate
3	Prefer to Adopt (Satisfied With) a 4GL Approach
4	Problems with Internal Adaptation to CASE Approach
5	Difficult to Justify Investment
6	Difficulties with Vendor Selection and Product Evaluation
7	Tools are not Good Enough to Meet Specific and Complex Requirements
8	The Tools are not Compatible with our Standards

Clearly, the market for CASE tools and techniques is immature and unstable. Users expressed concern about vendor selection and product evaluation. Surely, this is a reflection of the proliferation of products that have been developed which cannot:

- Support the complete system development life cycle.
- Support unique methodologies, i.e., in-house developed and proprietary.
- Combine architectural and functional integration.
- Interface effectively with databases and 4GLs.
- Adequately address the issue of maintaining existing systems.
- Adequately support a team working environment.

EXHIBIT IV-11

USERS' PERCEPTION OF PROBLEMS ASSOCIATED WITH THE CASE APPROACH (Total Western Europe – Synthesis of Most Frequently Mentioned/Valuable Comments)

RANK	COMMENT	NO. OF MENTIONS
1	Staff Education and Training	16
2	Adaptation of Personnel to Change	8
3	Excessive Rigidity	7
4	Increases Analysis Time	5
5	Increases Development Time - Short Term	4
6	Lack of Standards	3
7=	Cost	2
7=	Maintenance/Upgradeability	2

Many managers have perceived CASE tools as a panacea to the problems of system development. Innovative users have found that this is not the case.

The major inhibitor of market development is the adaptation of personnel to change. Cultural attitudes which tend to regard training as an inappropriate activity are a major barrier to CASE implementation.

The key education problem is not in learning the techniques but in the application of techniques. Consequently, successful CASE tool suppliers are those that can differentiate on quality of service and offer thorough implementation not only in the application of CASE products but also in

the management style and approach towards maintaining productivity and quality in systems development.

A futher major stumbling block for the CASE approach is the implication in terms of change in the traditional roles of analysts and programmers coupled with change in the organisation and structure of data processing development.

The software engineer of the future is a business analyst who spends more time working with end users as consultant and creative problem solver with experience and knowledge of the commercial environment.

Current traditional approaches towards systems development demand highly specialised staff grouped into specialist teams with little interchange of staff between them.

The organisational challenge and opportunity faced by the CASE approach is to share staff and skills between development projects thereby reducing inflexibility, improving internal skills development, improving job satisfaction, and achieving the objective of developing high quality, easily maintainable systems, on-time and within budget.

2. The Case for CASE

Despite our rather negative responses towards the implementation of CASE, INPUT's research reveals that the vast majority of data processing departments are aware of the potential benefits of CASE.

The challenge for CASE product suppliers and other interested parties such as government departments is not to educate the market about potential benefits but to introduce appropriate approaches towards effective implementation.

Exhibit IV-12 summarises the most frequently mentioned comments made by users with respect to the benefits associated with a CASE approach towards software development.

It is interesting to note that users considered the most important benefit of CASE to be improvement in the quality of software. INPUT suspects that users' view of quality is somewhat limited, i.e., does not include measurement of the efficiency of systems in terms of meeting objectives as part of a corporate IT strategy.

EXHIBIT IV-12

USERS' PERCEPTION OF BENEFITS ASSOCIATED WITH A CASE APPROACH (Total Western Europe – Synthesis of Most Frequently Mentioned Comments)

RANK	COMMENT	NO. OF * MENTIONS
1	Improved Quality	30
2	Easier Maintenance	28
3	Faster Development	20
4	Saving in Time	19
5	Standardisation of Procedures	18
6	Improved Productivity/Efficiency	13
7	Improved Documentation	9
8	Better End-User Relations	7
9	Simplification of Operation	6
10	Reduced Costs	5
11	Improved Security	4
12	Improved Management Control	4
13	Flexibility on Interchange of Personnel	3

* Multiple Responses Allowed

However, the research reveals that users are already perceiving potential benefits other than productivity - a very positive sign for the industry.

For the benefit of organisations seeking to evaluate the quality of their software INPUT has prepared a list (which is not exhaustive) of key factors to be addressed. Our list of quality factors is given as Appendix E of the report.

The research reveals that worries are increasing in the computer industry not only about the cost of developing software but also about whether

systems which affect a company's commercial performance and human safety are up to a high enough standard.

No matter how advanced the methods of computer programming they will not yield software of the highest quality unless the basic rules of quality assurance management and configuration management are observed.

INPUT's research revealed seven basic principles of good management practice for quality assurance:

- Commitment of management.
- Comprehensive set of standards, methods, and tools.
- Independence of the quality assurance organisation.
- Quality assurance involvement for the total project life cycle.
- A defined level of quality across the company.
- Defined quality requirement for each project.
- Continuous monitoring and review of the quality system.

INPUT's research revealed that the second most important benefit associated with the implementation of CASE is easier maintenance. Maintenance problems are eased via the standardisation of development procedures and major improvements in documentation. Developers can then focus their attention on the development of new systems rather than patching up old programmes.

Improved documentation and standardisation also facilitates flexibility in terms of interchange of personnel between different types of projects.

Consequently, although many organisations have pointed to the resistance of software developers to CASE approaches because of the deskilling aspects, the main benefit is improved job satisfaction. In-house developers can gain experience without becoming contractors and, above all, they can improve their image within the corporate organisation.

A further benefit of CASE is that it facilitates development by non-IS specialists. Although the major initial impact is in the development centre, CASE facilitates decentralized development by end-users as well as IS specialists.

INPUT foresees a future wherein the data processing department has two roles. Firstly, it will act as data manager maintaining the integrity of

company data. Secondly, it will act as manager for the company's core systems which provide an IS infrastructure for the company as a whole, i.e., operational systems.

End-users with access to user-friendly CASE products and 4GLs would be responsible for the organisations' peripheral systems, i.e., systems for planning and control.

INPUT estimates that by the mid-1990s only 45% of systems could be classified as core operational systems, 40% will be designed for management control, and 15% for planning purposes. These peripherals systems will increasingly be developed by end-users themselves or by IS specialists employed by external service companies under their supervision.

This solution to the problems of software development offers an alternative approach to the management of demand. However, a major benefit of CASE is in the improvement of productivity in the development centre. INPUT's research revealed that the vast majority of organisations did not measure productivity. Even those using CASE products were not able to estimate how much time (and money) they had saved via their implementation. Indeed, several respondents pointed to the fact that, in the short term, productivity had actually decreased owing to high implementation costs. This finding is a reflection of the fact that the productivity and quality benefits of CASE are project "tail-end" rather than project "front-end".

Exhibit IV-13 provides a summary of users' estimates of time saved on projects by using CASE tools and techniques in comparison with manual methods.

Our research indicates that the average percentage time saved on projects using structured methodologies was 18%. The average percentage time saved using CASE tools (including self-developed tools) was 35%. However, it should be noted that the sample sizes are small.

It is interesting to note that CASE product vendors are no longer emphasing the time-saving benefits from implementation as part of their promotional tactics. Many suppliers have claimed that 40% productivity gains are guaranteed. However, this assumes two factors:

- Firstly, that the products are used effectively by trained personnel.
- Secondly, that the productivity gain is measurable and indeed measured over the whole life cycle of a project.

EXHIBIT IV-13

USERS' ESTIMATES OF TIME SAVED ON PROJECTS BY USING CASE TOOLS AND TECHNIQUES (Total Western Europe)

TECHNIQUE/ TOOL	NO. OF RESPONDENTS	AVERAGE PERCENTAGE TIME SAVED ON PROJECTS
Structured Methodology	18	18
Tools in General #	21	35

Including Self-Developed

There are only a limited number of systems which have been developed using CASE products which are currently at an advanced stage in terms of operation lifespan.

H

The Implementation of CASE

When one thoroughly considers the implications of CASE implementation for end users it is not surprising to find that many organisations have adopted a step-by-step approach to implementation.

Exhibit IV-15 illustrates why organisations have given CASE a priority. The reality in systems development using traditional (but well-understood) methods means that timescales and budgets overrun and user requirements are not satisfied. The diagrams show a simplification of the phases of the software development life cycle illustrated in Exhibit IV-14.

Exhibit IV-16 illustrates not only the benefits but also the reality of implementing CASE solutions. There is a significant learning curve in terms of the realisation of the CASE approach towards systems development.

Advanced CASE products such as IPSEs and full life cycle workbenches only begin to pay for themselves in terms of productivity and quality gains in the medium to long term after their utilisation in a number of projects.

EXHIBIT IV-14

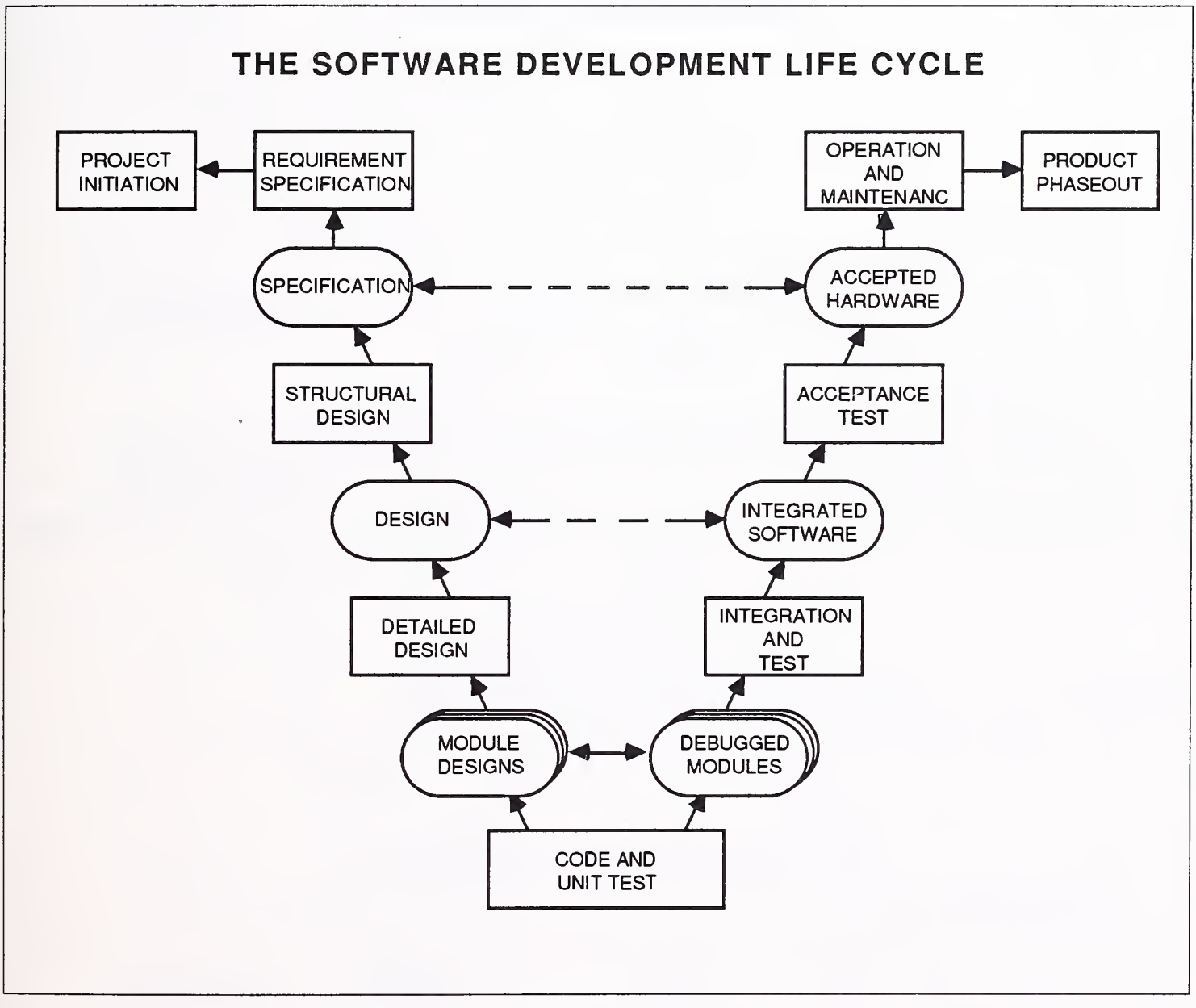
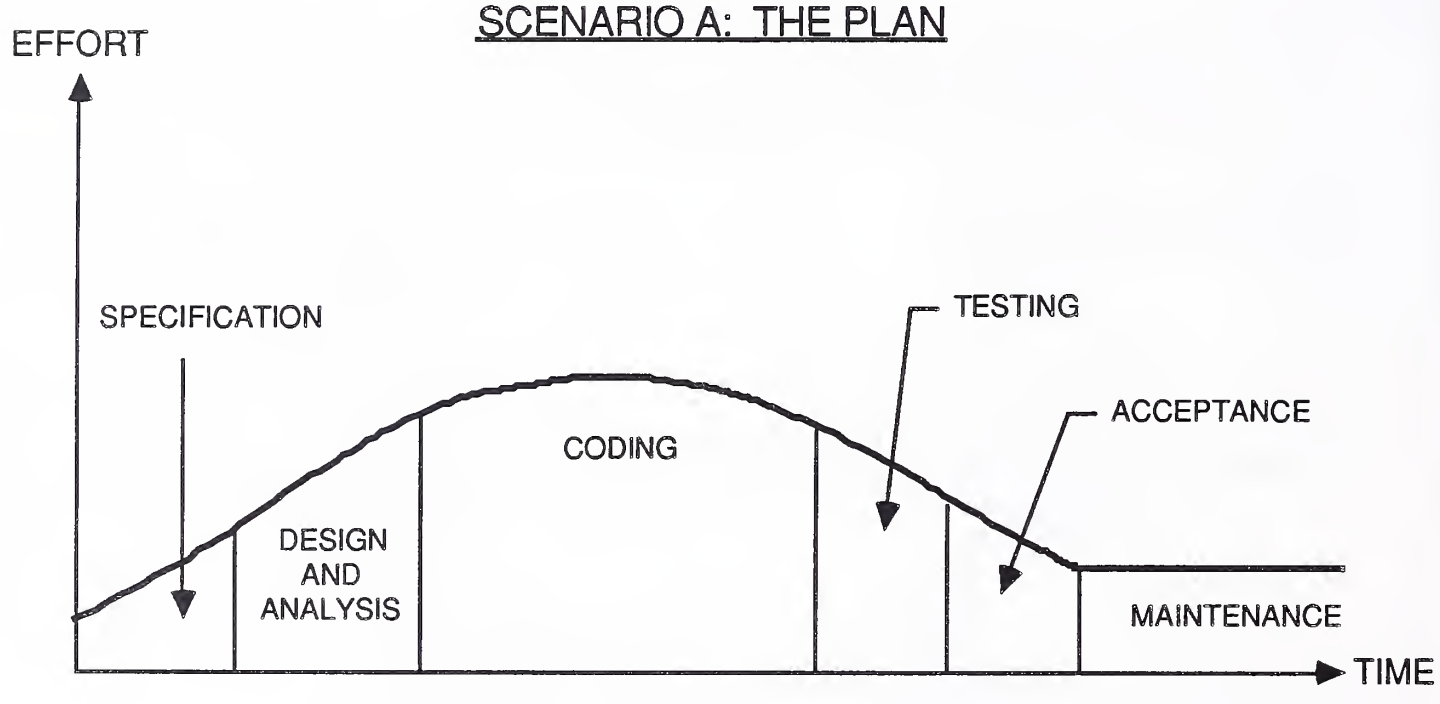


EXHIBIT IV-15

TRADITIONAL APPROACHES TOWARDS SOFTWARE DEVELOPMENT— THE PLAN AND THE REALITY



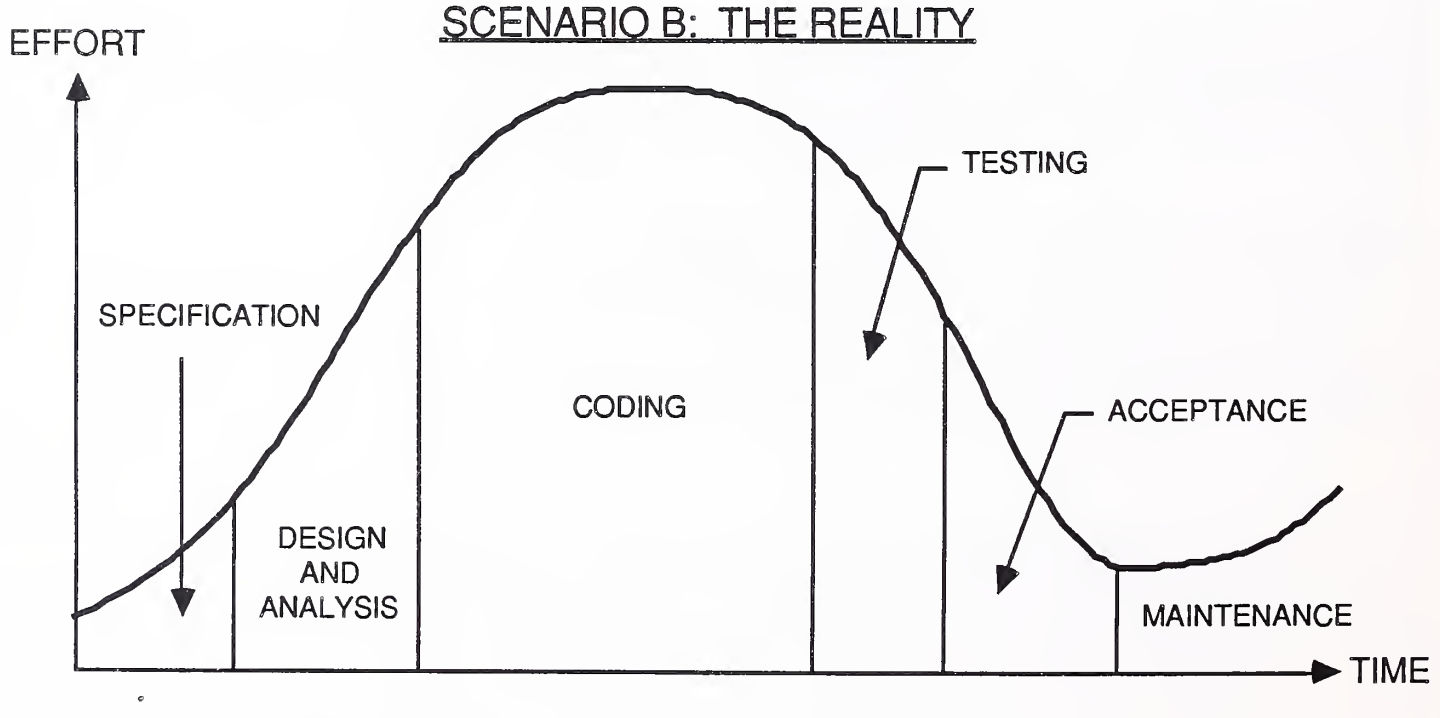
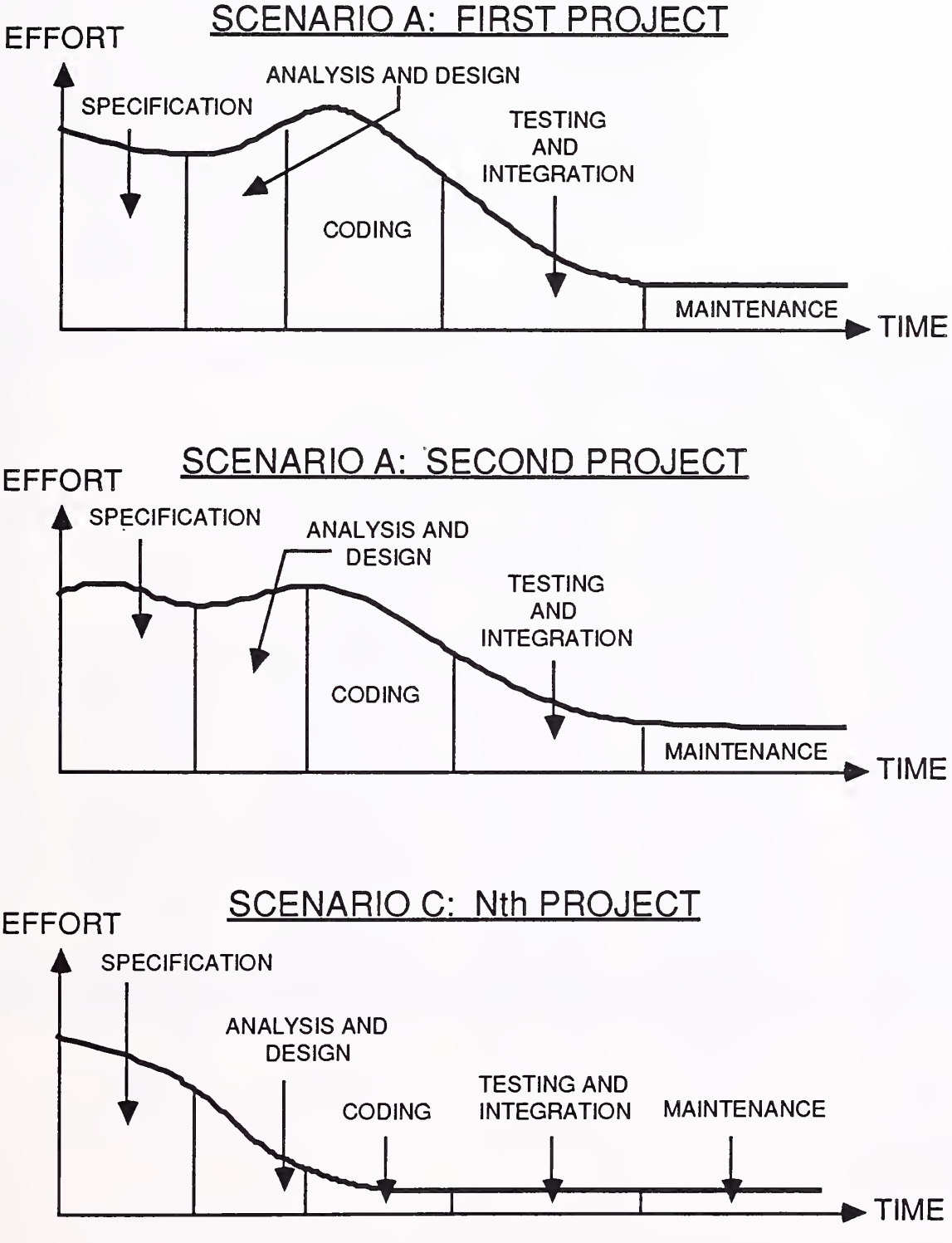


EXHIBIT IV-16

THE IMPACT OF CASE
A LEARNING CURVE



The challenge for vendors of advanced CASE tools and services is to:

- Find champions who are already well-advanced in their evaluation of CASE and probably already use basic CASE tools and techniques.
- Offer a full implementation service in terms of training and consultancy. Training is important to insure effective use of products and techniques. Consultancy is important because of the strategic nature of the decision. Changes in working practices need to be evaluated and reviewed on a project-by-project basis.This means change not only in terms of how software teams operate but, more importantly, change in the management disciplines that are needed to ensure effective implementation.

I

Incremental (Non-CASE) Approaches towards Systems Development Productivity—Languages, Programme Development Tools, and DBMS

INPUT's research would indicate that in recent years the majority of data processing departments in Europe have adopted incremental solutions to the problems of:

- The applications backlog.
- The challenge of connectivity and integration of distributed processing.
- The management of the growth of end-user computing.

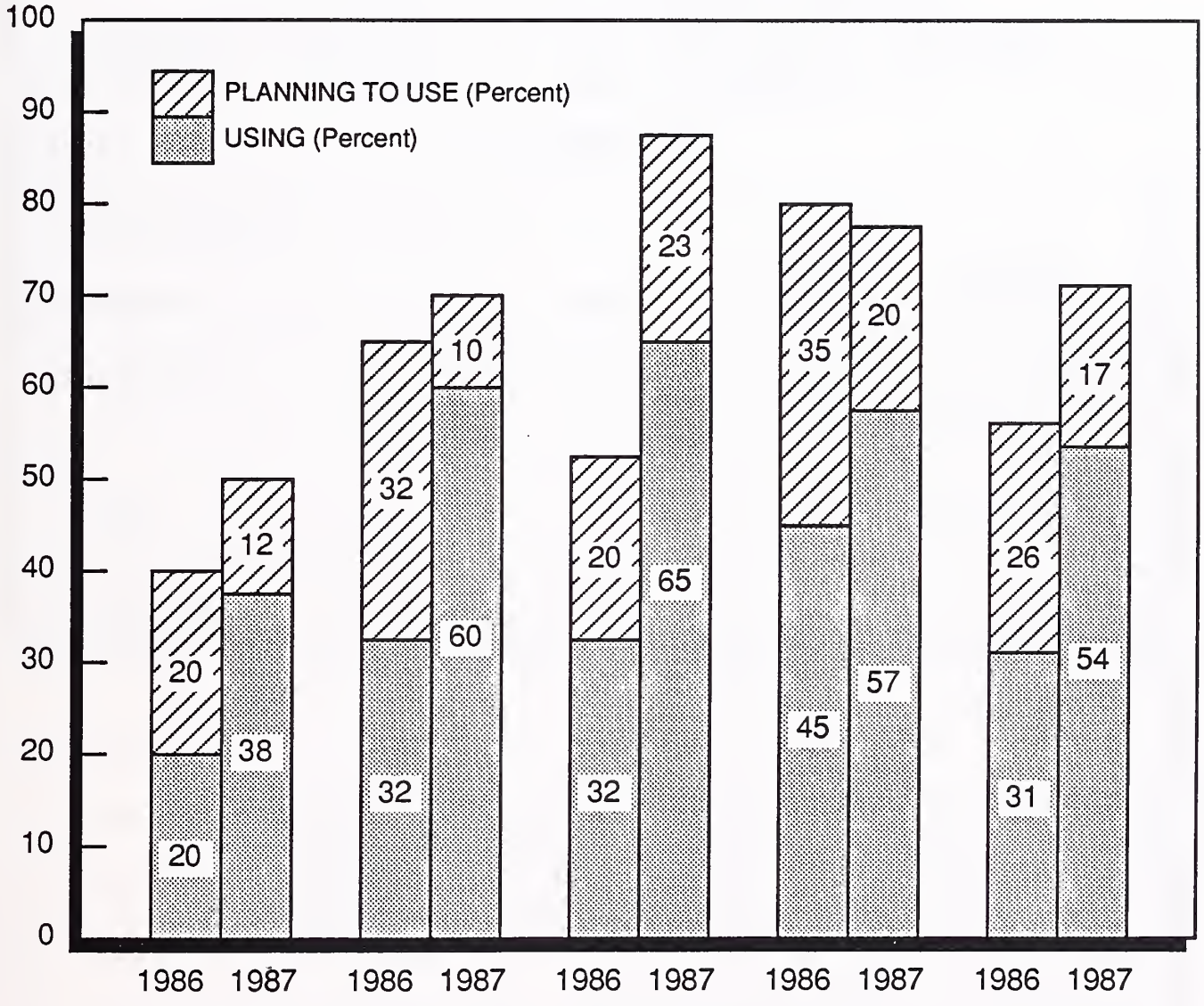
With reference to Exhibit IV-17 it is possible to observe rapid growth in the adoption of fourth generation languages across Western Europe. In 1986 they were used by 31% of establishments and in 1987 by 54%.

For the development of systems of medium size and medium complexity, the 4GL approach offers the data processing community the ability to combine the basic systems skills it has learned over the years with go-faster tools for coding and prototyping.

In addtion, 4GLs can potentially meet requirements for end-user computing given the development of less hostile human computer interfaces (HCI's).

EXHIBIT IV-17

FOURTH GENERATION LANGUAGES MARKET PENETRATION AND OPPORTUNITIES Comparison Between 1986 and 1987 (Research Results Analysed by Country Market)



SAMPLE SIZES					
1986	30	25	25	20	100
1987	40	30	40	30	140

With reference to Exhibits IV-18 and IV-19 INPUT's 1987 research revealed opportunities for 4GL product vendors in process manufacturing, insurance and health care (i.e., pharmaceuticals and private sector health). In terms of country markets there are opportunities to be exploited in the United Kingdom, Italy, and Holland.

EXHIBIT IV-18

FOURTH GENERATION LANGUAGES MARKET PENETRATION AND OPPORTUNITIES (Analysis by Industry Sector) 1987 Research

INDUSTRY SECTOR	SAMPLE SIZE	PERCENTAGE SPLIT		
		USE	PLAN TO USE	DO NOT USE
Process Manufacturing	25	32	[24]	44
Discrete Manufacturing	28	54	14	32
Retail Distribution	19	53	16	31
Finance and Banking	38	58	13	29
Insurance	18	39	[28]	33
Health Care	14	57	[36]	07
Government	26	50	15	35
Transport	22	59	23	18
Utilities	20	80	15	05
All Sectors	210	53	19	28

EXHIBIT IV-19

FOURTH GENERATION LANGUAGES MARKET PENETRATION AND OPPORTUNITIES (Analysed by Country Market) 1987 Research

COUNTRY	SAMPLE SIZE	PERCENTAGE SPLIT		
		USE	PLAN TO USE	DO NOT USE
West Germany	40	38	12	50
United Kingdom	40	65	[23]	12
France	30	60	10	30
Scandinavia	40	58	15	27
Italy	30	57	[20]	23
Benelux	30	44	[23]	33
All Countries	210	53	17	30

However, when one is examining opportunities for 4GLs it is important to define exactly what is meant by the generic term 4GL. There are a huge range of software products that are available on the market which have different design objectives, address different needs, and yet are all called 4GLs.

Essentially, there are three types of 4GLs:

- Code generators, e.g., TELON.
- Application development tools, e.g., Ideal, Natural, or Mantis.
- Information centre products, e.g., Focus, Ramis, or Nomad.

Essentially, there are four major trends impacting non-CASE application development tools summarised in Exhibit IV-20 as follows:

- The evolution towards CASE.
- The convergence between 4GLs and RDBMs/DDBMs.
- The trend towards IBM standards.
- Market shake out.

The trend towards distributed processing and demand for end-user computing has led to a link between RDBMs and 4GLs. All the major vendors of DBMs now offer compatible 4GLs, i.e., Software AG, ADR, IBM, Cullinet and CINCOM.

In the distributed processing environment 4GLs are now positioned as two-way interfaces between different systems and machines, effectively becoming a programming common denominator for prototyping new applications and filtering existing data.

EXHIBIT IV-20

TRENDS IN NON-CASE APPLICATIONS DEVELOPMENT TOOLS

- Convergence 4GLs and RDBMs/DDBMs
- IBM Standards
  - Controlling the Environment
  - SAA
  - SQL DBMs Standard
- Evolution Towards CASE
  - Life Cycle Coverage Increasing
  - Tool Integration Increasing
- Market Shake-Out

In the integrated environment they are having a significant impact on programmer productivity. With reference to Exhibit IV-21, INPUT's research revealed that among the establishments using 4GLs, 60% of companies could point to measurable productivity gains.

Across Western Europe the average percentage time saved on projects using a 4GL in comparison with a 3GL approach was 39%. It is interesting to note major benefits from a 4GL approach in Italy where data processing developments are smaller than in other European countries and the DP organisation less mature.

IBM is setting the standards in terms of the application development tool environment. IBM's DB2 and SQL have been chosen as lead products, offering users the ability to interface with several proprietary databases and update files.

EXHIBIT IV-21

USERS' ESTIMATES OF TIME SAVED ON PROJECTS BY USING FOURTH GENERATION LANGUAGES (Analysed by Country Market—Comparison between 4GL and Traditional 3GL Approach)

COUNTRY MARKET	SAMPLE SIZE	NO. OF RESPONDENTS	TOTAL SAMPLE USING 4GL's (Percent)	AVERAGE TIME SAVED ON PROJECTS (Percent)
West Germany	40	10	38	36
United Kingdom	40	10	65	34
France	30	12	60	43
Italy	30	14	57	[47]
Scandinavia	40	11	36	[58]
Benelux	30	9	43	37
All Countries	210	66	53	39

With reference to Exhibit IV-22 it is possible to observe a trend towards IBM solutions in terms of users' purchasing intentions.

INPUT predicts that SQL will become the standard relational language by 1989.

IBM standards are also controlling the environment following announcement of SAA. SAA (System Application Architecture) facilities a standard user interface and application development independant of databases and user interfaces. Essentially, this means application software development portability.

The third major trend is the evolution towards CASE, i.e., a total environment of RDBMs, data dictionary, 4GLs and graphical programming languages (i.e., integrated toolkits).

EXHIBIT IV-22

**USERS' PLANS FOR INSTALLING 4GL PRODUCTS*
THE EUROPEAN "BIG SIX" COMPARISON BETWEEN
1986 AND 1987**

1986 RESEARCH			1987 RESEARCH		
RANK	PRODUCT (Product Stable)	NO. OF MENTIONS	RANK	PRODUCT (Product Stable)	NO. OF MENTIONS
1	Ideal	3	1	IBM - CSP	4
2=	Natural	2	2	IBM - SQL	3
2=	Mapper	2	3=	Oracle	1
4=	IBM-Application Master	1	3=	Focus	1
4=	UFO	1	3=	Nomad	1
4=	H.P. Transact	1	3=	Link 2	1

* Summary of Users' Responses for 4GL Products they plan to install.

Traditional 4GLs are being linked with graphics front ends, code generators, data dictionaries and integrated toolsets.

Vendors of integrated 4GL/RDBMs products are slowly emerging as CASE companies respond to customer demand for full development and maintenance tools, such as Information Builders' plans for a full life cycle applications generator, and Cortex launching of Corvision, a graphics front end, to the Applicaiton Factory 4GL, fronting DEC databases. Mature products such as Pansophics' application generator Telon are being re-cycled via increasing life cycle coverage, i.e., the link with Exelerator, a design and analysis tool.

There is a major opportunity for the independent software houses to develop CASE products. IBM's DB2 and SQL do not have an adequate data dictionary or integrated set of tools for comprehensive project support. James Martin Associates with Information Engineering Facility (IEF) and Arthur Andersen with Foundation are already attacking this gap in the IBM marketplace.

The fourth major trend anticipated by INPUT is market shake out. The last couple of years have seen a proliferation of 4GL products launched onto the European marketplace by hardware manufacturers, domestic independents, and the traditional multinational U.S. independents.

INPUT's research revealed that the number of products available on the market had tripled between 1986 and 1987. Market penetration of 4GL products in terms of market leadership is summarised in Exhibit IV-23.

In the U.K. alone, there are over 100 Applications Development Tools (ADTs) available commercially. Users are confused and find increasing difficulty with vendor selection and evaluation.

This poses a threat and an opportunity for ADT vendors. The opportunity is in consultancy to aid users through the maze of product evaluation and implementation. The threat is market shake-out as users increasingly demand comprehensive project and programming support environments.

The vendors who will survive are those that can offer full life cycle support, maintenance support, project management support, prototyping and automatic code generation support linked with automated methodology and documentation support for design and analysis.

EXHIBIT IV-23

MARKET PENETRATION OF 4GL PRODUCTS THE EUROPEAN "BIG SIX"* COMPARISON BETWEEN 1986 AND 1987 RESEARCH

1986 RESEARCH			1987 RESEARCH		
RANK	PRODUCT (Product Stable)	NO. OF MENTIONS	RANK	PRODUCT (Product Stable)	NO. OF MENTIONS
1	Focus	9	1	Focus	9
2=	SQL	3	2	Natural	8
2=	Mantis	3	3	Ramis	6
2=	Natural	3	4	Ideal	5
5=	Ideal	2	5=	Mantis	3
5=	Mapper	2	5=	ADS	3

N.B. Sample Size = 100 in 1986 and 210 in 1987

NOTE: Total Number of Different 4GL Products mentioned in 1986 was 19.
Total Number of Different 4GL Products mentioned in 1987 was 72!

* Summary of Users' Responses for 4GL Products they have installed

J

Users' Ratings of Potential Approaches towards Improving Data Processing Development Productivity

Exhibits IV-24 and IV-25 illustrate users' ratings of the importance of a number of potential approaches towards improving data processing development productivity.

Although nearly all the potential approaches were rated as being important, the three key approaches to emerge were increased commitment to quality, the recruitment and retention of quality staff, and increased involvement of end-users.

EXHIBIT IV-24

USERS' RATINGS OF IMPORTANCE OF POTENTIAL APPROACHES TOWARDS IMPROVING DATA PROCESSING DEVELOPMENT PRODUCTIVITY (Total Western Europe)

NUMBER	POTENTIAL APPROACH	AVERAGE* RATING
1	Development Centre	6.1
	Automation	
2	- Design Methodologies	6.7
3	- Training	6.8
4	- Project Management	7.0
	Training	
5	- Project Management Skills	7.4
6	- Business/Administration Skills	7.1
7	- New Techniques, Tools, Methodologies	7.3
8	Increased Investment in Hardware	6.6
9	Upgrade of Systems Software (e.g. UNIX)	5.8
10	Increased End-User Involvement	[8.4]
11	Forming Joint Project Teams/Work Units with End-Users	8.1
12	Increasing Awarenes of your Company's Business Problems	8.1
13	Increased Investment in Development Tools	7.5
14	Increased Commitment to Quality	[9.1]
15	Recruiting & Retaining Quality Staff	[8.2]

* Ratings were expressed on a scale of 0 to 10, where 0 = Irrelevant and 10 = Critical.

N.B. Average Standard Error = 0.1

EXHIBIT IV-25

USERS' RATINGS OF IMPORTANCE OF POTENTIAL APPROACHES TOWARDS IMPROVING DATA PROCESSING DEVELOPMENT PRODUCTIVITY (Analysed by Country)

COUNTRY MARKET	AVERAGE RATING BY APPROACH*															SAMP. SIZE
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
West Germany	4.9	6.3	6.4	6.6	7.1	6.6	7.5	5.0	5.2	[8.6]	8.0	8.5	7.9	[9.5]	[8.6]	40
UK	5.1	7.0	7.8	8.2	[9.0]	8.7	8.6	8.1	5.2	[9.4]	9.2	8.1	8.8	[9.8]	8.9	40
France	7.7	6.6	6.5	6.5	7.5	6.7	6.6	[7.8]	7.1	[8.1]	[7.8]	7.1	7.4	[8.4]	7.0	30
Italy	7.3	7.2	7.2	7.1	7.4	7.3	7.2	7.6	7.3	[8.0]	[8.0]	7.9	6.6	[8.2]	6.8	30
Scandinavia	5.6	6.4	6.2	6.3	6.5	6.3	6.8	5.2	4.7	7.7	7.5	[8.7]	7.3	[9.2]	[8.7]	40
Benelux	6.8	6.7	6.8	6.9	6.9	6.7	7.0	6.7	5.6	[8.4]	8.1	7.9	6.9	[8.9]	[8.7]	30
Total Western Europe	6.1	6.7	6.8	7.0	7.4	7.1	7.3	6.6	5.8	[8.4]	8.1	8.1	7.5	[91.]	[8.2]	210

* Ratings were expressed on a scale of 1 to 10, where 0 = Irrelevant and 10 = Critical

N.B. Average Standard Error = 0.1

NOTE: The fifteen potential approaches are the same as those listed in Exhibit IV-24.

The CASE approach offers a solution to all three of these objectives. Quality people enjoy working in quality environments where there is a professional and effective approach towards software development.

In addition, with a CASE approach the user is king. Picture programming and other CASE techniques allow close involvement with users throughout the development cycle.

1. The Key to Success - A Professional Environment

A true science of software engineering can never be formulated with engineering techniques, tools and methodologies alone. Tools and tech-

niques are not a panacea to the problems of software development; they help in the understanding of problems but are not a replacement for creative solutions. The key for success in managing software development is not to stifle creativity by rigorous process control and complex methodologies.

A professional software development environment is one in which problems can be resolved quickly and communications are rapid and effective between developers and users.

It is an environment where:

- Everyone must be treated (and must behave) like a professional.
- Everyone must be personally committed to well-stated and reviewed objectives, i.e., project baselines.
- Common sense must prevail.
- Quality is expected from everyone.

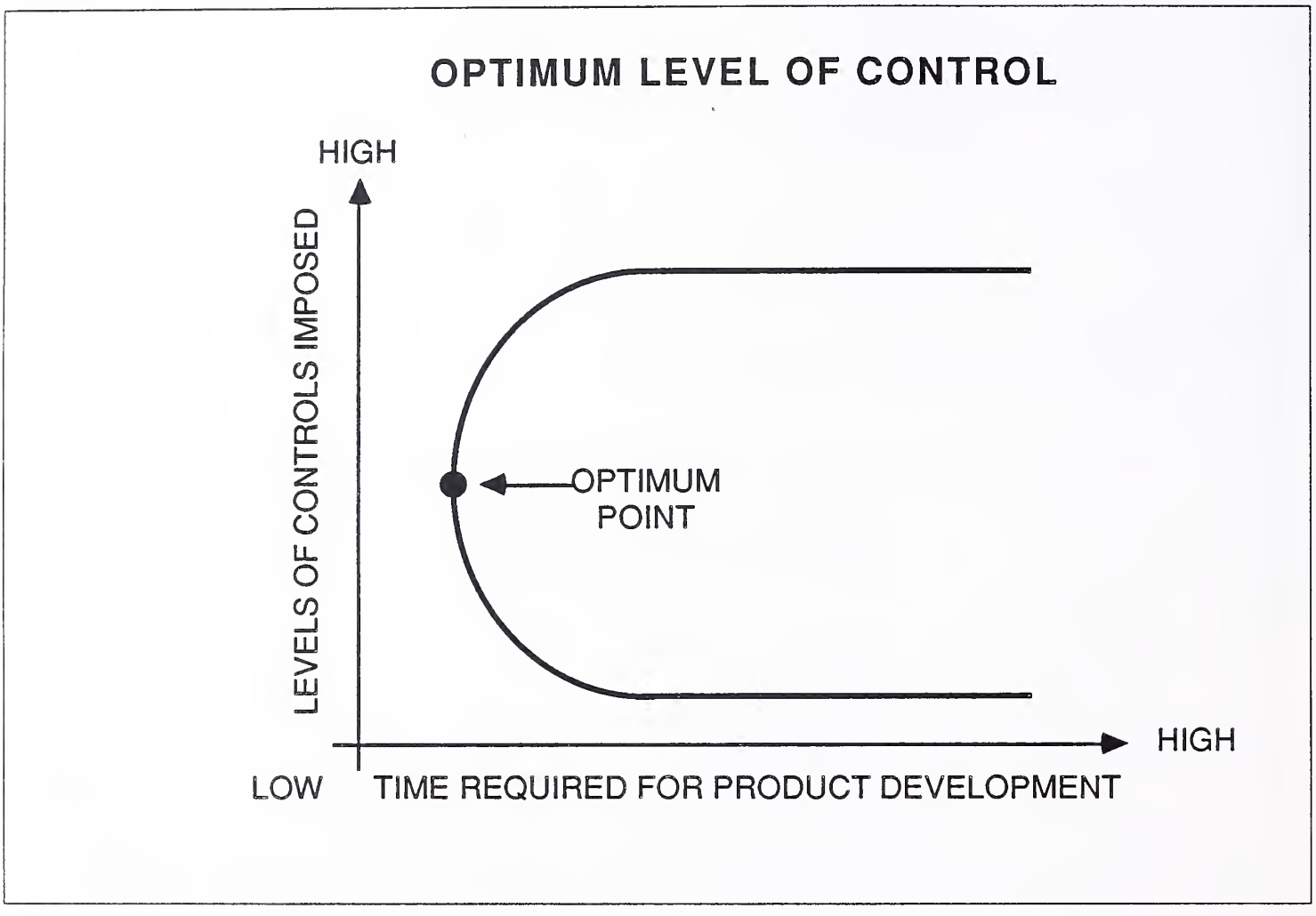
Too many organisations project manage software engineers' activity to the level of micro modules and micro milestones—not a motivating action. Software quality standards should not be rigid and should be a living, frequently updated set of documents.

With reference to Exhibit IV-26 there is an optimum level of control. Too few controls produce an undisciplined environment in which it is impossible to make progress. Too many controls, however, require a large investment in time and manpower in the controls process and reduce motivation.

The shortest possible development time results from an ideal level of controls, a level found only in professional development environments.

The key to true gains in productivity is the elevation in levels of professionalism in software development coupled with a scientific engineering approach.

EXHIBIT IV-26



K

Users' Ratings of Potential Approaches towards Supporting End-User Development Productivity

Exhibits IV-27 and IV-28 illustrate users' ratings of the importance of a number of potential approaches towards supporting end-user development productivity.

It is interesting to note that increased connectivity, i.e., micro-to-mainframe links and LANs were seen as a key priority. Exhibit IV-29 illustrates a user view of connectivity. Productivity and efficiency benefits accrue from linking up machines of different manufacturers and different databases into a coherent integrated corporate information network.

Viewing end-users as clients was also seen as a key objective, i.e., working closely with end users in project teams to resolve business issues and problems requiring IS solutions.

There are some interesting variations between the country markets.

EXHIBIT IV-27

USERS' RATINGS OF IMPORTANCE OF POTENTIAL APPROACHES SUPPORTING END-USER DEVELOPMENT PRODUCTIVITY (Total Western Europe)

NUMBER	POTENTIAL APPROACH	AVERAGE* RATING
1	Information Centre	6.5
	Increased Connectivity	
2	- Micro-to-Mainframe Links	[7.8]
3	- Local Area Networks	7.4
4	Prototyping	7.3
5	Fourth Generation Languages	7.6
	Training	
6	- Systems Analysis Skills	6.6
7	- Product Knowledge	7.4
8	Viewing End-Users as Clients	[8.0]

* Ratings were expressed on a scale of 0 to 10, where 0 = Irrelevant and 10 = Critical.

N.B. Average Standard Error = 0.1

West German respondents placed high emphasis on training end-users in the use of products and taking a more positive attitude towards supporting end-users as clients. The establishment of information centers was rated as a low priority, a reflection of the highly centralised environment of German data processing.

U.K. respondents rated highly the importance of prototyping and the use of 4GLs, a reflection of a British tendency to focus on the technology rather than the process and the people.

EXHIBIT IV-28

USERS' RATINGS OF POTENTIAL APPROACHES TOWARDS SUPPORTING END-USER DEVELOPMENT PRODUCTIVITY (Analysed by Country Market)

COUNTRY MARKET	AVERAGE RATING BY APPROACH *								SAMPLE SIZE
	1	2	3	4	5	6	7	8	
West Germany	5.4	6.8	6.8	6.5	6.7	7.3	[7.9]	[8.8]	40
United Kingdom	7.8	8.3	6.0	[8.9]	[9.5]	6.6	7.1	7.4	40
France	7.3	7.6	[9.5]	6.9	6.9	5.6	7.6	[7.9]	30
Italy	7.5	7.7	7.2	6.6	6.8	6.2	[8.0]	[8.1]	30
Scandinavia	5.2	7.1	6.7	7.0	[7.8]	6.9	7.2	[8.0]	40
Benelux	6.4	[9.6]	[8.5]	7.7	7.5	6.4	6.8	7.9	30
Total Western Europe	6.5	7.8	7.4	7.3	7.6	6.6	7.4	8.0	210

* Ratings were expressed on a scale of 1 to 10, where 1 = Irrelevant and 10 = Critical.

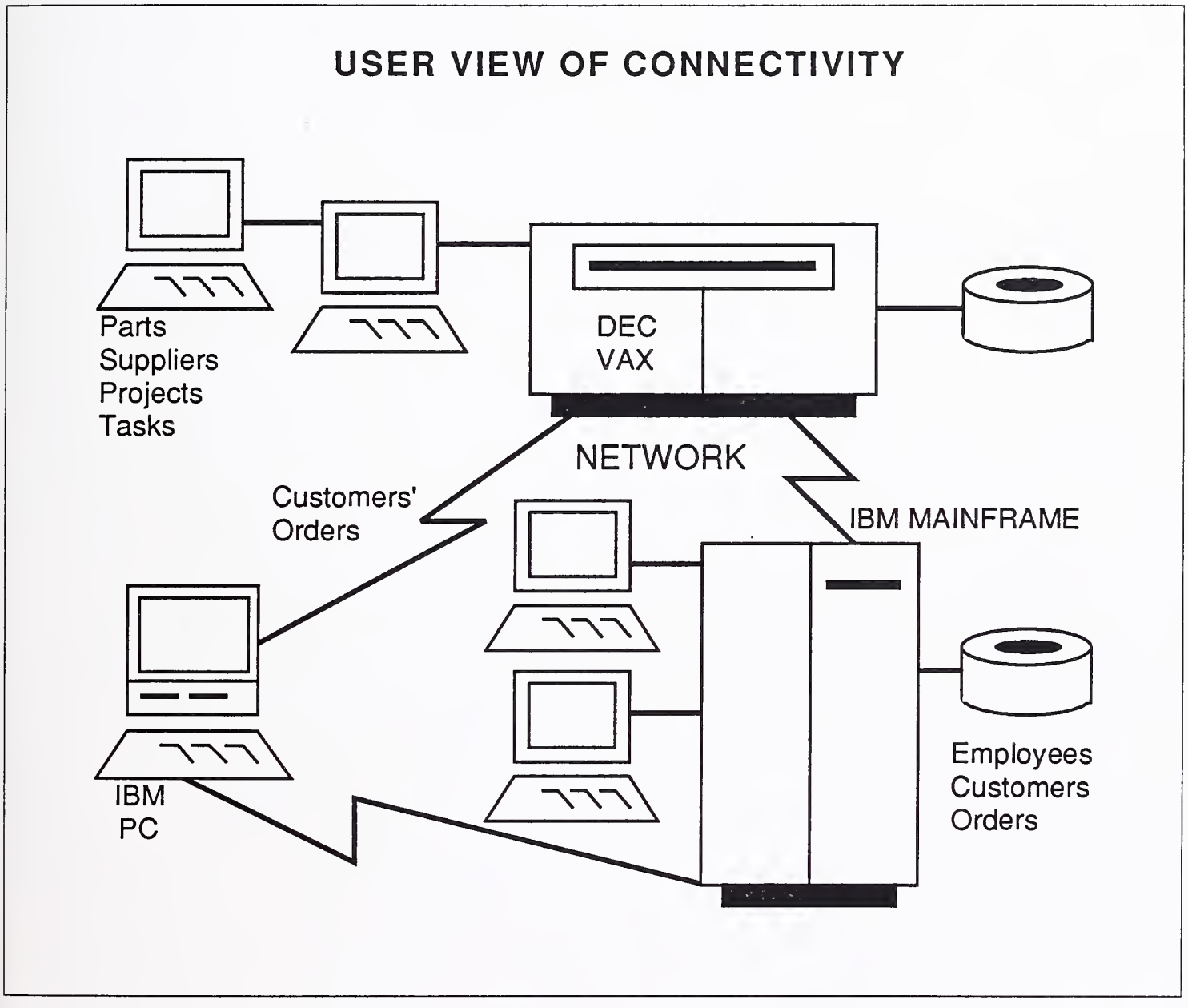
N.B. Average Standard Error = 0.1.

NOTE: The eight potential approaches are the same as those listed in Exhibit IV-25.

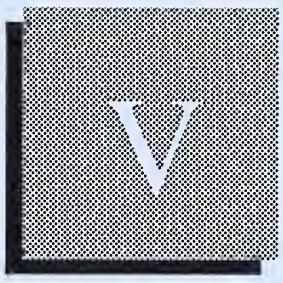
The Italians, like the Germans, placed high emphasis on working closely with end-users and pointed to the need for training.

The Scandinavians, like the British, are content with 4GL approaches to the problem of end-user computing.

EXHIBIT IV-29



The next chapter reviews the competitive environment with respect to CASE products and services, including analysis of some significant players and leading products in the commercial market.



Competitive Environment

Competitive Environment

A Market Overview

1. CASE Product Evolution

CASE, as previously defined, is an umbrella term used to describe a total development environment of RDBMs, a data dictionary, 4GLs and two-dimensional graphical programming languages which allow a software engineer to build a graphical image of an application and then automatically generate the code to produce it.

Currently, the majority of CASE products do not provide this fully integrated functionality that addresses all aspects of software engineering.

Essentially, CASE systems fit in three basic groups:

- Schematic design products, e.g., information engineering workbench (IEW) and Excelerator.
- Code generators, e.g., Telon, Transform, or APS.
- Complete Systems, e.g., Maestro, ISTAR, or Foundation.

a. Schematic Design Products

These products are also known as analyst workbenches or toolkits. They were initially developed to assist the analyst in producing specifications and documentation.

With reference to Exhibit V-1 which illustrates the evolution of CASE products it is possible to discern that these front-end tools represents the earliest available CASE products.

EXHIBIT V-1

CASE PRODUCT EVOLUTION

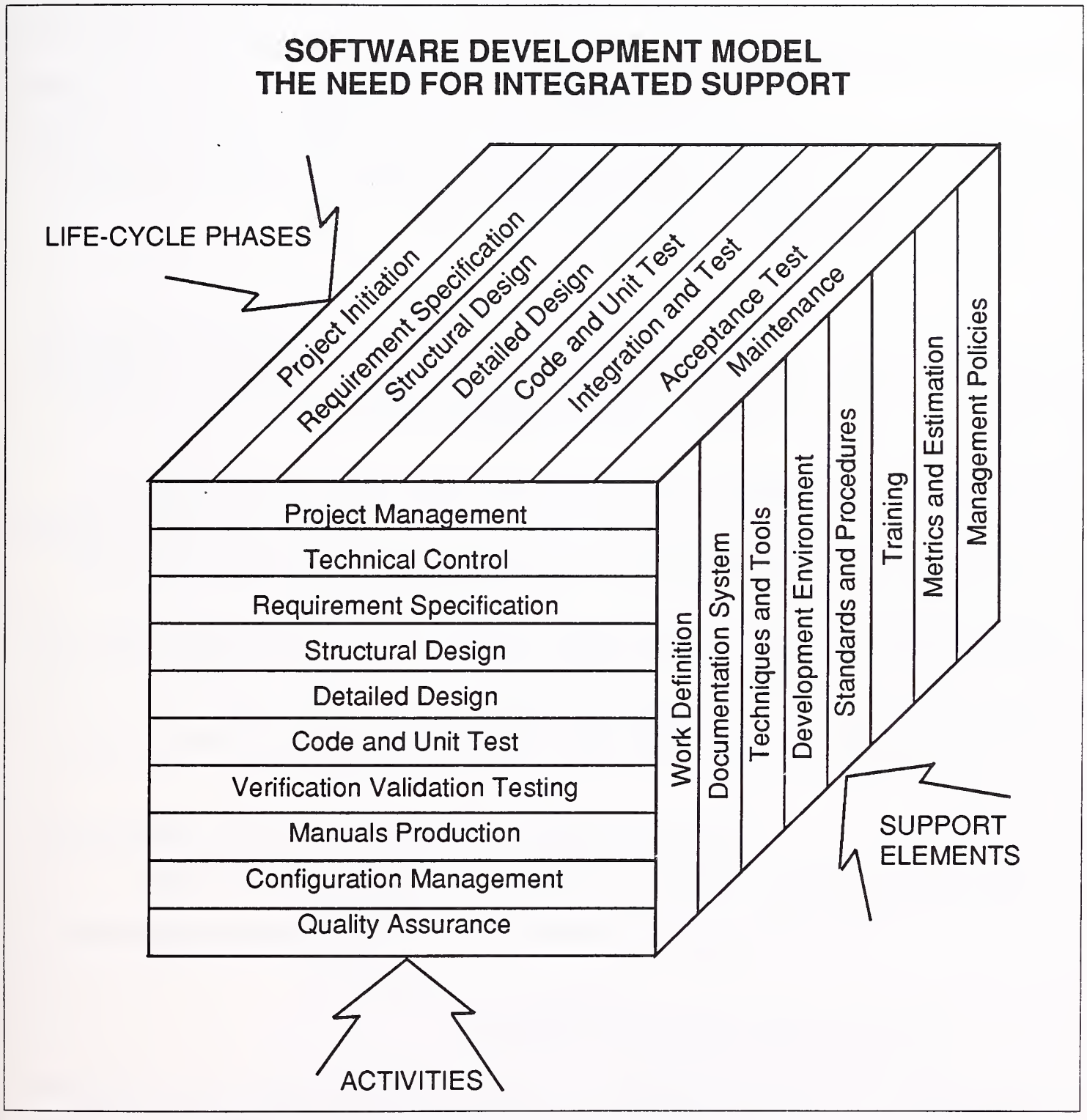
EARLY 1980's	MID 1980's	LATE 1980's	EARLY 1990's
Computer-Aided Documentation Tools	Automatic Design Analysis and Checking	Automatic Generation from Design Specifications	Intelligent Methodology Drives
Computer-Aided Diagramming Tools	Automated System Information Repository	Linking Design Automation and Program Automation	Habitable User Interface
Analysis and Design Tools		Linking Graphics Interfaces	Reusability Development Methodology

However, these early products have evolved to include diagramming and graphics capabilities, tools to support prototyping, a design dictionary, and analysis capabilties for error checking. In addition, they are moving towards offering full lifecycle support, i.e., supporting all activities in all phases of the life cycle as illustrated in Exhibit V-2.

Early CASE products face users with the problem of connectivity. Innovative users have purchased front-end systems for analysis and design which are not able to link directly with systems for code generation.

Vendors are responding to the need for a fully-integrated development environment which INPUT has chosen to call CADME (Computer Aided Development and Maintenance Environment). For example, IEW is now being linked to Knowledgeware's GAMMA Code Generator and Excelerator is linked to Pansophic's Telon. In the U.S.A., Nastec Corporation's CASE 2000 Designaid product is being linked to Telon and Xerox's Ventura system for electronic publishing features.

EXHIBIT V-2



b. Code Generators

Code generators aid productivity by automatically converting software logic designs into actual programme code. Products such as DEC's VAX COBOL Generator, APS from Sage Inc. and Transform from Scottsdale are also evolving to meet user demands for an integrated support environ-

ment. Code generators have evolved to provide for automatic documentation, analysis capabilities for error checking and include a data dictionary or central repository. For example, CGI's PACBASE has evolved to include a COBOL generator, schematic design and analysis tools, testing tools, a data dictionary and graphics front ends. Consequently, CASE products are evolving and converging towards complete systems as the need for structured techniques and systems development complexity grows within the software industry.

c. Complete Case Systems

New products appearing on the CASE marketplace offer full lifecycle coverage and address the development of commercial applications, for example, Computer Associates Programmers' Workbench or Arthur Andersen's Foundation products.

Many users have decided that CASE is not a bona fide strategy for Systems Development until complete systems are available. It is difficult for users to evaluate products and quite understandably they cannot justify the ad hoc investment in individual tools.

It is possible to isolate five criteria that categorise a product which provides a full CASE system:

- Front-end software that can pictorially represent the applications cycle.
- Prototyping software which makes use of graphics capabilities, i.e., automatic screen generation, data, and system modeling.
- A central, commonly accessable data dictionary and project data base.
- An efficient code generator.
- A methodology or engineering approach which needs to be flexible in the number of logical design approaches supported, e.g., Jackson, Yourdon, Chen, SSADM, etc.

With reference to Exhibit V-1, INPUT foresees that these complete systems will become commonplace as the CASE products of the 1990s.

d. The Way Forward - Integrated Project Support Environments

In the longer term INPUT foresees that the market will develop from the use of workbenches, toolkits and code generators towards an integrated project support environment approach.
The current problem with integrated workbenches such as IEW is lack of flexibility, although for many users it is a pragmatic, low-budget solution for medium-sized projects, especially for organisations that do not already use established structured methods and techniques in-house.

The major obstacles to the workbench approach are:

- Lack of effective support for project control, quality control, version control, and above all lack of configuration management for team work and baseline control.
- Lack of flexibility in providing toolkits to support a range of methods, including in-house methods.
- Size constraints of typical workbench architecture (i.e., PCs).
- Lack of ability to integrate additional tools as requirements evolve.

IPSEs such as ISTAR provide the key to the future for systems development by incorporating a design philosophy that supports the aims of integration, flexibility and openness in CASE product evolution.

Successful CASE products will provide:

- Support for the reliable construction of systems by all appropriate methods.
- Techniques for team work.
- Integrated project and configuration management support.
- A commonly accessable information database which allows access by graphical and textural tools.
- The ability for new tools to be added with common user interface access to the database.
- Flexible toolkits which allow for evolution in the techniques.

e. The Market Will Stabilise into Two Segments

Currently, the CASE tool market is unstable and relatively immature. However, INPUT foresees that in the future the CASE tool market will stabilise into two broad segments which can be classified by project size.

For large projects, i.e., greater than six months in duration or with a development team of more than twenty people, the IPSE approach will become a standard.

The IPSE tools help solve the problems of information exchange and facilitate objective assessment of quality against standards. They also allow standards to be incorporated into a number of appropriate methodologies.

For smaller projects the PC-based workbench approach will become a standard.

2. A Worldwide Market

Owing to the high R & D costs involved in developing advanced CASE products, the competitive environment is essentially worldwide.

INPUT's research revealed that leading players had invested in excess of $20,000 million in the development of IPSE products and full life cycle support systems.

For example, GPP's EPOS product has over 300 man years of effort invested in its development and IST's ISTAR product in excess of 200 man years.

In addition, a significant proportion of resources need to be allocated to ongoing research and development to upgrade product offerings in line with evolving market requirements. For example, SAGE Software Inc. spent 25% of their 1986 turnover of $15 million on R & D.

High R & D costs have had two strategic implications for CASE product vendors:

- Firstly, the need to establish consortia to share expertise and development costs.

- Secondly, the need to establish worldwide distribution channels. For smaller CASE tool suppliers this has been achieved via partnerships with major international players with extensive resources and support capabilities.

3. The United States: A Major Opportunity

The U.S.A. presents a major opportunity for vendors of advanced CASE tools for three reasons.

- Firstly, U.S. management style which exhibits suspicion of the commercial benefits of consortia.

- Secondly, the willingness of U.S. companies to purchase and experiment with leading edge software developments in comparison with Western Europe.

- Thirdly, the share size of the potential market.

Until the loosening of the anti-trust laws and reenactment of the 1980 Stevenson/Wylder Act in 1986, R & D consortia were effectively prohibited. Consequently, European players have a lead over the U.S. in terms of advanced CASE tool development.

Leading European players such as CGI and Softlab already have a presence in the U.S. market and there is room for further European players and ample growth opportunities.

Large U.S. pioneers of CASE such as IT&T have found it difficult to attract a critical mass of people to build an IPSE and are frightened by the long-term commercial payback from high initial investment.

Although, fifty-five U.S. R & D consortia have been established since 1986 to counter the threat of the Japanese fifth generation project and the encroachment of the Europeans, it is unlikely that U.S. vendors will produce an IPSE product before the 1990s.

Vendors with a track record of successful implementations in Europe are well-placed in the U.S. market.

B

Some U.S.-Based Players

Exhibit V-3 illustrates a list of some major existing and potential players in the CASE tool market. An analysis of some of these vendor organisations in terms of product and marketing strategy is given below.

EXHIBIT V-3

US-BASED VENDORS CASE PRODUCTS

COMPANY/CONSORTIUM	PRODUCT(S)
Arthur Andersen	Foundation
Computer Associates	CA Programmers Workbench
Cortex Corporation	Application Factory/Corvision
Index Technology Corp.	Excelerator XL/Real Time
Knowledgeware Inc./James Martin	Information Engineering Workbench
McDonnell Douglas	Prokit Workbench
Nastel Corporation	Life Cycle Manager Case 2000 Design Aid
Sage Software Inc.	APS
Texas Instruments/James Martin	Information Engineering Facility
Yourdon Inc.	Analyst/Designer Toolkit

INPUT has been able to identify over fifty U.S. owned players in the CASE tool market. The vast majority of their products address automation in the design and analysis or coding phases of the lifecycle.

The challenge of the future is to offer full life cycle support—a truly integrated CASE approach.

1. Cortex Corporation

a. General Background

Cortex Corporation is an American software house which specialises in building integrated application development tools to increase the productivity of software developers in the DEC VAX environment.

The company was founded in 1977. It has developed a number of advanced systems - notably Application Factory and CorVision. These products are marketed in the U.K., Ireland, and Benelux countries by Cortex Ltd., a subsidiary of RTZ Computer Services. Cortex Ltd. was set up in March 1986 with three people. Turnover in the first year was about £1 million and the number of employees increased to eighteen in this time. The company has its headquarters in Bristol and also has offices in London and in Antwerp, Belgium. In addition to supplying customers directly, Cortex Limited uses specialist distributors (value added resellers - VARs).

b. Products

i. Application Factory

Application Factory is an automated development system whose aim is to reduce the time and cost of application software development for DEC's VAX/VMS computers. It generates production applications directly from specifications, eliminating most of the hand coding required with conventional languages.

Approximately 500 Application Factory systems have been sold worldwide. The total cost of a system is about £35,000 in the U.K. It is expected that Application Factory will be phased out over the next few years, as its successor, CorVision, is made available on the market.

ii. CorVision

CorVision was launched by Cortex in September 1987. This product automates the entire software life cycle. Using a PC/AT compatible workstation linked to a DEC VAX, the developer uses the Picture Programming system to create a visual representation of an application. When the design specification is completed the code is created using an application generator (based on the same technology employed in Application Factory). Applications are also maintained using Picture Programming.

Cortex Limited expects to sell thirty to forty CorVision systems in 1988. The cost per system starts at £35,000. Target markets are in commercial applications such as retail systems, financial systems (e.g., Unit Trust administration), travel business, etc. Cortex has sold Application Factory to a number of software houses.

2. Knowledgeware, Inc./Arthur Young

a. General Background

Arthur Young currently employs around thirty people in the U.K. and generated approximately £3,000,000 in revenue from IEW during 1987.

b. Product

IEW is essentially a set of tools built around a database and designed to increase the productivity of systems builders. This is done by both automating the systems development cycle from the planning stage through design and construction and by using the database in conjunciton with the built-in rule base, the knowledge co-ordinator, to ensure internal consistency. As a result Arthur Young claims productivity increases of between 30% and 40%.

Knowledgeware has been shipping a PC AT version of the product since May, 1986. 3,500 copies of this version were sold worldwide by September, 1987.

Since launching the PC version, Knowledgeware has developed a similarly fully-linked unit of products for the mainframe environment. The mainframe version includes a fully-developed encyclopedia and the Gamma code generator. The new IEW/MF encyclopedia runs on MVS and DOS/VSE systems and is used with a VSAM database.

Other interesting features available in the PC and MF versions include an inport/export facility which allows the output from the planning/analysis and design phases to be linked to the other third-party products, for example, 4GL's.

Arthur Young claims that IEW is method independent and can be used with a wide range of methodologies, as well as their own version of Information Engineering.

IEW on the enhanced IBM PC AT costs about £5,450, with the mainframe version selling at around £35,000.

c. Users

The IEW products are designed to address the needs of anybody who develops computer systems. Knowledgeware believes that the way to educate the market is to provide a fairly low-cost entry-level version. This has been achieved with the PC-based workbench. Users now have the option to build up from there.

Arthur Young has 110 customers and 280 users for IEW in the U.K. There are about ten companies seriously using the product, including British Petroleum, B&Q, American Express, Glaxo, VAG, Central Independent Television, and Schroeders.

3. Texas Instruments (Information Engineering Facility)

a. General Background

Information Engineering Facility (IEF) was developed by James Martin Associates (JMA) in conjunction with Texas Instruments, and is marketed in Europe by JMA subsidiary Information Engineering Products (IEP).

James Martin Associates is a leading consultancy specialising in the information needs of major corporations. The company developed the Information Engineering Methodology from James Martin's original principles. The methodology enables information systems to be aligned to business objectives.

Texas Instruments, a $5 billion a year turnover leader in high technology industry, employed JMA and Information Engineering Methodology on internal projects. This led to the development of IEF prototypes.

IEF was first introduced by TI in 1985 for internal use and since June 1987 is available commercially.

b. Product

IEF is a software toolset consisting of an encyclopedia, a workbench, and a code generator.

The encyclopedia contains both an information base and a rule base, so it can be compared to an expert system. It provides facilities that allow the user to construct a model of the business, at all levels of detail from basic business objectives down to the content of an invoice.

The encyclopedia resides in a mainframe. Information is entered using the workbench on a personal computer. The information is obtained through discussion between an analyst and a user, with the workbench providing prompts at every stage.

Once the systems design has been completed, the code generator comes into play. This component is said to eliminate the need for conventional programming and to produce guaranteed error-free code.

The operating environment for the IEF's mainframe components is IBM's MVS, in conjunction with DB2, TSO and ISPF version 2. The operating environment for IEF's PC components is PC/MSDOD 3.0 or higher running on either TI's Business-Pro or the IBM PC/AT and compatibles.

JMA says that the IEF's design goal is to reduce the length of the system's development process by 20 - 80%, and to increase the productivity of systems developers by 100 - 300%.

c. Marketing

There are approximately twenty-five users of IEF worldwide. Some of its main competitors are Information Engineering Workshop from Knowledgeware, Pacbase from CGI and, more recently, a new product from Inforem.

TI and JMA had been working on IEF for about four years before it was launched on the market. The reason for this long development period, they say, is that they wanted to be absolutely sure that the product worked at all levels before releasing it to the general market.

It is interesting to note that the chairman of JMA, James Martin, is also a substantial shareholder in Knowledgeware. Martin was quoted in 1986 as saying that "the IEW is immesely more practical than IEF". More recently he has said that he is "happy that both the IEF and the IEW are now maturing into products which are capable of automating information engineering". He sees his role primarily as an independent observer and advisor, though his interest in both companies appears somewhat conflicting.

d. Price

The price depends on requirements; an installation comprising six workbenches, and both the other components, would cost around £210,000.

4. Sage Software Inc.

a. Company Background

Sage Software Inc. was founded in 1981. Based just outside Washington D.C., the company employs 125 people. Turnover for the 1986/1987 financial year was $14.5 million, representing an increase of almost 40% on the previous year. Total assets of the company amount to $24.9 million. Sage spends 40% of its resources on development and customer services. In December 1986, Sage Software Inc. went public.

Software Generation, Sage's U.K. distributor, was set up in May 1987 to market the APS Development Center products. The company is based in London and employs five staff members.

b. APS Product

The APS Development Centre offers integrated products which cover the entire development cycle. These products support the design, prototyping, development and maintenance of batch and online application systems.

All products are ISPF-based and dictionary-driven. They operate in MVS, MVS/XA, VM and PC DOS, targetting IMS DB/DC, CICS/DL1, DB2, VSAM, IDMS and Datacom/DB production environments.

The APS family of products address integration in the following ways:

- APS products are integrated with each other to provide the Development Centre with a single system image. This helps reduce the learning curve.

- APS products are integrated with IBM's strategic software direction. This means that investment in existing hardware and software is protected.

APS products share a layered language structure. This allows the user to choose from a full spectrum of non-procedural to semi-procedural language constructs, based on the specific project, staff, and performance requirements.

c. Markets

APS products are used on about 125 sites worldwide, with a total of over 4,000 users. The main user companies are large IBM mainframe sites and blue chip companies. Some of the major U.S. users are AT&T, Boeing, Philip Morris and Westinghouse.

There is an installed base of fifteen in Europe, two of them in the U.K. Most of the customer base is in the financial sector, though a recent addition was Ericsson in Sweden. Among the other users are Finivest in Italy and the Sedgwick Group in the U.K.

Sage has distributors in France, Germany, Italy, Benelux, and Scandinavia. These companies were all existing software vendors, unlike the U.K. distributor, which was set up specifically to market APS.

Software Generation hopes to increase the U.K. installed base to about eight in 1988.

d. Price

APS Development Centre is a modular system, and the price depends on which of the components are bought. Typically it would cost between £100,000 and £200,000.

5. McDonnell Douglas

a. Company Background

McDonnell Douglas Corporation is primarily a manufacturer of military and commercial aircraft and associated systems integrator. In the late 1970s the company diversified into external computer services.

McDonnell Douglas' information systems business was built up via acquisition (e.g. Tymshare) and organizational growth to reach a turnover of $1.2 billion in 1986.

McDonnell Douglas Information Systems has been expanding its U.K. operation through acquisitions and other investments over the past few years. In 1986 it took over Applied Research of Cambridge for £10 million, and in April 1987 acquired Isis Computer Services (turnover £3.3 million). It also announced a £12 million expansion plan in July 1987, which will create 700 new jobs.

b. Products

In the mid 1970s, McDonnell Douglas developed a structured approach to internal software development. Through working on their own projects, they were able to devleop the know-how which forms the basis for the CASE tools which were subsequently sold commercially.

In 1980 Stradis/Draw was launched. This was a graphics tool for the IBM mainframe VM environment.

In 1985 McDonnell Douglas launched the Prokit Analyst on the PC for automation of data flow and programming diagramming, including an integral dictionary.

McDonnell Douglas launched the Prokit Workbench in 1987. Prokit Workbench bundles together McDonnell Douglas' separate software development tools, Prokit Analyst, Stradis/Draw, DFDdraw, and SCdraw, providing data flow diagramming and entity modeling. In addition, Prokit Workbench contains a prototyping facility and an expanded data dictionary that underlies and unifies data from all the tools.

The main benefit of using Prokit Workbench, as outlined by McDonnell Douglas, is that system development efforts are performed efficiently and with greater success. The user's time can be spent analysing business needs and making professional decisions, rather than on manual drawing, checking and reporting.

McDonnell Douglas plans to link Prokit to a code generator (e.g., Transform or Telon). This is typical of the incremental approach to product development as adopted by many CASE tool suppliers.

6. Arthur Andersen

a. Company Background

Arthur Andersen is one of the "Big 8" accouting firms. One third of its revenues of almost $2 billion come from management consultancy, making it the largest consultancy company in the world. The main part of this business is information services sector (1986 revenues $485 million).

Arthur Andersen's revenue from information services in the U.K. and France amounted to about $30 million for each country in 1986. Total European information services revenues in 1986 were approximately $100 million.

Most of Arthur Andersens revenues (about 70%) comes from its U.S. activities. The company, however, is worldwide, with 214 offices in forty-nine countries.

The most recent initiative to be announced by Arthur Andersen in the U.K. is its Total Support Service (TSS). This offers to shoulder complete responsibility for customers' IT needs, from developing and installing new systems to managing networks and operating and supporting existing systems, in some cases buying a client's hardware and taking on its staff. The company is investing "tens of millions" of pounds in the service worldwide, indicating that it is moving further away from its accounting roots, and intensifying its competition with computer services firms.

Other European countries where Athur Andersen has a strong presence are France, Germany, Italy and Spain. In March 1987, the French firms formed GIE Arthur Andersen Informatique, bringing together its computer consultancy and its Coritel and Euromatique systems houses.

b. Products

Since the mid 1970s, Arthur Andersen has been building custom integrated environments for its clients. The testing facilities, structured architecture and standards it developed for its client work created the base for its new integrated systems development toolset, called Foundation.

Foundation is used for building transactions-orientated applications using IBM's DB2 database management system. It was developed to work as a single unit composed of three components: Method/1 for project management, Desing/1 for planning and design, and Install/1 for implementation and support. Each component can stand alone, but for highest productivity, Arthur Andersen recommends that all three be used in concert.

Some of the features foundation provides are:

- Full life cycle methodology.
- Project management.
- Diagramming tools.
- Screen and report painters.
- Prototyping.
- Checking and analysis.
- Dictionary and reports.
- Code generation.
- Test data management.
- Production systems support.
- Configuration management.

Arthur Andersen perceive the main advantages they can offer as:

- Good service (resulting from their long experience).
- A high degree of integration of the components.

c. Markets

While Design 1 and Method 1 have been used by clients of Arthur Andersen for some time, they are only now being sold separately as products. Install 1 is still undergoing tests in the U.S. It will be launched there at the beginning of 1988, and in the U.K. towards the end of the year. Only then will Arthur Andersen start to actively market Foundation in the U.K.

The U.S. potential market for Foundation is seen to be far bigger than the European one. There are, for example, ten times as many DB2 licenses in the U.S. as in the U.K.

d. Price

Method 1 costs £50,000. For automated methodology the customer pays an additional £5,000.

Design 1 is sold on a machine basis (i.e. depends on number of PCs using it). The first machine costs £10,000; each additional one costs £3,000. For large quantities Arthur Andersen gives volume discounts.

Install 1 will cost in the region of £125,000 to £130,000 when it becomes available in 1988. Its predecessor, PWB (which does not have the data dictionary of Install 1), costs £30,000.

C

Some U.K.-Based Players

Exhibit V-4 illustrates a list of some major existing and potential players in the CASE tool market. An analysis of some of these vendor organisations in terms of product and marketing strategy is given below.

INPUT's analysis of the U.K. market in terms of product supply revealed a plethora of CASE products, the vast majority of which only offer limited life cycle support, generally analysis and design.

EXHIBIT V-4A

UK-BASED VENDORS
CASE PRODUCTS

COMPANY/CONSORTIUM	PRODUCT(S)
Arthur Young (Knowledgeware)	IEW, RECODER
BIS	BIS/IPSE
British Telecom	Axion
GEC Software	Genos
ICL	Quick Build
Imperial Software Technology	ISTAR
Information Engineering Products	IEF
LBMS	Supermate
Manager Software Products	Manager View
Peat, Marwick, McClintock	Structured Retrofit
Scicon	Express
Systematica	Virtual Software Factory
Systems Designers	Perspective Kernal
Yard (Cap Group)	Lifespan

The challenge for the future is product integration. Inevitably the market will shake-out as successful vendors seek to enhance their product offerings via strategic partnering, distribution agreements, and lengthy research and development.

1. BIS (NYNEX)

a. Company Background

BIS Applied Systems is part of the BIS Group which was founded in the mid 1960s. By the end of 1986, the BIS headcount had grown to approximately 1,400 and the group turnover reached £62.7 million. (See detailed analysis of turnover below).

BIS Applied Systems provides a full range of consulting, training and system development services covering the information technology sector.

Clients include financial, government, manufacturing, distribution, oil, and public organisations.

The different services which make up BIS Applied Systems are:

- Management consultancy.
- Technical consultancy.
- Development division.
- Contract division.
- Training division.
- Productivity services division (Modus consultancy; IPSE/CASE marketing).

In February, 1987 BIS was taken over by the New York company Nynex, a spinoff of AT&T, for £75 million. This was followed by sweeping boardroom changes. Three directors from the U.S. telecommunications corporation were appointed and five members of the board resigned. Ron Yearsley stepped down as a director to become chairman of BIS Applied Systems.

EXHIBIT V-4B

BIS
TURNOVER ANALYSIS BY INDUSTRY SECTOR

INDUSTRY SECTOR	1984/85 (Percent)	1985/86 (Percent)	1986/87 (Percent)	1986/87 (£M)
Banking/ Finance	61.2	56.3	61.8	38.8
Manufacturing/ Engineering	19.8	20.9	15.9	10.0
Government/ Utilities	7.1	10.2	8.2	5.1
Other	11.9	12.6	14.1	8.8
Total	100.0	100.0	100.0	62.7

(Financial Year End February)

b. BIS/IPSE Product

The close link between methodologies and IPSEs is reflected in the development of the BIS/IPSE. The origins of this IPSE go back over twenty years. In the 1960s, emphasis within the data processing industry was placed on the definition and production of formal standards and procedures. In the 1970s, emphasis turned to structured techniques for systems analysis, systems design, and programming. At BIS, the acceptance and evolution of these techniques led to the development of the Modus set of methodologies.

The next step was to look at means of automating the system development process itself, which, using modus, is largely manual. Methodologies such as Modus are normally used within some sort of project-oriented environment, and an IPSE needs to support this sort of environment fully. It was not until the emergence of Unix and powerful multi-user micros that the development of an effective IPSE became possible, thus the arrival of the BIS/IPSE which is now one of the established IPSEs on the market.

EXHIBIT V-4C

BIS
TURNOVER ANALYSIS BY GEOGRAPHIC REGION

GEOGRAPHIC REGION	1984/85 (Percent)	1985/86 (Percent)	1986/87 (Percent)	1986/87 (£M)
United Kingdom	49.5	50.6	50.1	31.5
Rest of Europe	16.6	23.8	20.3	12.7
USA	16.9	13.5	11.7	7.3
Rest of World	19.0	12.1	17.8	11.2
Total	100.0	100.0	100.0	62.7

(Financial Year End February)

BIS/IPSE is designed to achieve benefits in terms of productivity gain, improved quality, and control by providing an environment which supports the formal conduct and management of systems development projects.

BIS/IPSE offers management of, and access to, all project documentation; support tools for formal methods and structured techniques; project control facilities; document completeness and consistency checks, together with design verification, through its integrated data dictionary software; and electronic document mailing through in-tray mechanisms. It brings particular benefit to the ongoing tasks of system maintenance and enhancement.

The IPSE support the Modus and SSADM strutured methodologies.

c. Users

BIS/IPSE is now being used at twenty sites. Typical users include banks and other financial institutions. The number of users for every IPSE varies between six and eighty.

One of its users, and one of the first organisations to make use of an IPSE in its day-to-day data processing work, is Pearl Assurance. Pearl adopted and implemented Modus procedures and supported them with the introduction of the BIS/IPSE. The intention is to provide an IPSE screen on each of the analyst and programmer's desks. Pearl maintains that the cost of an IPSE, at about £3,000 a user in this case, is justified by increased productivity.

2. GEC Software

a. Company Background

A subsidiary of The General Electric Company plc, GEC software was set up in 1983 specifically to make a profitable business in the market for software development tools. The GEC group had identified its own need for tools to improve productivity, quality, and control in big software projects and realised other companies had the same needs. GEC Software currently employs about ninety people.

In four years GEC Software has developed and brought in a set of products aimed at scientific and defense contractors who need an integrated project support environment (IPSE).

b. Products

GEC Software's first phase products were individual tools. Its "Project Management Toolset" is made up of:

- GECOMO (a cost modelling and resource allocation tool).
- G-TASKPLAN (for defining and evolving detailed work plans).
- G-CONTROL (automatically monitors work progress and reports on variances from plans).

In addition, GEC software is the European technology centre for the VERDIX Ada Development System (VADS), which comprises an ADA compiler, a symbolic debugger and a set of sixteen associated tools.

GEC Software's "second phase" product is an integrated product support environment. GENOS, a system manager, links together in one coherent system software tools for the design, development, management, control

and documentation of software projects. The above tools all work within GENOS, and are also usable as standalone tools. GENOS is a framework which is also open to users' own software tools, flexible in accepting any project methodology.

A major benefit of GENOS is its flexibility. It provides the freedom to use any software tool whether it has been developed by GEC Software or by another supplier. It is therefore more easily integrated with multiple methods of working.

It may be used on specialist networked workstations (Sun, Apollo) or on "COTS" (commercial off-the-shelf) hardware - HP9000 series, DEC VAX-VMS. There was a rapid uptake of orders for the VAX-VMS version following its launch in autumn 1987.

c. Marketing

GEC Software has been expanding its business activities outside the U.K. in 1987. It now has nine distributors in Sweden, France, Norway, Germany, Holland, Italy, and Switzerland. (European sales in 1986 accounted for 40% of GEC Software's revenues). In a shift away from its traditional defence sales territory, the company has made significant sales to software and consultantcy firms in France, West Germany and Sweden. It expects to double direct sales staff in 1987 to cope with growing demand.

A major coup was pulled off by GEC Software by selling its IPSE to Hewlett-Packard early in 1987. The deal, which could be worth several million dollars over the next few years, reverses the normal flow of sophisticated software between the U.K. and the U.S. HP has bought the GENOS and GECOMO software suites which will be run on the HP 9000 series 300 machine. GEC Software will also integrate into the IPSE a software engineering tool by HP and a software design system.

The Hewlett-Packard deal is the first major order which GEC Software has won for its IPSE. But is has built up a business worth some £3 million a year with other work.

d. Pricing

The price of GENOS ranges from £6,000 on a single Sun workstation to £32,000 for the VAX version. GEC software's price policy encourages customers to install several workstations if they are required. This is done

by charging a lower price for run time licences (i.e. extra workstations on a site) than for the master copy. Some other prices are: G-Taskplan at £10,000 to £31,000; G-Control £3,000 to £15,000; the enhanced GE-COMO £9,000 to £28,000, or £14,000 to £38,000 for the staff option. In addition to this cost, GEC Software's normal "recommended minimum" charge for training and support is £3,000 per licence, depending on machine size.

GENOS, GECOMO, and G-TASKPLAN are trademarks of GEC Software Ltd. VERDIX and VADS are trademarks of the VERDIX Corporation. ADA is a trademark of the U.S. ADA Joint Program Office.

3. Imperial Software Technology

a. Company Background

Formed in 1982, Imperial Software Technology (IST) is an independent company funded by Advent Capital Fund, Imperial College London, National Wetminister Bank, Murray Johnston Capital Fund and The Plessey Company. IST has earned a reputation as one of the U.K.'s leading software engineering houses, having found a ready outlet for its products and consultancy skills in the British and export markets.

b. ISTAR - IPSE

At the end of 1985 the company launched its ISTAR system, an IPSE developed in conjunction with British Telecom after the collapse of BT's MCHAPS (Minimum Chill Ada Programming
Support Environment) project in 1983.

ISTAR required 200 man years of development effort. The design process started with a definition of the overall requirements for software project support and associated database needs.

ISTAR supports the full product lifecycle with language independent tools for project management, configuration management and a wide range of technical development methods and languages. It may be adopted for either a single, major, project or a succession of smaller activities.

A key design objective for ISTAR was to provide the ability to smoothly integrate sets of "foreign" tools as part of workbenches that exploit ISTAR's user interface and data mangaement facilities. This ability has

been utilised to include workbenches for languages such as C and Pascal, using existing compilers and other language-oriented tools, in the initial ISTAR toolset. This approach has now been extended to Ada.

Few of the tools in the initial ISTAR toolset are unique. For example, the project management tools support planning, scheduling, resource allocation, and progress monitoring in much the same way as other proprietary project management systems. The difference is that ISTAR tools for project management, technical development and configuration management are integrated. Thus, completing a technical development task itself generates the management information needed to keep track of project progress, and the product of the task is inherently under the control of the ISTAR configuration management system. Also, in recognition of the continually evolving needs of software and systems producers, the ISTAR toolset is open-ended, allowing new or existing tools to be incorporated. Furthermore it supports a wide range of methodologies, e.g. SSADM, YOURDON, etc.

ISTAR avoids the pitfall of the massive central development database that forms a bottleneck in the previous generation of software support systems. Its linked network of databases, one for each active project task, lets it be distributed over a network of host machines. This is intended to improve ISTAR's efficiency, and allows it to provide support for distributed projects, whether the work is conducted on several hosts at the same site, or is geographically spread across different sites in the company.

c. Markets

The target markets for the ISTAR system are:

- Systems integrators.
- Defence contractors.
- Hardware manufacturers.
- Telecommunications companies.
- Electronics companies.

Among the first users of ISTAR was Motorola Israel. They bought the system in order to improve the productivity of their engineers and the quality of their software, and claimed a five-fold improvement in the first year. ISTAR was chosen as it could address both the hardware and software design problems throughout the life cycle and it enabled Motorola to use their own tools and those from IST.

d. Pricing

The cost of the ISTAR system for twenty users is around £100,000. This goes up to about £300,000 for a full corporate license.

Another alternative pricing arrangement which IST offers is a project licensing agreement, e.g. with British Aerospace for a NATO project.

e. Outlook

A key issue restraining IST's growth is the shortage of skilled people, as users have a very high need for training and support.

4. Learmonth & Burchett Management Systems (LBMS)

a. Company Background

LBMS was formed in 1977 and has offices in London and Houston. Consulting is its prime area of activity and the comapny has over seventy professional consultants.

Consulting assignments encompass many aspects of computer systems development in the public and commercial sector. The range of consultancy provided by LBMS covers:

- Systems development.
- Strategic studies.
- Structured methods support.
- Database and associated software product advice.

The LDMS trading record for the past five years is shown below (year end April; figures in £000s)

	1983	1984	1985	1986	1987
Pre-tax Profit	209	318	705	774	1,455
Turnover	2,140	3,031	3,659	4,922	7,563

In 1987, some 41% of turnover came from its consultancy operation, 28% from software and 29% from training. Out of the turnover, £1.79 million was generated in the U.S.

In September 1986, LBMS took over Sympact and in March 1987 they acquired Infotech Consultants. These acquisitions are part of LBMSs strategy to develop a full-service information system consultancy.

In June 1987, LBMS entered the Unlisted Securities Market (USM) via a placing valuing it at £21.3 million. The impetus came from former backer John Connelly, who wanted to reduce his stake to 12.5%, while partners Rainer Burchett and Roger Learmonth increased theirs to 32% each.

b. Products

In 1980, LBMS was selected to work with the U.K. government's Central Computer and Telecommunications Agency to develop a complete structured development method covering all aspects of systems analysis and design. The result was SSADM (Structured Systems Analysis and Design Method), which was made mandatory for U.K. government IT projects in January, 1983.

LSDM (LBMS Structured Development Method), the commercial version, is established as the U.K. data processing industry's system development standard. The method has been used in several hundred projects in both public and commercial data processing organisations.

Also used for the development of computer systems are the LBMS Database Design Method, which is concerned specifically with analysis and design of information in the system, and Office Technology Analysis Method, which covers the activities required before introducing office automation into an organisation. Other LBMS methods cover strategic planning through LEAP, LBMS Enterprise Analysis, and Planning Method; and management disciplines through PROMPT, a project and product management method.

To back up these methods, LBMS has a software products division which offers some advanced CASE tools.

Auto-Mate Plus is the main CASE tool. An updated version of Auto-Mate, it was launched in mid-1987. It can be used by systems analysts, designers and database administrators. Developers can use Auto-Mate Plus to assist with the definition of business and system requirements. It will help to extend that requirement's definition into an outline of a computer-based system. This can be extended into a detailed physical design and transported to the target hardware environment for final development, testing and implementation.

Auto-Mate Plus supports the Adabas, DB2 and IDMS management systems.

LBMS also has a strategic planning tool, Supermate. This is currently on the market and an enhanced version is being developed.

c. Marketing

IDMS database supplier Cullinet sells Auto-Mate Plus worldwide.

Auto-Mate Plus is considered to be a direct rival of James Martin Associates' Accelerator and Arthur Young's Information Engineering Workbench.

LBMS see good potential growth in the U.S. for Auto-Mate Plus. In 1987, some 50% of their U.S. business was in software sales. LBMS methods and consultancy should benefit from the success of the software products.

LBMS is jointly working with Hoskyns, developing a new product, LINK, combining LSDM and Hoskyns' project support environment, Prism-SDM.

5. Peat Marwick McClintock

a. Company Background

Peat Marwick McLintock was created in April 1987 by the worldwide merger of accountancy groups, Peat Marwick and Klynveld Main Goerdeler, which includes KMG Thomson McLintock in the U.K.. Worldwide the firm has 1,400 staff in IT consultancy. The 1985 worldwide consultancy revenues of Peat Marwick totalled around $240 million.

Peat Marwick McLintock is the U.K. biggest auditor, with a consultancy staff of 550, of whom 220 work on information technology. IT consultancy is the company's biggest growth area. In 1986 Peat Marwick McLintock's consultancy practice would have turned over £22.5 million in the U.K with around half coming from the information systems business.

b. Product

Structured Retrofit is an automated process that uses software tools and software engineering techniques to transform old, convoluted COBOL

programs to structured form. It is based on the premise of salvaging existing software systems and creating a workable foundation.

Some of the areas where structured Retrofit can be beneficial are:

- System development (incorporating exisitng software).
- Technology transfer throughout an organisation.
- Improvement in programmer productivity.

Structured Retrofit is designed to refurbish old, operational systems that:

- Are poorly documented.
- Tie up expensive staff in maintenance.
- Lack standards.
- Provide no firm foundation for quality assurance.
- Are heavily patched and modified in different styles.
- Are difficult to change.

Some of the benefits which result from using Structured Retrofit are:

- Improvement in structure.
- Cleaning up of language.
- Improved formatting.
- Improved understandability.
- Faster maintenance.
- Minimised disruption.

Peat Marwick McLintock also market a set of software tools called Tool Box, which was developed in the U.K. by their systems development group. Tool Box was created to help develop systems in Prime INFORMATION. (Peat Marwick McLintock's Systems Development Group is one of the largest Prime INFORMATION software houses in the U.K.).

Tool Box consists of a series of integrated utilities which include:

- Central data dictionary and security.
- Menu builder.
- Adhoc enquiry.
- High speed cross reference.
- INFO/BASIC pre-compiler.
- Screen handler.
- Transaction logging.
- Program change control.
- Performance and activity modelling.

Prime INFORMATION services include helping clients to create their own systems using Tool Box. Peat Marwick McLintock has developed systems for a number of U.K government bodies, an investment bank and for Prime's own in-house customer information system.

D

Some West German-Based Players

Exhibit V-5 illustrates a list of some major existing players in the CASE tool market. An analysis of these vendor organisations in terms of product and marketing strategy is given below.

The German software houses are the most innovative in terms of their usage and development of CASE tools which offer full life cycle support.

EXHIBIT V-5

WEST GERMANY-BASED VENDORS CASE PRODUCTS

COMPANY/CONSORTIUM	PRODUCT
GPP	EPOS
Softlab	Maestro
GEI	Promod
SCS (Scicon)	Prados

1. Softlab GMBH

a. Company Background

Maestro was developed by Softlab in cooperation with Philips. Over DM 25 million were spent on its development. Maestro is distributed in Europe by Philips.

Softlab GmbH is a German software house which is active in the European and U.S. markets. It employs about 300 people and had a turnover in 1986 of around DM 68 million.

In 1986 a U.S. subsidiary, Softlab Inc., was set up in California.

In August 1987, car manufacturer BMW took a 10% share in Softlab. It is expected that BMW will buy a further 18% by the end of 1987, and eventually buy out Softlab completely.

b. Maestro Product

Maestro is a dedicated software engineering system based on the Philips P7000 minicomputer and provides information management with multi-file access; syntax guidance and menu prompting; structured design aids and programming aids; code generators for various target languages; integrated tools links to target processors; machine testing support; and project management functions. Code development work is done on the dedicated Maestro machine, P7000, and then the generated code is sent to the mainframe where it is compiled and tested.

Softlab claims that Maestro is the "most successful dedicated software development system in the world". It has over 12,000 users in the U.S. and Europe in over 500 sites.

c. Markets

Around 60% of Maestro customers are big IBM users. Among the companies using the system are Boeing, British Airways, Dresdner Bank, Shell, Volkswagen.

Softlab achieves 55% of its $28 million revenues from Maestro and two other products. Philips in the U.K. gained around £7.5 million in revenues from Maestro alone in 1986.

The Bank of Scotland used Maestro to develop the software for the IBM mainframe side of its Cabinet project. It is also planning to use Maestro to develop C code for the P9000, which will be the first time C code has been developed using Maestro in the U.K., although it has been done a couple of times before on the continent.

Philips claims that the networked Maestro systems at the U.K. Inland Revenue's Telford Development Centre constitutes the world's largest integrated project support environment. The 24 Maestro ipses support 656 users and can all access a central project database.

d. Product Positioning

Much of the appeal of Maestro has been to data processing departments with a large maintenance overhead. Softlab maintains that this can be

reduced with the help of Maestro, while general productivity is improved and higher quality software created.

Maestro is positioned at the top end of the development aids market. It is unusual in that it is not wedded to a particular systems design methodology. Philips argues that many data processing departments would like to use a proprietary methodology but do not have time to retrofit it to the large, complex systems that they must maintain.

e. Features

Another unusual aspect of Maestro is that it relies on customised hardware. Softlab developed Maestro for its own use and specified a hardware design which would suit a development environment. Two key characteristics which they highlight are Maestro's ability to provide networking features and fast response times to terminal users.

At a fundamental level, Maestro acts as a working environment for the whole data processing department. It can be used for electronic mail, source-code creation and editing, version control, and configuration management. It leaves the technicalities of code generation to the programmer and the mainframe compiler. Maestro's new graphics workstation can be programmed to work with different design methodologies and display them graphically.

2. GPP GMBH

a. General Background

EPOS was developed by the Institute of Control Engineering and Process Automation at the University of Stuttgart with German government funding. Over 300 man years were invested in the product.

EPOS is marketed in Germany by GPP (Gesellschaft fuer Prozessrechnerprogrammierung). It is distributed in the U.K. by Systematica, a company founded by a group of former Plessey software specialists.

Systematica specialises in software engineering, providing a full range of supporting services, including training, technical support, and project management consulting, to aid in the transition from software-development tools to the latest support systems available.

b. Product

EPOS is an acronym for Engineering and Project-management Oriented Support-system. It is a fully integrated tool system designed to improve productivity in software/hardware projects, and at the same time make improvements in quality. To achieve these goals, the following principles are applied:

- Integrated computer support for all activity areas and all phases of a project.
- Support for the entire software/hardware system.
- Ability to use various design methods.
- Extensive graphics support.
- Support for development in teams (using decentralised EPOS work stations).

Some benefits of EPOS are:

- Reduction in delivery times.
- Elimination of documentation maintenance.
- Fully re-usable designs.
- Good quality and reliability.
- Reduced cost of changes.
- Easy integration with existing tools.

In view of EPOS's open architecture, it can be used with several methodologies, depending on their appropriateness for a particular project, e.g. Jackson, Yourdon, Petrinet, Core, Mascot.

c. Markets

EPOS has been widely applied in commerce, telecoms, defense and process control. By early 1987, there were over 250 installations worldwide, with a forecast of 400 for the end of the year.

EPOS is approved by the U.K. Ministry of Defense and is being used, for example, in Ferranti's Mandated European Fighter Aircraft project. It is also being used by Grumman Data Systems for their $40 million systems integration contract with Rolls Royce. This is the largest SI contract in

the U.K manufacturing sector to date. Other leading companies which use EPOS are Intel, IBM, Digital Equipment, Plessey, British Aerospace, Philips and Renault.

d. Price

The cost of a single-user software license is about £6,000. In addition there are normally training and consultancy fees.

3. GEI GMBH

a. Company Background

GEI (Gesellschaft für Elektronische Informationsverarbeitung) is one of West Germany's leading systems companies. Worldwide revenues were 87 million DM in 1986 and 100 million DM in 1987. GEI is committed to international expansion via subsidiaries (e.g., Switzerland and the U.S.A.) and licensing arrangements. Nevertheless, 90% of 1987 revenues were generated in West Germany.

AEG, which has had a 50% share in GEI since 1981, has now taken a majority shareholding and complete ownership of the company. The AEG subsidiary, AEG Software Technik, (whose revenues are mainly captive), has been integrated with GEI. One of their main customers willl be Daimler Benz, which in turn owns AEG.

GEI employs approximately 550 staff of which 90 are outside Germany. GEI's key market focus is in the areas of CIM and CASE tools. GEI generated approximately 9 million DM worldwide during 1987 from the sale of their integrated CASE environment PROMOD and associated training.

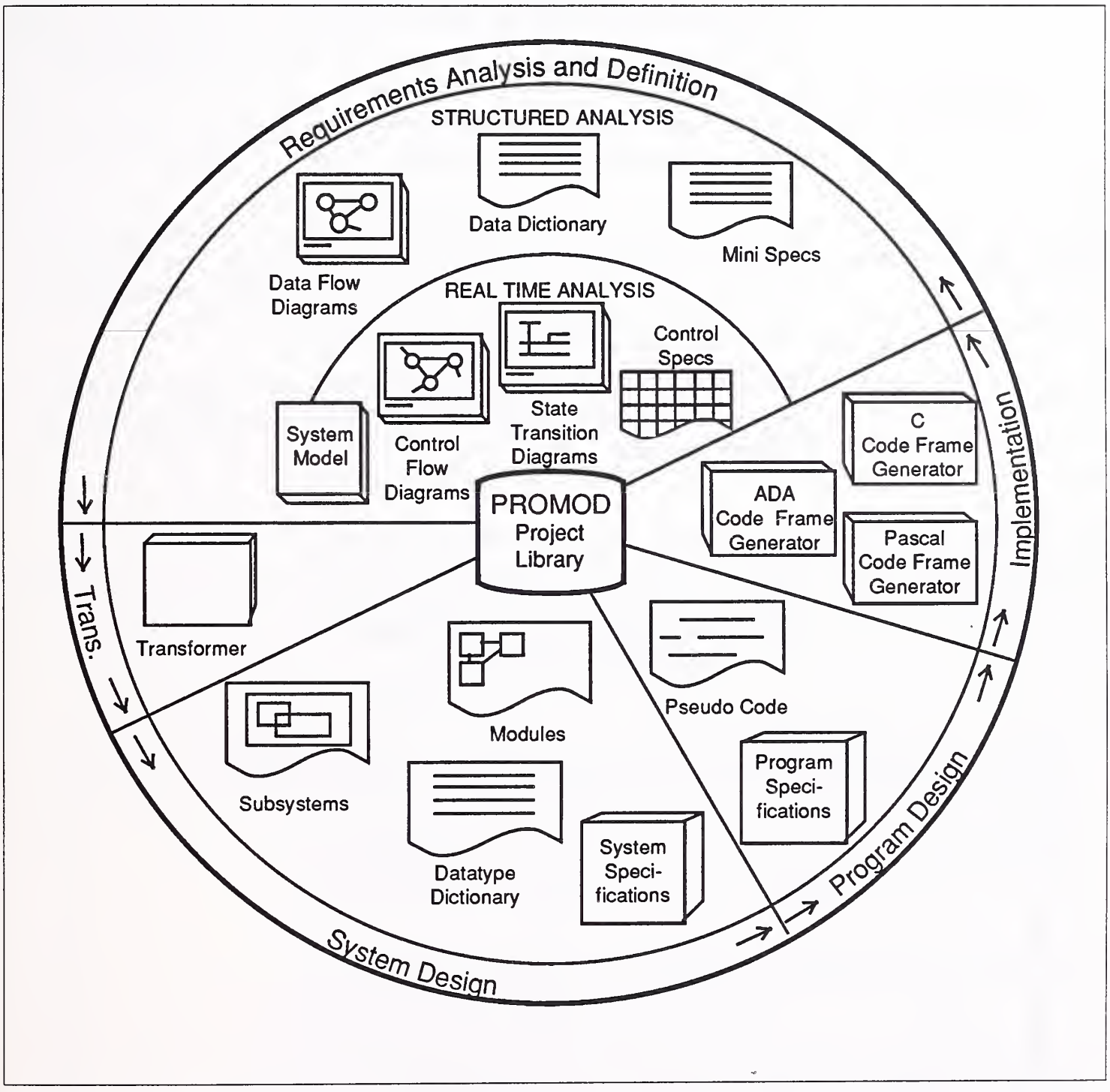
b. Product

GEI-PROMOD is currently vying with GPP's EPOS product to become the leading CASE tool in the German market for technically oriented applications.

PROMOD provides users with an integrated CASE environment and a linked set of tools covering all phases of the development life cycle from analysis to the creation of pseudo code. PROMOD effectively supports a wide variety of proprietary and in-house structured methods.

A schematic of the product functionality is illustrated on the next page.

SCHEMATIC



c. Price

PROMOD is available on an IBM PC plus compatibles with 640 KB minimum memory priced at 25,000 DM. It is also available for the full DEC VAX series. VAX prices start at 30,600 DM for the top of the range VAX 8800.

d. Market Strategy

GEI's strategic goal for PROMOD is to become the leading German CASE supplier in the defence area. The product set already supports real time structured analysis and includes an ADA code frame generator.

Future plans include the addition of a full code generator and effective tool support for reverse engineering.

E

Some French-Based Players

Exhibit V-6 is a list of some major existing players in the CASE tool market. An analysis of some of these vendor organisations in terms of product and marketing strategy is given below.

Until recently, the French software houses have been content to focus their product development efforts around the Espirit projects, for example, Emeraude.

EXHIBIT V-6

FRENCH-BASED VENDORS
CASE PRODUCTS

COMPANY/CONSORTIUM	PRODUCT(S)
GIE Emeraude Group	Emeraude
CGI Infomatique	Pacbase
MOD/Aerospace Consortium	Enterprise 2
CNET	Concerto
CAP Gemini Instruments	Multipro
CISI	Atlas

In addition, existing CASE products tend to be highly specialised in terms of application. Examples include Enterprise in the defense contracting area and Concerto in telecommunications.

However, there are signs of the development of CASE products for general commercial users and software houses developing commercial applications.

1. CGI Informatique

a. General Background

CGI Informatique developed the Pacbase product in France in the early 1970s. The system is based on the Merise methodology, a structured programming technique popular in France. The system runs on IBM mainframes under DOS, DOS/VSE and MVS as well as on Honeywell and Unisys large-scale systems.

Pacbase is installed at 500 sites worldwide, including 90 in the U.S. Worldwide revenue was $77 million in 1986. In 1987, U.S. revenue is expected to exceed revenue derived from other countries for the first time. The U.S. subsidiary which markets Pacbase is CGI Systems, Inc.

b. Product

Pacbase is a COBOL generator that also provides schematic design and analysis and a central data dictionary. It features a series of fill-in-the-blank screens. The programmers create screen layouts or report layouts, and it generates basic skeleton programs. The next step is filling in the cryptic patterns of the fill-in-the-blank parameters.

The programmers do that in place of writing code, which develops the next step. For a last option, or for sophisticated programs, procedural code is available. This is shorthand COBOL.
All those pieces are put together and run through a pre-compiler, and COBOL source code is produced and compiled like any other COBOL compiler.

CGI has developed project management tools to help guide the management as well as the application life cycle. The company has announced that it is developing a graphics front-end.

c. Users' Perceptions of Product

Some of Pacbase's strong points are:

- Good cross-reference, e.g. whenever an element in any kind of program is used, it is automatically stored as a cross-reference in the dictionary.
- Good technical documentation is automatically produced.
- The on-line system is good.

Some weak points outlined by users are:

- Implementation is often cumbersome and primitive.
- Difficult to use.
- CGI's education is poor.

d. Marketing

Users have commented that Pacbase is a long-term investment. Data processing departments using Pacbase can expect productivity drops in their first project. The payoff comes with the second, third, fourth projects and so on.

CGI itself says that generator-based products such as Pacbase (another one being Application Factory from Cortex) are not for everybody. Some organisations who simply want to generate third-generation language code faster should use a straight code generator, they say.

2. CAP Gemini Sogeti

a. General Background

CAP Gemini Sogeti (CGS) is Europe's largest independent software house. Its three billion franc turnover is geographically divided evenly in thirds between the U.S., France, and the rest of Europe.
The company has been expanding rapidly in the past few years, mainly through acquisitions. In 1985, it bought U.S.-based CGA-Computer. Expansion into Germany culminated in the takeover of Ibat, while in Italy, GD-DA was added to the list last year. In June 1987, CGS took a 36% share in the French company, CISI (1986 turnover FF1.53 billion), and the remainder of its shareholding in SESA. CGS has a full quotation on the Paris Stock Exchange since June, 1987.

CGS concentrates mainly on custom software for large companies. It has also developed some software products, e.g. the Multitel range of videotex software and the Multipro software development workbench.

b. Product

The Multipro workbench is an intelligent workstation consisting of an IBM PC with Multipro software operating under MS-DOS, the PC operating system. The workstation is linked to the mainframe, where Multipro software manages the Multipro project libraries and communicates with the workstations.

The Multipro project libraries on the mainframe are accessible by any workstation. Multipro can be used by all members of the development staff, not just programmers. Analysts use it in the design stages to create system specifications, secretaries use it to generate system documentation components, and project leaders use it to manage the overall project.

c. Marketing

Multipro is marketed and supported in France by Cap Sogeti Instruments, a subsidiary of CGS. It is also marketed throughout the rest of Europe, although it is not pushed very strongly in countries where the CASE market is underdeveloped, e.g. Sweden. In the U.S., Multipro is marketed and supported by Cap Gemini Software Products Inc., which was set up in 1983 for this specific purpose.

More than 1,500 Multipro workstations, spread over fifty sites, have been installed worldwide. Among the French users are Banque Nationale de Paris, Societe Generale, Aerospatiale and Michelin. In the U.S. The Bank of America and the Security Pacific National Bank are users.

F

Some Western European Research - Based Players

In recognition of the high R&D costs coupled with the strategic importance of leading CASE developments, European governments have helped fund a variety of consortia to undertake projects as part of the Esprit and Eureka programmes.

Advanced third generation IPSE products are expected to emerge from collaborative research efforts such as the IPSE 2.5 project.

Second generation IPSEs, such as Systems Deseigners "Perspective Kernal" and Software Sciences "Eclipse", have emerged from projects partially funded by ALVEY.

Exhibit V-7 illustrates some current research-based CASE projects. The list is far from exhaustive.

Current Esprit projects include the development of the Sphinx Software Factory, which uses the PCTE Standard as a common framework for development.

Other interesting research projects include the Esprit IMP project which has the objective of developing a knowledge-based project management workbench.

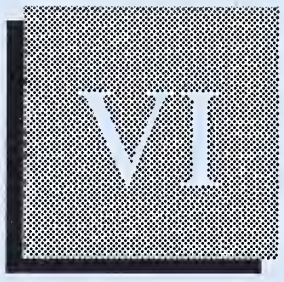
EXHIBIT V-7

SOME RESEARCH-BASED CASE PRODUCTS/PROJECTS

PROJECT	OBJECTIVE/PRODUCT
Esprit (Pan-European Consortium)	Sphinx Software Factory
Esprit (Consortium)	Tool Development Standard PCTE (Portable Common Tool Environment)
ALVEY	Eclipse - Project Support Environment
Eureka (Logica/IST/Alsys)	Tribune - IPSE
Esprit (Consortium)	Integrated Management Process Workbench (Knowledge Based Project Management System)
Mjolner (Swedish Consortium)	Object-Oriented Programming
Eureka	European Advanced Software Tools Project (EAST)

The Mjolner project in Scandinavia is evaluating the commercial use of object-orientated programming. Object-orientated programming will facilitate more easily understandable modeling of reality in programme development.

The next chapter gives INPUT's conclusions and recommendations for both user and vendor participants in the market for CASE software and services.



Conclusions and Recommendations

Conclusions and Recommendations

Difficulties arise in specifying recommendations to either vendors or users of CASE products and services. Appropriate strategies differ depending on the type of service, stage in company development and/or product life cycle, competitive conditions, and country market lifecycle.

General trends, issues and opportunities for CASE market participants have been dealt with in Chapter III of this report; however, INPUT feels that it is appropriate to emphasize a few key points that organisations need to consider, but not necessarily address, if they are to exploit fully the opportunity of this evolving approach towards systems development.

A

Recommendations to Vendors of CASE Products and Services

Vendors should be cognizant that the market for individual CASE tools and toolkits will shake-out.

Users are dissatisfied with products that do not offer full life cycle support, do not support multiple methodologies, do not interface effectively with existing databases and 4GLs, do not address the issue of maintaining existing systems and are not flexible to allow for evolution in software development methods and techniques.

There is a plethora of individual tools on the market and shake-out will follow users' increasing preferences for complete CASE systems.

In recognition of the fact that the major issue in CASE implementation is changing working practices, both of software developers and also management organisation and style, vendors should provide either individually, or via strategic partnerhsip, a comprehensive implementation support service.

Comprehensive implementation support means training in structured methods and tools to support methods, as well as training in business/ management skills and software project/quality management. Consultancy is essential to ease the process and management of change. In other words, one should work with the people, not the technology.

As the CASE market is currently unstable and immature, vendors need to address the issue of market education. Missionary marketing initiatives such as discussion workshops, seminars and quality circles are key to this process.

As CASE is perceived as a high risk technology by many user organisations, vendors not only need a comprehensive service solution, but also need to target and support the CASE "product champions" in user organisations.

Effective marketing of CASE products and services has essentially a double-edged benefit for vendor organisations. As well as being a potentially very profitable line of business, CASE also has an impact on the perennial problem of recruiting and retaining personnel.

The personnel benefit of effective CASE product marketing is that quality people enjoy working in a quality environment. CASE products by themselves do not create a quality professional environment but they are a central plank in its implementation.

Another key marketing-related recommendation is to carefully evaluate corporate strategy with respect to distribution channel management.

Distribution, joint marketing agreements, and strategic partnerships are key to achieving the objectives of providing a full and independent implementation support service. They are also key to addressing the challenges of operating in a worldwide marketplace.

Vendors should be strict in the criteria used to select strategic partners and distributors. To marry in haste could well lead to divorce in tears, with a large number of dissatisfied clients. INPUT will provide you with help in selecting partners and appropriate selection criteria if this is required, on request.

One potentially successful strategy for channel management is to create a network of specialist vertical market value-added resellers (VARs). As knowledge of a client's business needs and problems is more important to

marketing success than technical excellence, this is an attractive approach.

Another key recommendation relates to the effective synchronisation of marketing and research and development activity. The image of CASE product vendor has been tarnished by organisations marketing products which have not been fully developed and rigorously tested.

The credibility of the CASE solution and users' resistance to change have not been helped by the "Vapourware" marketing tactics and unprofessional sales tactics of some CASE vendors.

The key to future success in terms of product development is essentially a commitment to openness, flexibility, and integration on the part of CASE product suppliers.

Essentially, this means an integrated approach towards supporting all development activities, including configuration and project control. It also means integrated support for team-working and access to central and distributed databases by all team members.

It means flexibitity in the ability to incorporate in-house tools and methods and flexibility to allow for evolution in techniques and methods.

It means openness in the ability to incorporate new tools and methods.

Vendors also need to address the major issue of maintenance of existing programmes. CASE implementation will only be fully effective when users can incorporate existing programmes into new systems which are developed using CASE tools.

Exhibit VI-1 highlights INPUT's recommendations for vendors of CASE products and services.

B

Recommendations for Users of CASE Product and Services

In recognition of the fact that many potential users of this report are also existing and potential users of advanced CASE products and services, INPUT has prepared some general guidelines for consideration.

In the broadest possible terms, it is recommended that users should only contract with vendors that meet the challenge of INPUT's recommendations outlined in the previous report section.

EXHIBIT VI-1

RECOMMENDATIONS FOR VENDORS OF CASE PRODUCTS AND SERVICES

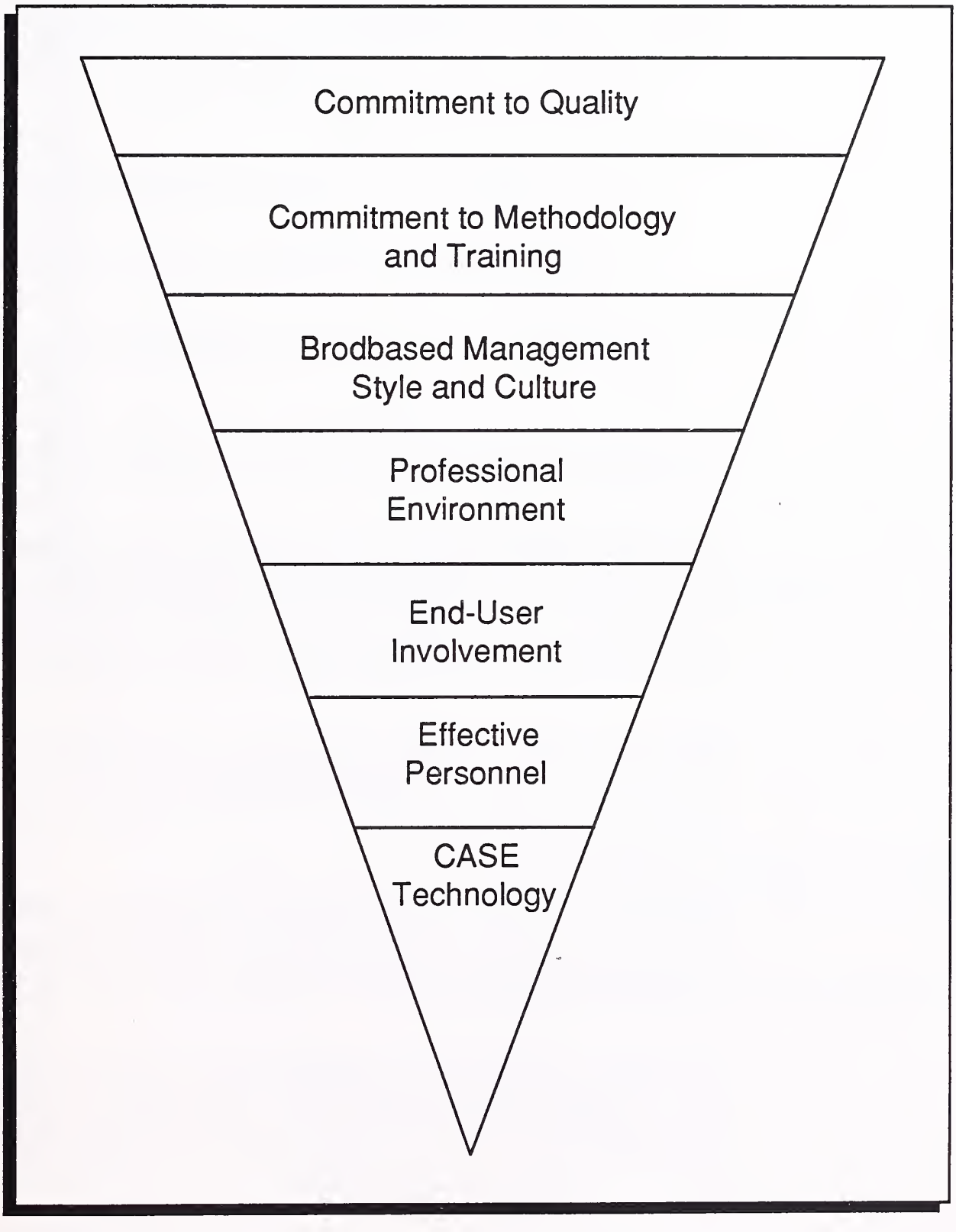
- Achieve Critical Market Position in Terms of Product/ Service/Distribution Offering—Avoid the Shake Out!
- Provide Comprehensive, and Preferably Independent, Implementation Support Services.
- Carefully Review and Evaluate Strategy for Managing Channels of Distribution.
- Consider Joint Ventures and Strategic Partnerships.
- Effective CASE Marketing—Personnel Management Benefits.
- Educate the Market.
- Target the CASE Champions.
- Dispel the "Vapourware" Marketing Image.
- Commit to the Principles of Openess, Flexibility and Integration in Product Development.

The cornerstone of INPUT's recommendations to users is the establishment of an effective software productivity improvement plan, the elements of which are illustrated in Exhibit VI-2.

The overall rationale behind this plan is the recognition that the major factors impacting on software quality and productivity are firstly, management style and methods, secondly, methodologies employed, and thirdly, technology used.

EXHIBIT VI-2

INPUT'S SOLUTION FOR USERS PRODUCTIVITY IMPROVEMENT PLAN



Consequently, it is important to focus on the human resource management issues in systems development.

This means a professional environment and broad-based management style and culture.

Success in software development requires a mature, professional environment with sound reporting structures and effective communications.

The features of this environment are:

- Everyone is treated (and must behave) like a professional.
- Everyone must be personally committed to quality and well-understood objectives.
- Management control must follow common sense and not be overly rigid.
- There must be a broad-based commitment to methodology and training.

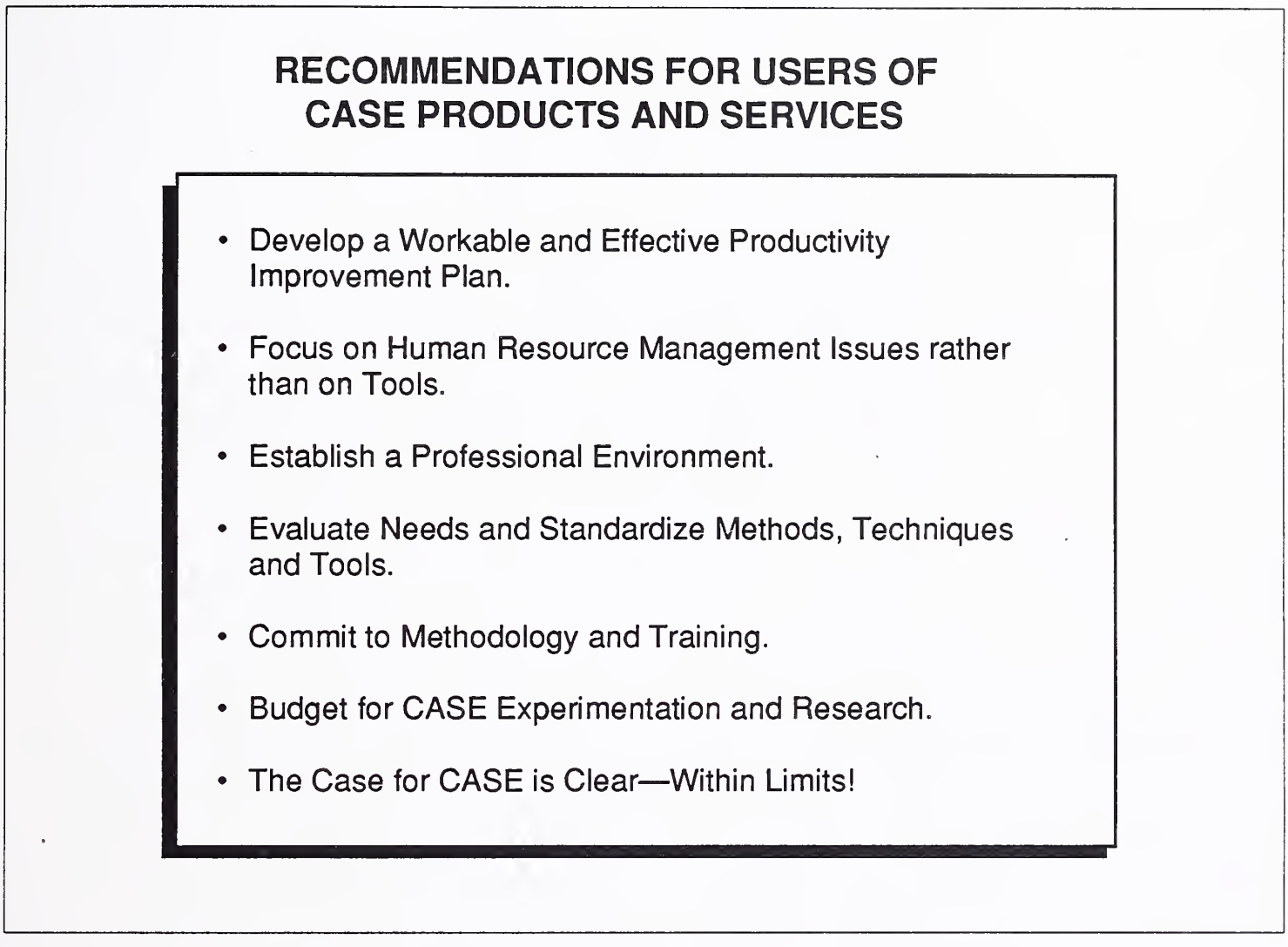
In terms of CASE technology, it is recommended that organisations standardize the use of tools and methods that will give them the greatest productivity. A central element of this standardization process is internal market research on the existing usage and attitudes towards productivity tools. Also, requirements for the future should also be captured.

Internal and external training courses are also a key investment. In addition, it is recommended that regular briefing sessions and discussion workshops are organised in order to disseminate knowledge on advanced techniques, tools and methods.

In addition to specific training on advanced techniques, it is recommended that developers and managers commit to general training. That is, management and personal skills training as well as general updating via books, seminars, etc.

With respect to CASE, users need to be cognizant of the fact that in its current, relatively immature state, CASE is a potentially risky technology which needs to be carefully evaluated against precise short- and long-term needs.

EXHIBIT VI-3



It is recommended that users budget for initial experimentation with CASE technology.

Ideally, CASE could be evaluated within an advanced technology group that is isolated from the rest of the organisation. This approach has the benefit of minimal description to ongoing activities.

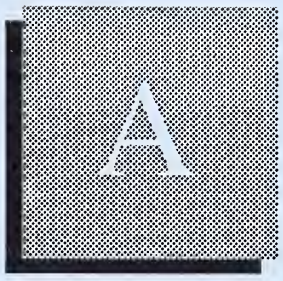
Overall, the case for CASE is clear. It gives users the benefits of:

- Improved morale and job satisfaction.
- It makes structured techniques practical.
- It improves software quality.

- If frees developers to focus on the creative aspects of software development.
- It simplifies programme maintenance.
- It reduces development lead times.
- It supports a disciplined information engineering approach.
- It allows system development to respond to business development.

INPUT's key user recommendation is to adopt CASE with caution. It needs to be implemented in the context of a mature, professional management environment.

Exhibit VI-3 summarises INPUT's recommendations to CASE users.



Appendix: Software Engineering Terminology

Appendix: Software Engineering Terminology

The following are definitions in alphabetical order of terms used in software engineering:

- Aid - A computer based-technique.
- Automated - Computer based.
- Data Analysis - A formal method of examining data structures to ensure their freedom from embedded information (such as repeating groups, inter-item dependency, etc.) which would increase maintenance difficulties.
- Definition or Proposal Stage - Completes when the customer for the system agrees an overall target specification.
- Design Test Plan - Indicates the types of tests considered at each level of implementation.
- Development Library - Holding different versions of software and controlling their release and use.
- Diagrammatic Representation of Data Flows - Can be used in entity, logical, and physical modeling. Techniques include Structured Analysis and Design Techniques (SADT), Yourdon, Warnier-Orr and Gane and Sarson.
- Enhancement - Meet revised requirements.

- Entity Modeling - Identifying the source and destination of data, data in the system, and its transformation and storage.

- Integration Testing - Testing of linked modules or programs as a unit.

- Logical Model - States what the system is required to do, expressed in data flow and data relationship diagrams, in such a way as to be readily understood by non-technical users. The logical model may then be verified in conjunction with the user and against the stated used requirements.

- Methodology - A methodology is considered to be a formal collection of explicit or referenced principles, policies, methods, procedures, practices, and definitions detailing how something is to be accomplished. Essentially, it provides a route map for project personnel-related to each key project development aspects.

- Maintenance - Correction of the operational system.

- Physical Model - Maps the logical model onto the hardware and software. It is created by the technical system designer. Diagramming techniques may be used to derive a structured design which can be verified.

- Portability - Capacity of the software product to be moved to other hardware.

- Project Management - Sets out how projects will be accomplished and specifies the interfaces with other organisational groups such as users. Techniques can include project estimating, planning and control of resources, monitoring progress, authorisations, use of other methodologies and development techniques. A policy statement for the project or, indeed all projects, may be one technique; use of a Project Manual as a reference and a record of decisions can be another.

- Project Stage Documentation and Change Control - Ensures that each stage of a project has a formally documented end-product that is subject to formal review and acceptance sign-off and from then on subject to formal change control to ensure that all levels of documentation fully incorporate approved changes; also ensures configuration control of product build-state.

- Project Stages - Are of necessity somewhat arbitrary but are considered to complete as follows:

  - Definition or proposal completes when the customer for the system agrees on overall target specification.

  - System specification and design are complete when an initial design document identifies and to some extent defines the six to twenty most significant items in terms of design effort.

  - Software design and development are completed when operational use of the system begins. In certain circumstances an earlier completion is acceptable where:

    i. Extensive parallel running is required for customer acceptance.

    ii. The system is put "on ice" awaiting an external event.

In these cases it is usually identifiable that a maintenance team is in post and therefore the project is complete.

- Prototyping - Involves demonstrating, confirming, or improving the system specification by building a subset or model of the full system.

- Quality Assurance (QA) - Applied to specifications and software end products. Also QA may be applied to the means by which such products are produced, i.e. project organisation, plans, communications control procedures, standards, reviews, or walkthroughs.

- Software Construction and Documentation Methodology - Using techniques such as:

  - Structured program design through use of data structure analysis, e.g. Warnier-Orr, Jackson.

  - Structued coding to implement programs through use of sequence selection and iteration language constructs.

  - Top down software build using a module hierarchy beginning at the root level then progressing through more detailed levels.

- Development library holding different versions of software and controlling their release and use.

- Software generators providing automatic code generation for screen formats, reports and programs.

- Walkthroughs of code in an informal manner to critique all related aspects against the programming concensus.

- Software Design and Development - Implementation of the software system through detailed design, code, and test.

- Software Engineering - Encompasses all technical and management activities undertaken during the development of software systems from initial requirements defintion through implementation to customer acceptance and beyond into systems maintenance.

- Software Engineers - Are those people involved in the production of software at either a management or technical level and thus include specifiers, designers, programmers, quality staff, document action specialists and managers.

- Software Engineering Tools - Are automated techniques which support some component part of a methodology. Tools are available to support all types of methodologies; including:

  - Project management.
  - Stage documentation and change control.
  - Quality assurance.
  - Systems requirements.
  - Systems design.
  - Software construction and documentation.
  - Software and system testing.

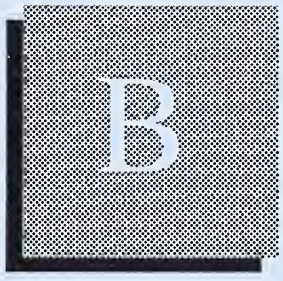
When discussing the potential market for software engineering tools, only substantial tools which might cost in excess of £5,000 for medium to large systems, have been considered.

- Software Life Cycle - Used to cover all aspects of product definition, development, installation, and maintenance. This is in contrast to software development cycle.

- Software Development Cycle - Covers only system design, software engineering, coding, testing, and integration.

- Software Generators - Are created by a project team using the SE disiciplines. The project life cycle is split into project stages.
- Specification - A software system requirement.
- Specification Team Walkthroughs - Review and confirm in an informal manner the system specification.
- Structured Coding - Implementation of programs through use of sequence selection and iteration language constructs.
- Structured Program Derived Test Data - That which uses the program structure to ensure each path through the developed code is exercised.
- Structured Program Design - Through use of data structure analysis, e.g. Warnier-Orr, Jackson.
- System Design - Initial translation of the specification into the main components for implementation.
- System Requirements - Producing the specification using techniques such as entity modeling, prototyping, or specification team walkthroughs.
- Systems Specification and Design - Completes when an initial design document identifies and to some extent defines the six to twenty most significant items in terms of design effort.
- Techniques - Component parts of a methodology.
- Test Plans - Setting out test objectives, timescales, inputs, and expected results.
- Test Team - Responsible for independently conducting tests and formally returning or accepting delivered programs.
- Testing Methodology - The collection of techniques for testing including test plans, top down testing, test team, and integration testing.
- Tool - Automated technique.

- Top Down Program Testing - To progressively exercise more and more of the programm as it is built to ensure all module paths and linkages are tested.
- Top Down Software Build - Uses a module hierarchy beginning at the root level then progressing through more detailed levels.
- Top Down System Testing - Used in a similar incremental way to ensure consistency between programs regarding use of files.
- Unit Testing - Carried out by the programmer responsible for developing code.
- Visibility - Capability for outsiders to assess project progress.
- Walkthroughs of Code - To critique, in an informal manner, all related aspects against the programming concensus.



Appendix: Analysis of Research Sample

Appendix: Analysis of Research Sample

In-depth interviews (nearly all face-to-face) were conducted amongst major software houses and vendors of systems development productivity tools, workbenches, applications generators, advanced programming languages, and integrated project support environments (IPSEs).

Exhibit B-2 shows the analysis of the vendor organisations interviewed.

Telephone interviews were also conducted amongst a wide cross section of user organisations in France, West Germany, Italy, Belgium, the Netherlands, Sweden, Finland, Norway, and the United Kingdom.

This user research was conducted as part of INPUT's bi-annual survey of Western European Software and Services markets which focused on usage and attitude towards systems development productivity tools and techniques as well as software product pricing and support.

Exhibit B-1 shows the analysis of user organisations interviewed.

In addition, a number of key software development executives in user organisations were interviewed face-to-face in order to clarify INPUT's understanding of key trends and issues.

EXHIBIT B-1

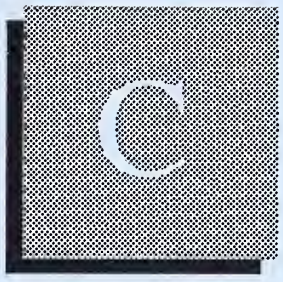
CORPORATE USER INTERVIEWS SAMPLING FRAME

INDUSTRY SECTOR	NO. OF INTERVIEWS	COUNTRY MARKET	NO. OF INTERVIEWS
Manufacturing	53	United Kingdom	40
Finance & Banking	38	France	30
Insurance	18	West Germany	40
Retail Distribution	19	Italy	30
Transportation (Private Sector)	22	Scandinavia	40
Government (Central & Local)	26	Belgium (Netherlands)	30
Health Care (Pharmaceuticals & Private Sector)	14		
Public Utilities	20		
All Sectors	210	All Countries	210
JOB FUNCTION	NO. OF INTERVIEWS	ESTABLISHMENT SIZE	NO. OF INTERVIEWS
MIS/DP Manager (or equivalent)	210	> 1000 Employees	127
		> 500-1000 Employees	83
All Functions	210	All Establishments	210

EXHIBIT B-2

ANALYSIS OF VENDOR INTERVIEWS

COUNTRY	SOFTWARE PRODUCTS/ PROFESSIONAL SERVICES VENDORS
United Kingdom	20
France	6
West Germany	2
Holland	2
Scandinavia	5
TOTAL	35



Appendix: User Questionnaire

Appendix: User Questionnaire

Bi-Annual INPUT Survey of European Information Services Markets

(Relevant Questions to Systems Development Only)

Systems Development Productivity

QU:1a Over the period 1986-1987 has your applications backlog increased, decreased or remained the same?

Increased

Decreased

Remained the Same

QU:1b IF NOT DECREASED, what are the major constraints on your ability to reduce the applications backlog?

1. _____

2. _____

3. _____

QU:1c IF DECREASED, what are the major factors enabling you to reduce the applications backlog?

1. _____

2. _____

3. _____

QU:2 Do you use or are you planning to use any systems development productivity tools/methodologies/systems in your organisation?

QU:2a Integrated Project Support Environment (IPSE)

USE	PLANNING TO USE	NOT USING
1	2	3

If Use, which system? _____
(e.g. ISTAR, GENOS, PERSPECTIVE KERNAL, MAESTRO)

If not using, why not?

Comments (Probe) _____

_____

_____

QU:2b System Design Methodologies

USE	PLANNING TO USE	NOT USING
1	2	3

If Use, which _____

_____

_____

(e.g. Information Engineering)

If not using, why not?

Comments (Probe) _____

_____

QU:2c Other Development Tools/Techniques/Methodologies

USE	PLANNING TO USE	NOT USING
1	2	3

If Use, which? 1. _____

2. _____

3. _____

(e.g. Computer-Aided Support Environment (CASE), Process Quality Control Tools, Analysis Tools, Artificial Intelligence)

If Use, why and for what purpose/benefits?

Comments (Probe) _____

_____

_____

If not using, why not?

Comments (Probe) _____

_____

_____

QU:2d What do you consider to be the most important features/benefits of the tools/methodologies/techniques you are using?

Comments (Probe) _____

_____

_____

QU:2e What are the most serious problems (concerns) about using these tools/methodologies?

Comments (Probe) _____

_____

QU:2f Fourth Generation Languages

USE	PLANNING TO USE	NOT USING
1	2	3

If Use, which products?

1. _____

2. _____

3. _____

(e.g. FOCUS, IDEAL, MAPPER, LINK, NATURAL, COGNOS)

QU:3 Could you please estimate how much time you have saved (approximate percentage) on particular projects by using productivity tools/techniques/methodologies?

a. Fourth Generation Language _____%

b. System Design Methodology _____%

c. Other _____ _____%

d. Other _____ _____%

e. Other _____ _____%

NB: Interviewers should relate tools/methodologies used for this question to responses given in QU: 2. For example, if the respondent uses an IPSE or a struc tured design methodology include them as prompts in this question.

QU:4 Please rate the effectiveness of the following potential approaches towards improving development productiv ity within your department. (On a scale of 0 to 10, where 0 = irrelevant and 10 = critical.)

READ OUT, ROTATE ORDER.

Approach	Rating
Development Centre	_____
Automation	
- Design Methodologies	_____
- Training	_____
- Project Management	_____
Training	
- Project Management Skills	_____
- Business/Administration Skills	_____
- New Techniques, Tools, Methodologies	_____
Increased Investment in Hardware	_____
Upgrade of Systems Software (e.g. UNIX)	_____
Increased End-User Involvement	_____
Forming Joint Project Teams/Work Units with End-Users	_____
Increasing Awareness of Your Company's Business Problems	_____
Increased Investment in Development Tools	_____

Increased Commitment to Quality _____

Recruiting and Retaining Quality Staff _____

Other, Write in _____ _____

_____ _____

QU:5 Please rate the effectiveness of the following potential approaches towards supporting end-user development productivity. (On a scale of 0 to 10, where 0 = irrele vant and 10 = critical.)

READ OUT, ROTATE ORDER.

Approach	Rating
Information Centre	_____
Increased Connectivity	
- Micro-to-Mainframe Links	_____
- Local Area Networks	_____
Prototyping	_____
Fourth Generation Languages	_____
Training	
- Systems Analysis Skills	_____
- Product Knowledge	_____
Viewing End-Users as Clients	_____
Other, Write in _____	_____
_____ (e.g. Integrated PC Software)	_____

FINANCIAL INFORMATION

QU:11 At this stage I would now like you to give me some brief details covering your expenditure on computer services.

QU:11a Are you anticipating an increase in your total data processing expenditure during 1987?

Yes ☐ No ☐ Don't Know ☐

If yes, by what percentage in comparison with your spending in 1986?

_____%

QU:11b Are you anticipating an increase in your expenditure on external computer services (excluding hardware) during 1987?

Yes ☐ No ☐ Don't Know ☐

If yes, by what percentage in comparison with your spending in 1986?

_____%

QU:12 For the following items that are potentially purchased externally would you please indicate how much of your 1987 expenditure is anticipated to increase in compari son with your spending in 1986?

- Application Software Packages _____%
- Application Software Maintenance _____%

NB: Only ask if response to QU: 11b was yes.

QU:13a Could you please tell me how much you anticipate spending on software productivity tools and systems in 1987?

_____ Amount _____ Local Currency

Don't Know ☐

QU:13b Will this expenditure increase during the next couple of years?

Yes ☐ No ☐ Don't Know ☐

If yes -

QU: 13c By what percentage in comparison with your spending in 1987?

_____%

We appreciate your co-operation in our survey and will send you a copy of the Executive Summary of our research findings.

D

Appendix: Lst of Related INPUT Reports

D

Appendix: List of Related INPUT Reports

European Professional Services Markets - Trends and Opportunties, 1987-1992

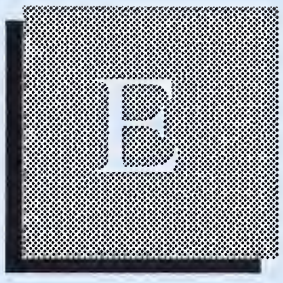
Software Product Pricing and Support - European Market Directions, 1987

Western European Information Services Markets - Industry Analysis and Forecasts, 1987-1992

Software Productivity (U.S.), 1986

European Trends and Opportunities in Fourth Generation Languages, 1985

Future DBMS Markets (U.S.), 1987-1992



Appendix: Key Factors in Software Quality

Appendix: Key Factors in Software Quality
(Translated in French)

- Accessability: ease with which the different components of a soft ware product can be used selectively. This characteristic influences the effectiveness, testability, and ease of use.

- Adaptability: ease with which a software product can be modified, with a view to its adaptation to different working constraints or different usages (modification, remodeling).

- "Banality" of use: ease of possible reuse of a software product, or part thereof, in other applications. This ability depends on the structure of the software in question and on the range of its functions.

- Clarity: combination of the factors which make up the intelligibility of a software package, i.e., conciseness, comprehensibility, readability, modularity, structuration, absence of obscure points, etc.

- Coherence: respect for notation, symbols, uniform terminology or established conventions. This characteristic influences comprehensibility and readability.

- Completeness: the state of a software package, such that its constituent elements are present and are completely written.

- Complexity: measure of the writing difficulty of the software. Different characteristics reflect the simplicity of a programme, i.e., structure of the data, structure of the control graph, the number of calculations to obtain a given result, the size, the effort necessary to demonstrate the correction of the programme, the effort required to modify it, etc. In general, the structural complexity, the textual complexity and the complexity of the dynamic behaviour are distinguished.

- Comprehensibility: ease with which the aims, the hypotheses, the limitations, the data, the results, the components, etc. of a programme can be deduced from reading its text and its internal and external documentation. The concepts of autodocumentation, clarity, complexity, conciseness, and readability all depend on this characteristic.

- Conciseness: absence of useless or redundant elements. This characteristic requires that the programmes are not cut up into too many modules, functions, overlays, or sub-programmes, and do not contain repetitions of pieces of indentical code without defining a subprogramme or a macro instruction.

- Correction: Adherence to the specification, i.e. agreement between all the possible results and the corresponding results indicated in the functional specifications and in the conditions specified in the physical specs (performance). Agreement between the code and the functional specifications.

- 'Attachability': ability for a package to be coupled with others.

- Cost: factors of development cost, cost of implementation, maintenance, and learning related to the software.

- Level of documentation: the quality and quantity of the documentation in its different forms (internal and external documentation, utilisation and programming documentation).

- Efficiency: ability of software to limit itself to the utilisation of the resources (memory, CPU, length of executions) strictly necessary for the accomplishment of its function.

- Extendability: ability of a package to accept the addition of new functions or the extension of its processing capacities.

- Reliability: ability of software to function correctly (functions, precision, performance) for given utilisation conditions. Related to this are the following characteristics: availability, resistance to errors, functional safety, tolerance to faults.

- Integrity: ability of software to be protected against alterations of its code or its data or against access by unauthorised persons. Related to this are confidentiality, security, and functional safety.

- Readability: ease with which the function of the software can be deduced by examining its code. This characteristic is essential for comprehensibility.

- Maintainability: ease with which software can be modified with a view to either correcting its mistakes or extending its abilities. This means that the code must be comprehensible, modifiable, and testable. The following characteristics are related: adaptability, complexity, comprehensibility, conciseness, extendability, readability, repairability, testability.

- Modularity: degree to which a programme can be divided into independent components (modules, sub-programmes, functions, etc.). This degree can be measured by the structural characteristics.

- Resistance to errors: ability to function correctly despite the occurence of conditions which disagree (i.e. are outside their "domain of definition" or are of non-conformal format).

- Stability: ability to function in the presence of perturbing events. Ability to return to a coherent state after a perturbatory event.

- Structure: arrangement of different constituent elements of a program according to a well-defined organisation.

- Time: factors influencing the time required for development, implementation, maintenance, and learning related to a program.

- Testability: ease of elaboration of sets of data, of test, and of verification of the correct functioning of these. This characteristic refers both to test and diagnostic aids and to the effort necessary for the test.

- Tolerance to faults: ability to function correctly in the presence of faults. This supposes that the software has an adequate arrangement and features which permit the detection of faults and the elimination of their effects.

- Transferability: ability to transfer a program from one material to another, or from one operating system to another. The evaluation of this characteristic depends in particular on the control of compatibility (relationships between two environments).