AD-A225 524

DTIC FILE COPY

ESPRIT II

J.F. Blackburn

31 January 1990

Office of Naval Research European Office

90

REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION UNCLASSIFIED		1b RESTRICTIVE MARKINGS			
2a SECURITY CLASSIFICATION AUTHORITY		3 DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited			
2b DECLASSIFICATION/DOWNGRADING SCHEDULE					
4 PERFORMING ORGANIZATION REPORT NUMBER(S) 90-1-R		5 MONITORING ORGANIZATION REPORT NUMBER(S)			
6a NAME OF PERFORMING ORGANIZATION Office of Naval Research European Office	6b OFFICE SYMBOL (If applicable) ONREUR	7a NAME OF MONITORING ORGANIZATION			
6c ADDRESS (City, State, and ZIP Code) Box 39 FPO, NY 09510-0700		7b ADDRESS (City, State, and ZIP Code)			
8a NAME OF FUNDING/SPONSORING ORGANIZATION	8b OFFICE SYMBOL (If applicable)	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER			
8c ADDRESS (City, State, and ZIP Code)		10 SOURCE OF FUNDING NUMBERS			
		PROGRAM ELEMENT NO	PROJECT NO	TASK NO	WORK UNIT ACCESSION NO

11 TITLE (Include Security Classification)

ESPRIT II

12 PERSONAL AUTHOR(S)
J.F. Blackburn

13a TYPE OF REPORT Report	13b TIME COVERED FROM _____ TO _____	14 DATE OF REPORT (Year, Month, Day) 31 January 1990	15 PAGE COUNT 23

16 SUPPLEMENTARY NOTATION

17	COSATI CODES		18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Information Systems Communications Technology
05	02		
25	01		

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

The second 5-year phase of the European Strategic Program for Research and Development in Information Technology (ESPRIT II) began at the start of 1989. (For information on the first phase, see ESNIB 89-03:13-17.) This report will provide more detail on ESPRIT II, which is funded at 3.2 billion European Currency Units; half will come from participating companies.

This report includes a list of all approved projects in: Information Processing Systems, Office and Business Systems, Computer Integrated Manufacturing, Information Exchange System, and Basic Research. A brief description of each of the 47 projects currently supported within the Information Processing Systems area is included.

20 DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED	
22a NAME OF RESPONSIBLE INDIVIDUAL Ms. Connie R. Orendorf	22b TELEPHONE (Include Area Code) (44-1)409-4340	22c OFFICE SYMBOL 310

DD FORM 1473, 84 MAR
83 APR edition may be used until exhausted
All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE

☆ U.S. Government Printing Office 1986—607-044

Contents

ESPRIT II

Introduction

The second 5-year phase of the European Strategic Program for Research and Development in Information Technology (ESPRIT II) began at the start of 1989. (For information on the first phase, see *ESNIB* 89-03:13-17.) This report will provide more detail on ESPRIT II, which is funded at 3.2 billion European Currency Units (ECUs); half will come from participating companies.

This report includes a list of all approved projects in: Information Processing Systems, Office and Business Systems, Computer-Integrated Manufacturing, Information Exchange System, and Basic Research. A brief description of each of the 47 projects currently supported within the Information Processing Systems area is included herein.

Information Processing Systems

The projects in this sector include systems engineering, knowledge engineering, advanced systems architectures, human/computer interfaces, and sensor-based systems. The project acronym and the project number is given in parentheses following the project name.

European Declarative System (EDS, 2025)

The objective is to produce the prototype of an industry technology combining parallel computing with the declarative/symbolic languages used to build large information and knowledge-based systems (KBS). The system will run System Query Language (SQL), Lisp, and Prolog and will be hosted on UNIX-based systems. The work program packages are shown in Table 1.

Table 1. EDS Work Program Packages

- Definition of the execution model to implement the EDS Process Control Language
- Definition of EDS intermediate language set for Lisp, Prolog, SQL, and Hope
- Definition of the EDS machine architecture
- Development of the project software development environment and hardware development tools
- Definition of the full software structure of the declarative languages
- Definition of the kernel and run-time system
- Development of debugging and performance monitoring tools
- Project management and marketing.

by J.F. Blackburn, the London representative of the Commerce Department for industrial assessment in computer science and telecommunications.

EDS applications are shown in Table 2.

Table 2. EDS Applications

- Relational database management to a high bandwidth data store
- Database coprocessor
- Prototypical EDS implementation of a fourth generation language
- Apportionment of components between several chips and circuit boards to meet operational constraints
- Use of EDS Prolog
- Handling large volumes of mixed media data in biomedical and health care applications
- Application of expert systems to the service and utility business areas including banking, finance, and telecommunications.

Metrication and Resource Modeling Aid (MERMAID, 2046)

The aim of MERMAID is to strengthen software project management by improving the provision of information. Tools to estimate interrelationships between the characteristics of a program and the manpower needed for its development, to assess risk levels in program performance, and to monitor the use of manpower in project implementation will be provided for use in the construction of the required information systems.

Hence, the objective of the project is to develop a set of rule-based estimation tools for inclusion in Portable Common Tool Environment (PCTE) and other European industry standard software engineering environments. The project will formulate the relationships between software development productivity and metrics relating to the product characteristics and the development process. The tool set will share a common data model and success will be possible through a variety of user interfaces. The three components of the MERMAID program are: (1) identifying the product characteristics and the development process metrics; (2) developing an estimation model relating product and process characteristics; and (3) developing a set of tools for use in resource estimation and management.

Neurocomputing (PYGMALION, 2059)

The aim is to create an independent European technological base for the applications, algorithms, and software aspects of neurocomputing. The PYGMALION will lead to the coordination of research on connectionist computing techniques and provide the means for developing the necessary software tools for productive research. The objectives are: (1) to provide information

exchange within the European research community; (2) to develop a portable European Neural Network Specification Language and tools to stand as an interface between applications and emulation architectures; and (3) to demonstrate the potential of a neural approach through industrial applications, mainly of image and speech processing.

Pattern recognition and interpretation from remote sensing data, and recognition and classification of workpieces in a factory automation context constitute the image processing application domain. For speech processing, the investigation is centered on the application of efficient learning algorithms on artificial neural network architectures to isolated word recognition in various environments.

Reconfigurable and Extensible Parallel and Distributed Systems (REX, 2080)

The aim is to facilitate the development and management of parallel and distributed systems. The objective is to develop a methodology and a set of integrated support tools.

The strategy is to retain formal specification as the main point for checking system consistency, predicting timing behavior, and assisting in system modifications. The project will develop

● Methodology to construct reconfigurable and extensible distributed and parallel systems
● Tools to support all phases of the distributed application development and maintenance
● Techniques to change running systems and to achieve reliable configuration
● Tools to advise user on how to obtain better performance, based on the separation of the host development and a target operating system.

The main work is specifying distributed and parallel systems, system programming, methodology for system development, analysis and evaluation, dynamic configuration and reconfiguration, run-time support, and demonstrator applications.

Real applications will be used for proving the feasibility and usability of the REX methodology and its associated tools in industrial practice. Two testbed demonstrator applications from industrial automation and from telecommunications systems will be developed in the early phase of the project.

Application of Neural Networks for Industry in Europe (ANNIE, 2092)

The aim is to learn which of several generic problem areas; i.e., pattern recognition, sensor fusion, and adaptive control and optimization, are best approached using neural networks. To do this, the partners will create prototype neural network solutions in simulated problems chosen from automatic inspection, condition monitoring, robotics, control, and scheduling. Specifically, the neural network program will be applied to

● Recognizing and classifying ultrasonic images in nondestructive testing
● Parallel signal processing of acoustic emission from multiple transducers in condition monitoring
● Sensor fusion and planning in mobile robots
● Scheduling the optimal allocation of hospital beds.

The ANNIE work packages and their objectives are shown in Table 3.

Table 3. ANNIE

Work Packages	Objectives
Systematics and capabilities neural networks	Defining useful features of neural and publication of a taxonomy; evaluating the approaches
Problem analysis and criteria for success in generic problem areas	Defining and understanding problem classes; developing criteria and benchmarking for neural networks
Software and hardware for neural network emulating and developing user interface	Evaluating simulators; developing user interface; establishing software standards
Applications	Solving real industrial problems using neural networks; small-scale prototyping and organizing a seminar for industry
Project managing and information disseminating	Organizing an industrial workshop

Integration and Design of Speech Understanding Interfaces (SUNSTAR, 2094)

The objective is to show the benefits and enhancements that human/computer interfaces can offer when they are based on speech input/output (I/O). The project will demonstrate this by realizing prototypes in two fields of speech application that represent market sectors of rapidly growing importance: (1) a professional office-type environment and (2) a public telephone network environment.

The project is application-driven in the sense that it concentrates on the integration of speech functions into demonstrator systems, and not primarily on fundamental research issues of speech recognition and speech output. Dialogue design and the associated ergonomic aspects are highly important to the project, to gain wider acceptability of speech interfaces in real applications. Another key issue in the project is integrating speech technology with other I/O devices.

Adverse Environment Recognition of Speech (ARS, 2101)

The objective is to develop improved algorithms for speech understanding in the presence of noise, and to build a hardware real-time demonstrator. The demonstrator will verify algorithm performance and address the problem of speech-based man/machine dialogue as a system interface in practical applications.

Two application environments have been chosen--car and factory. The system will have a vocabulary in the order of 100-500 words, chosen by each national group of partners and tailored to the specific application environment. Advances will be made in

- Reducing the effects of noise on speech signals in speech preprocesssing
- Feature extracting through pattern matching
- Studying and refining speech algorithms for speech pattern processing in noisy environments
- Dynamic adjusting of the system to user feedback and developing error correction strategy in the human interface
- Developing hardware modules for real-time speech recognition.

The real-time demonstrator is based on a parallel processing architecture attached to a host computer, typically a personal computer. The host, directly accessed by the parallel processor, provides for development support for software algorithms and for fileserver function to the required databases. Performance evaluations will first be made in the laboratory by simulating operating noise conditions and measuring the resulting rate of correct recognition. Performance under field conditions will then be assessed from a prototype fitted in a car. The project will have liaison with DRIVE and other European projects dealing with car applications.

Multi-Language Speech-to-Text and Text-to-Speech System (POLYGLOT, 2104)

The goal is to demonstrate the feasibility of multi-language voice I/O for a number of commercially promising applications. The objective is to integrate lexical, phonetic, and syntactic knowledge common to text-to-speech and speech-to-text conversion, providing greater generality, lower cost, and easier extensions. A set of technologies and working prototypes is provided in Table 4.

Table 4. POLYGLOT Prototypes and Working Technologies

- Very large vocabulary (tens of thousands of words), isolated words, speaker-adaptive speech recognition on a personal computer (PC), for speech-to-text conversion in seven European languages to be implemented and tested. Continuous speech recognition will be addressed as well, but merely up to the stage of producing a strategy of implementation on a PC.
- High quality, language-sensitive, unrestricted vocabulary, text-to-speech conversion in seven European languages.
- Tools for easy extension of both the recognition and synthesis technologies to additional languages.
- Application development tools for recognition and synthesis.
- Development of demonstrators for language teaching, voice feedback for a blind typist, remote access to an electronic mailbox, access to a mail directory, and natural language access to a database.

The project is based on the results of ESPRIT I Project 860. An existing isolated word speech recognition sys-

tem, which was developed by one of the partners, will be extended under this project to six other languages.

An Integrated Modeling Support Environment (IMSE, 2143)

The aim of IMSE is to facilitate using systems engineering methods throughout the design cycle by developing a support environment for performance modeling. The related goals are to improve the integration of currently available modeling tools with one another and with the design process, and to extend their scope to cope with new systems architectures and increasingly complex designs.

These goals are supported by the project objective, which is to provide the means whereby performance modeling tools and techniques can be used effectively to support systems design and development. Development work will be focused on developing support environment, integrating tools, and extending tool functionality.

Among the features of the new environment will be advanced graphics workstations, an extensible toolset supporting alternative performance modeling paradigms, queue networks, process-interaction, and Petri-nets. Also featured is integrating tools via a common object management system; graphics support and reporting facilities; and a structured system specification. Integration with systems engineering environments is an important consideration.

The toolset will be targeted on the applications areas of computer systems architecture, real-time systems, and distributed systems; there will also be evaluation exercises in the area of network systems, computer-integrated manufacturing, and information systems. Research work will be concerned with automating the modeling process, deriving performance models from system descriptions, and developing new modeling techniques for existing and new architectures.

Validation Methods and Tools for Knowledge-Based Systems (VALID, 2148)

This program aims to advance the ability to verify and validate KBS. The objective is to develop a toolbox, or general logical model, within which the verification and validation concepts can be defined and analyzed. The methodologies are shown in Table 5.

Table 5. VALID Methodologies

- Definition of a KBS quality plan, with special attention to validation actions during the different phases of KBS life cycles
- General logical model in which verification and validation issues may be defined
- Common conceptual representation in which most KBS can be expressed
- Definition of a validation metalanguage (VETA) to provide the abstract mechanism necessary to build verification and validation tools.

These will provide the basis for several software products--implementation of VETA, set of representative tools for validation and verification, interfaces to a set of existing KBSs. The work program is organized around four work packages--defining validation models, implementing, evaluating, and managing.

Software Certification Program in Europe (SCOPE, 2151)

The aim is to support supplier/customer relationships throughout the internal market by facilitating product certification. Product certification will assist customers in the selection of information technology (IT) products and clarify the legal position of suppliers. For this to happen, a strengthened certification procedure must be established.

The objective is to demonstrate the feasibility of a European software certification procedure. The software will define, experiment with, and validate an economic European software certification procedure applicable to all types of software acceptable and legally recognized throughout Europe. The goals are shown in Table 6.

Table 6. SCOPE Goals

- Define procedures to evaluate software compliance with a specified set of attributes, enabling a seal of approval to be granted
- Develop new, efficient, and cost-effective certification technologies for approving the award of this seal of approval
- Promote using modern software engineering technologies to be used during the software development and contributing to the delivery of the seal

---

The project will apply certification procedures that will be developed by their definition by a panel of experts, experimental validation on real industrial case studies, evaluation of the results followed by modification or release of the procedures. The different certification procedures will be evaluated on various kinds of software through numerous experiments. A sophisticated database system will be provided for the exchange and analysis of data. The project will be implemented through two work packages--definition and feasibility demonstration phase and main experimentation and evaluation phase.

Visual Inspection and Evaluation of Wide Area Scenes (VIEWS, 2152)

The objective is to demonstrate the feasibility of a computer vision system for real-time surveillance of outdoor scenes that comprise moving objects in a known, wide-area, and structured environment. The package addresses issues in the field of computer vision related to situation representation and reasoning and real-time control of various asynchronously arriving data streams. The

VIEWS will also include the provision of support tools for acquiring the scene and object knowledge.

The VIEWS system prototype will be integrated into a generic architecture from several different components developed by different partners in the project, and will be demonstrated in two areas of application: (1) surveillance of ground traffic in airports, as an assistant to the ground movement controller; and (2) surveillance of vehicle traffic on roads and motorways for traffic control purposes. These demonstrations will be developed and assessed in close cooperation with potential end users (airport and road authorities).

Machine Learning Toolbox (MLT, 2154)

The project will build a basis for the commercial use of machine learning techniques. This will be done in three ways:

1. A toolbox of different learning algorithms will be adapted and developed. The algorithm will cover a broad range of applications and will be made robust enough to stand industrial use. The delivery hardware and software will comply with industry standards.

2. An advisory consultant for the system (MLT consultant) will be developed. The consultant will allow users unfamiliar with machine learning to incorporate this technology into their application.

3. A series of applications will be used to guide development of the above components. Evaluation and implementation methodologies will be developed from the experience gained, which will be in the field of vision and network maintenance.

Shipboard Installation of Knowledge-Based Systems: Design and Installation (KBS-SHIP, 2163)

The objective is to develop design concepts to implement and use advanced integrated IT systems in ships. The project provides a decision support system to safely and economically operate a ship and efficiently handle complex equipment. The project also provides a framework for the integration of data communication and information systems in ships.

In earlier design specification phases of the project, performed in ESPRIT I (Project 1074), concepts were prepared for the overall architecture for the KBS-SHIP system. In this phase, the architecture will be further developed and implemented.

The viability of the concept will be ensured by building a prototype KBS-SHIP system incorporating a limited number of expert systems. This will include one expert system managing the communication and collaboration among several task-solving expert systems operating on a common database structure. The task-solving expert systems have been selected because of their potential contribution to cost savings and improved safety in ship operations. They will be decision support tools to optimize voyage plans, prepare maintenance schedules,

handle alarms, and make loading plans. Each will be supported by a system for applying ship operational regulations.

The project will focus on methods for resolving conflicts between cooperating expert systems integrated and demonstrated in a suitable land-based facility. Methods for validating and assessing system performance will be developed and used in connection with the integration. The project will exploit the results of earlier work. Two of the expert systems will be evaluated in a real environment. Krupp-Atlas Electronik will develop the Voyage Pilot Expert System (a navigation system), and Soeren T. Lyngsoe will develop an expert machinery operation system.

An Intelligent Real-Time System for Signal Understanding (AITRAS, 2167)

The objective is to design and develop a knowledge-based system for signal understanding. The AITRAS is to comprise a system shell and the design tools for building systems for the real-time analysis and interpretation of instrumentation signals.

Though sophisticated real-time techniques have been applied successfully in this field, their success has been limited for several reasons. The huge amount of data to be processed slows the response; the experts needed to interpret the results are in short supply; and even the use of KBS to emulate expert reasoning does not reduce the delays sufficiently. In attempting to reduce these shortcomings, the intention is to design AITRAS around a real-time inference engine (RTIE), tightly coupled to a signal-processing subsystem, and to design and develop an architecture for real-time environments.

Subsequent validation of the AITRAS system will be by three industrial applications:

1. Examine nondestructively steam generator tubes in nuclear plants by analysis of eddy current signal

2. Use same technique for heat exchangers and condensors

3. Predict faults by the analysis of vibration signals for the maintenance of bearings.

The work program for each application begins with eliciting expert knowledge and creating the system specification. An initial prototype is then demonstrated. Static and dynamic validations are then made and the performance is refined within an operational environment.

Generation of Interactive Programming Environments II (GIPE, 2177)

This project will use a generic interactive environment as a basis for work in designing, implementing, and experimenting with real-size environments for industrial applications and advanced research in the area of interactive environments based on formal specifications.

The project starts with the interactive programming environment generator (an outcome of ESPRIT I Project 348 GIPE). This system uses as input the complete formal description of a programming language and produces a specific environment for that language. The resulting environment includes an editor, an interpreter/debugger, and other tools, all of which have uniform graphics man/machine interfaces.

The main result of the GIPE project has been to demonstrate that the technology is feasible. Its follow-up--GIPE II--aims to make it mature by providing support for constructing large formal language definitions, extending the functionality and performance of the generated environments, and demonstrating the approach in several selected industrial applications. For instance, GIPE II could construct a development environment for scientific computing and the development of an environment for the LOTOS specification language.

Speech Understanding and Dialogue (SUNDIAL, 2218)

The SUNDIAL addresses the problem of speech-based cooperative dialogue as an interface for computer applications in the information services domain. The main technology to be developed will be continuous speech recognizing and understanding, and oral dialogue modeling and managing.

Speech input will be sentences of naturally spoken utterances of good quality in the first phase of the project, and standard pulse code modulation (PCM) "telephone quality" speech in the final phase. The recognition vocabulary will be about 1,000-2,000 words for each of the applications. The grammar will be a subset of the four partner's languages (English, German, French, and Italian). While the project will begin with speaker-dependent recognition, the objective for the second phase is to achieve automatic on-line speaker adaptation. The dialogue manager will allow users to express themselves in a restricted natural language.

Prototypes will demonstrate the technology for two applications--voice mail servers allowing access to an electronic mail system, and telephone access to a public enquiry system; e.g., for tourist information. Each demonstration system will be evaluated through extensive user trials. For all demonstrators, the project will define a common general architecture, common formalisms for grammar representation across languages, and common semantic representations for dialogue management and message generation.

Definition and Design of an Open Dependable Distributed System Architecture (DELTA-4, 2252)

Based on an existing ESPRIT project, the aim is to formulate, develop, and demonstrate an open fault-tolerant distributed system architecture. The proposed project has direct applicability to wide application areas which include computer-integrated manufacturing, office systems, integrated information processing systems, and process control systems. As an open system DELTA-4 has three important properties:

1. Implementation based on off-the-shelf heterogeneous computer systems is possible; the fault-tolerant properties of the architecture are not dependent on the use of proprietary, fail-safe, or self-checking processors.

2. The architecture conforms to the OSI model and implementations are able to coexist and interwork with systems communicating by using current standard ISO/OSI protocols.

3. The dependability and distribution properties of the architecture are offered to the user in a transparent way. Both incremental dependability and incremental performance are offered on a service-by-service basis.

The proposed project has both a generic component and an implementation component. The generic component is already contributing to basic concepts in the areas of dependability, multi-point communications, system administration and open distributed processing; these concepts are valuable to other ESPRIT projects and to the standards community. The implementation component demonstrates the validity of the concepts using an experimental site, and pilot sites should establish their correctness.

Tools for Processing Multi-Sensorial Signals for Plant Monitoring and Control (TOPMUSS, 2255)

The aim is to facilitate the development of systems for multi-sensoral signal processing for advanced process monitoring and technical systems' diagnosing. The approach is to provide the means for evaluating the feasibility of evaluating a multi-sensoral processing task and for swift and economic prototyping.

Research will be centered around the acquisition and representation of knowledge, and on signal and information processing techniques. The development objective is to provide a toolbox for assisting the development of run-time systems for plant monitoring and diagnosing. The work will be concentrated in three main fields:

1. Developing techniques and methods for the acquiring and representing knowledge about plants, signal processing, and context information

2. Developing knowledge-based signal and information processsing techniques for heterogeneous sensor channels

3. Developing an open software architecture configurable to particular domains.

The goal is to design a software system for the construction of fault monitoring and diagnosis systems, containing all the tools developed in the course of the project.

Architecture for Cooperative Heterogeneous Online Systems (ARCHON, 2256)

This proposed project aims at developing an architecture for cooperative expert systems for industrial applications. This development will take place in two phases and several steps. The first phase will be creating a development environment, and the second phase will be its application to two large-scale demonstrators. The first step concerns developing the basic concepts of cooperation between autonomous and semiautonomous systems.

These concepts, together with those concerning possible interactions (in the sense of protocols or messages), will feed into the computational model of a virtual machine. The virtual machine will be realized in extensive prototyping testbed environments of large-scale industrial applications. The development environment around the virtual machine will provide appropriate tools to port any prototype developed on this machine onto existing run-time environments. These run-time environments may be of very different natures, ranging from simple processor machines via multiprocessor environments to distributed systems consisting of different hardware and software.

New Architectures for Optical Processing in Industrial Applications (NAOPIA, 2288)

The feasibility of optical processors for object recognition has been successfully demonstrated within the ESPRIT I projects 534 and 1035. The goal is to integrate the latest device and material developments with these new architectures in practical processors to use industrial inspection, quality control, associative classification, and signal processing. The first phase is a definition phase in which different architectures in optical processing will be considered to fully exploit the technology developed in Projects 534 and 1035 towards new applications.

Despite being extremely powerful, optical systems often lack the flexibility of electronic systems. Consequently, it will be necessary to also consider hybrid optical/electronic approaches for multi-layer type architecture implementations. To achieve this goal, a comparison of optical processing schemes with device technology will be accomplished and the impact of non-linear optical devices will be analyzed.

At the end of the first phase, the question of whether or not to implement practical demonstrators for the selected applications will be decided. If the evaluation is positive, the program will continue and demonstrators will be produced and tested.

The Development of a Methodology for Specifying Nonfunctional Requirements (ORDIT, 2301)

The specification of information technology systems gives emphasis to functional requirements. Most specification methods have difficulty in making formal and explicit variety of nonfunctional requirements that may be critical if the system is to be acceptable in an organizational environment. These requirements may be on behalf of the whole user organization; e.g., the requirement for dependability, flexibility, and compatibility. They may also arise from the needs of end-users for systems that are usable and acceptable, which do not threaten privacy, job satisfaction, and health. Studies of the IT uptake process

often demonstrate that it is these factors that prevent implementation or encourage disuse.

The aim of this project is to produce a computer-based methodology that will enable design teams to make explicit the nonfunctional requirements for their product or system. The methodology would use concepts at the organizational and workrole levels of description to represent the requirements and will contain a database of standards, models, and knowledge about the characteristics and needs of different classes of computer user. The methodology will also facilitate the testing of proposed technical solutions for matching with organizational requirements and for their implications within the organizational environment.

The construction of their methodology will be by a user-centered, iterative development process. The participants will be organizational theorists and human factor specialist working with system software specialists. Early prototypes of the core of the methodology will be produced, which will be subjected to user testing. The methodology will thereafter undergo several rounds of modification, extension, and progressively more rigorous and realistic user testing. This process is to ensure that the resultant methodology not only meets the functional needs of design teams, but also meets their nonfunctional requirements for acceptable and usable tools.

Lotosphere (LOTOSPHERE, 2304)

The aim of LOTOSPHERE is exploitation by industry of mathematically sound formal design techniques. This will be achieved by product engineering and by using a viable, fully tool-supported, formal system design and development methodology.

The methodology is centered on the emerging international standard Formal Description Technique (FDT) LOTOS (IS8807). The project intends to convert LOTOS into an industrial tool applicable to system design and system implementation. This includes the development of design structuring techniques, integrity preserving transformations, a test theory and methods, language enhancements, and an integrated toolset. These methods and tools will be applied to industrial pilot specifications and implementations.

Until now FDTs have been primarily developed and used with the objective of reporting the architectural design and specification phase. The aim has been the development of correct, implementation-independent specifications that faithfully reflect an architecture. The LOTOS is a particularly flexible and expressive FDT that can support a variety of specification types (object-oriented, constraint-oriented). This makes LOTOS a natural and appropriate foundation on which to base high-quality software engineering of concurrent and distributed systems. Such a LOTOS environment will support the entire implementation path, thus permitting the rapid development of correct, high-quality implementations from architectural specifications.

Multi-Sensor Image Processing (MUSIP, 2316)

The objective of MUSIP is to develop general purpose tools for the implementation of multi-sensory imagery and the fusion of image information with nonimage data from many other sources. The MUSIP system will provide an integrated environment for the interpretation of image and spatial data in conjunction with other disparate data and knowledge.

Existing and novel image processing tools will be integrated into a knowledge-based environment with powerful data handling facilities. New tools will be developed for sensor fusion, for the incorporation of auxiliary knowledge, and for knowledge-based control structures.

Two demonstrators will be produced: (1) for application in flood and deforestation monitoring in remote sensing, and (2) for medical imaging. The data to be interpreted will include spatial models (digital maps or anatomical atlas), spectral databases, and models (radar cross-section models or a tissue response database), and applications facilities will be provided for image interpretation on a powerful workstation with transputer-based accelerators.

Demonstration of Advanced Reliability Techniques for Safety-Related Computer Systems (DARTS, 2354)

The aim is to facilitate the selection of reliable systems for safety-critical applications. This will be accomplished by moving towards establishing a single, universally agreed method of certifying computer-based, safety-critical systems and by reporting the cost effectiveness of various approaches to software reliability measurements. Accordingly, the objective is to demonstrate using of advanced reliability techniques to select safety-critical computer systems. Table 7 shows the main stages of the project.

Table 7. DARTS Main Stages

- Provide cost-effectiveness measurements of currently available methods and tools for creating and assessing software systems for safety-related applications from industrial trials
- Prepare guidelines to produce and assess computer-based systems for use in safety-critical applications
- Evaluate the level of reliability achieved by combining several diversely produced channels in one configuration. The results of REQUEST (project 300) will be used in the evaluation
- Provide a benchmark environment as a basis for a service to European enterprises involved in the development of safety-critical systems

---

The DARTS is based on the design and licensing of a four-channel, safety-critical, computer-based system. The development of the four diverse software versions will focus on a variety of methodologies, each using different tools and techniques, and each originating from

different project teams. Each of four teams will design one of the channels—one team will design the test environment and two teams will act as licensing bodies. Once a license has been granted, the prototype system will be installed in the test environment. An evaluation of comparative channel performance and of the selection process will be provided by monitoring system performance.

Metrics Education Toolkit (METKIT, 2384)

The aim is to accelerate transfering results on software metrics into the industrial workplace by providing enhanced tools to educate current and future software engineers, project managers, technicians, and other practitioners.

The objective of METKIT is to produce a metrics educational toolkit capitalizing on the results of ESPRIT and national projects. The toolkit will include computer-aided learning packages as well as a wide range of other educational approaches. The toolkit will be evaluated and exploited by the partners, and also will be offered to a wide circle of users across Europe throughout the project. Table 8 provides the main stages in the project.

Table 8. METKIT Main Stages

- Identify industrial and academic needs in the application of software quality and productivity metrics, and determine the associated training requirements
- Rationalize the existing knowledge in metrics and institute mechanisms to exploit new advances
- Identify educational technology suitable for use in metrics training activities
- Design and implement courseware and computer-based tools to teach software quality and productivity metrics
- Evaluate the academic and industrial use of the tools and enhance them accordingly

Process Operators Multimedia Intelligent Support Environment (PROMISE, 2397)

The aim is to promote safer and more efficient real-time operator action in process control. This will be achieved by providing KBS multimedia interfaces. These interfaces will handle complexity, be operator-centered, and will respond effectively to rare or unforeseen situations. The objective is to develop the techniques and tools to create such interfaces.

The approach will be user-centered and the ensuing changes in system performance will be evaluated. Presentation mechanisms will assist knowledge-based designers in visualizing the reasoning processes of the intelligent systems under construction. The project will use the Expert System Builder from Project 96, and will draw on the knowledge-based dialogue for process control of GRADIENT (Project 857). The approach will be tested through frontending intelligent KBS applications to an existing system in corrosion monitoring and to large new control systems in power generation and chemical production.

Environment for Qualitative Temporal Reasoning (EQUATOR, 2409)

The objective is to bring together existing techniques for reasoning about the behavior of process-based systems in time, to develop and extend these techniques as necessary, and to make them available for use in constructing large-scale KBSs in several industrial and commercial application areas that require them for their future development.

The proposed project will deliver a software development environment for time-dependent applications within which a system builder may find the reasoning mechanism(s) appropriate to his problem, apply them singly or in combination, and appropriately tune them to function within applicable runtime operational constraints. This environment will be designed to meet the needs of several specified applications selected for their diversity and economic importance, and will be evaluated by reimplementing two of these as demonstrators.

Advanced Knowledge-Based Environments for Database Systems (KIWIS, 2424)

The purpose is to design and develop the KIWIS industrial prototype of a KBS to support sophisticated applications requiring complex operations on data and knowledge, possibly located in other systems. The KIWIS will provide an integrated knowledge representation language and programming environment for user modification, and establish an open environment with extensible, friendly, graphical interfaces to distributed databases and knowledge bases.

The KIWIS system will connect to external, traditional database systems, and give a complex-object view of their data. The system will also establish a cooperating network environment consisting of distributed KBS by enabling an intelligent dialogue with other KIWIS systems.

KIWIS will draw on the results of ESPRIT I KIWI (Project 1117). The work packages are concerned with the specification, design, implementation, and testing of the components shown in Table 9.

Table 9. ESPRIT I KIWI Work-Package Components

- Interface to external databases for data integration
- Object virtual machine, providing an environment for the manipulation of complex objects
- Basic language machine, which adds functionality to the object virtual machine
- Abstraction layer for supporting the knowledge representation language
- Knowledge-base view module for the user interface and the cooperation manager
- User interface to exploit all the capabilities of the knowledge-base language
- Cooperation manager, which uses metalanguage to extract information from other KIWIS systems

Transparent Object-Oriented Parallel Information Computing Systems (TROPICS, 2427)

The aim is to ensure the exploitation of the next generation of data processing systems by reducing execution time for a wide range of applications through the use of parallelism, reducing cost of computation through the exploitation of very large scale integration (VLSI) technology, and cost of producing computer software by the developing methods and tools.

The project will research and develop a symbolic parallel computer system technology to meet the high performance demands of symbolic applications that constitute the next step forward in office automation and in other fields. This will be expressed in the delivery of an industrial prototype of a parallel processing system, using the object-oriented approach for exploiting parallelism. The work package functions are shown in Table 10.

Table 10. Functions of TROPICS Work Packages

- Defining the architecture of the processing node and its constituent components, followed by VLSI designs and implementations for the components
- Designing and implementing the parallel processing units and integrating them into a complete TROPICS system
- Designing and implementing the TROPICS operating system
- Designing and implementing support software to provide a platform on which applications can be developed
- Making available the object-oriented programming language POOL as the TROPICS main programming language; other languages will offer a migration path and a set of software development tool
- Realizing application software targeted for office environments on the TROPICS system
- Facilitating the transfer of results and the investigation of the application domain by transfer desk
- Providing proper management.

Demonstrator and industrial prototypes of a workstation will be developed to serve as a user interface to the system. Multimedia, cartographic, multilingual, and knowledge support applications will be made in the office area.

Extensible KBMS for Large Knowledge-Base Application (STRETCH, 2443)

The objective is to design, implement, and experiment with a system supporting the representation and manipulation of knowledge bases of large size. Storage of persistent objects will be provided by a disk-resident database that will support nontraditional data structures and operations for knowledge representation and management. Moreover, the system will exploit large central memories of recent workstations.

A major effort is to provide a physical and conceptual object manager to store and manipulate knowledge and data in an integrated manner. The system will allow extendibility of typing, operations, access methods, and concurrency control. The object manager will support several linguistic paradigms:

- Rule-based language defined by extending current logic languages to include complex objects, negation null values, and update facilities
- Object-oriented language, based on hierarchies of abstract data types, integrating declarative and procedural knowledge
- SQL-based language to enable the continuity of relational applications in the new environment.

These three languages will constitute a multi-paradigm programming environment to accommodate several applications. Early experimentation with the proposed rule-based and object-oriented linguistic styles will be obtained by rapid prototyping. The conceptual object manager will provide an internal notation, suitable as a pivot for all language processors.

A European, Distributed Memory, Parallel Supercomputer for Numerical Applications (GENESIS, 2447)

The objective is to develop a highly parallel architecture for very high-performance numerical computing. The GENESIS architecture is a concept for a family of supercomputers meeting the goals indicated in Table 11.

Table 11. GENESIS Architecture Goals

- Highest possible performance through a highly parallel MIMD-SIMD architecture with distributed memory
- Optimum cost-effectiveness by exploiting the latest advances of VLSI technology
- Wide range scalability and extendibility of the system
- Long product life defining the abstract machine independent of the current state of technology
- High reliability by an optimal degree of fault tolerance
- Highly parallel architecture exploitation by providing a programming environment.

The specific goals of the definition phase are to define all aspects of GENESIS; the goals are shown in Table 12.

Table 12. GENESIS Definition Phase Goals

- Specify a complete architecture, covering node architecture, interconnection, operating system, languages and compilers, user interface and peripherals
- Make a preliminary specification of the consequential system requirements
- Specify performance goals and demonstrate their feasibility of achievement
- Plan and specify the development of application software
- Define the organizational structure of the main phase
- Initiate the development of the next generation of components
- Define the marketing options and the organization structure for exploiting project results.

Integrating Database Technology, Rule-Based Systems and Temporal Reasoning for Effective Software (TEMPORA, 2469)

The aim is to support the development of reliable, flexible, and maintainable business information systems. The objective is to implement a paradigm for specifying a business system by a knowledge base of rules. Another objective is to extend the implementation by incorporating a commercially available database management system for industrial use and mechanisms for explicitly modeling temporal relationships. The project will build on such components of current technology as relational databases, rule-based modeling, and logic programming.

Parallel Execution of Prolog on Multiprocessor Architectures (PEPMA, 2471)

The objective is to refine and complete the design of an existing parallel execution model for Prolog for possible industrial exploitation on the range of shared-memory machines available on the market. In addition, the program aims to design two alternative execution models to run on message-passing machines designed from the outset to be transputer based.

One model will implement a combination of OR-parallelism with dependent AND-parallelism via a shared virtual address space, while another model will implement OR-parallelism and possibly AND-parallelism via communicating processes. The models will be compared on the basis of how well they perform on large-scale applications.

Situation studies will be made to investigate how the software models can best be supported by hardware, with particular reference to architectures based on the transputer or similar principles.

Finally, an extension of Prolog will be defined and implemented that further exploits the potential of these parallel models. The ultimate aim is to have a language and implementation that combines the capabilities of the existing classes of logic programming languages.

A Multi-Modal Interface for Man-Machine Interaction with Knowledge-Based Systems (MMI2, 2474)

The MMI2 proposal will undertake research and development (R&D) into multimodel interfaces to KBS. The interface will allow the user to communicate with the underlying application KBS using natural language, direct manipulation and gesture, as well as a command language. The application is allowed to communicate with the user through natural language and structured graphics. Natural language modules will be developed for English, French, and Spanish. The English system will be based on previous results and prototypes obtained in an ESPRIT I project. In order to maximize its readability and efficiency and favor its use even in real-time situations, the ergonomic aspects of the interface will receive special attention.

The interface will incorporate a dialogue management and mode selection system using knowledge of the specifics of individual modes, the context of previous interactions, and the application domain to determine the content of system output, selecting the most appropriate mode in which to present particular information. A user modeling module will interact with dialogue management so that the system will react appropriately to different classes of users and individual users.

The interface developed will be designed to be portable across a range of potential applications of Prolog-based KBS. The project will focus on real, practical problems and will be demonstrated and evaluated using a specific application, an expert system in computer network design. This application will also be developed inside the project based on preliminary studies and exploratory prototypes already performed and developed by the lead partner. Two demonstrator prototypes of the interface integrated with the application will be produced after 2 and 5 years.

Maintenance, Validation, and Documentation of Software Systems (REDO, 2487)

The aim is to assist software engineers in maintaining, restructuring, and validating large software systems and their transportation between different environments. The objective is to articulate a theoretical framework for doing this and to develop the required methods and prototype tools.

The approach will be to integrate artificial intelligence techniques with current developments in the field of formal methods, software engineering, software validation, and human factors. The work will be structured as follows: problem definition, domain specific prototype applications, R&D in maintenance and validation, application of knowledge-based systems, toolkit construction and integration, and evaluation.

Variable Object Identification, Location, and Acquisition (VOILA, 2502)

The aim is to develop the basis for advancing computer vision applications in a range of tasks in varied industrial environments. The project will extend the capability of systems for handling environmental, object, and task complexity and will develop system intelligence. The objective is to demonstrate the applicability of a flexible, robust, high-level vision system to an autonomous guided vehicle operating; e.g., in a factory, stockyard, or on a road. The intent is to develop the concepts and techniques required for rapid, reliable, and efficient object identification, location and acquisition; a flexible, high-level vision system; and demonstrate its applicability.

Meeting these objectives will require significant advances in the state of the art in the fields of early vision and the use of stereo and motion; in using and exploiting of predictive feed-forward and dynamic vision; in the range and type of models that can be used in a vision system; in the design of the vision system architecture and

its control structures; and in the implementation of vision systems in a multiprocessor environment. The VOILA targets are shown in Table 13.

Table 13. VOILA Targets

- To develop a dynamic vision system capable of continuous, on-going information integration in a changing environment
- To be able to handle a wide variety of scenes, objects, and shapes
- To develop a vision system capable of focusing and tracking objects
- To develop an integrated internal model or knowledge base of the system's capabilities
- To develop a vision system capable of reasoning about the geometry, location, and motions of objects in the environment
- To explore parallel implementations of the system using communicating dynamic processes.

Operating Systems and Programming Environments for Parallel Computers (SUPERNODE II, 2528)

The proposed project aims to design and develop appropriate operating systems and environments for general purpose parallel computers, for a range of tasks including run-time support, program development, multi-user support, and real-time applications. To do this, it will use and enhance an existing parallel computer architecture--Supernode--developed under ESPRIT 1 and its existing software base. Additional software components, which will be thoroughly integrated through clearly defined interfaces, will be based primarily on existing environments, tools, and languages with modifications for parallel systems. Specific tools and libraries required for effective use of parallel architectures will be designed and implemented.

A primitive executive kernel, resident on each processing element in the system, will be developed. Several operating systems and environments will be constructed on top of the primitive executive, including an X/OPEN compatible system and the Portable Common Tool Environment (PCTE). An Integrated Project Support Environment (ECLIPSE) will be ported onto a version of PCTE running on the parallel machine; this will be subsequently parallel.

Software tools will be developed that will readily interface to a parallel version of the PCTE. These will include compilers for widely used languages; e.g., Fortran 77, C, and SQL. Libraries of numerical routines will be written that are suitable for parallel computer architectures with various numbers of processors.

Whenever possible, the higher-level components will be designed so as to be architecture independent. Four applications (1) image synthesis, (2) simulation of heterogeneous systems, (3) computer-aided design (CAD) for VLSI, and (4) oil reservoir simulation will also be

implemented to provide a test bed for the major software components.

By capitalizing on some of the most successful work on parallel architectures achieved in ESPRIT I, this project has an extremely high potential in terms of industrial exploitation. The Supernode machine is already marketed by two companies--Parsys in the U.K. and Telmat in France--but the near-future availability of a large range of basic software running on this equipment will undoubtedly boost its market penetration. The market segment currently addressed is mostly scientific applications (with Fortran, Ada, and numerical libraries to be made available). But the machine is flexible enough to cover multi-purpose needs (hence the plan to offer PCTE and an SQL front end).

Incremental Construction and Reuse of Requirements Specifications (ICARUS, 2537)

Requirements Engineering (RE) is the activity of investigating the customer's needs in the context of a software development project. The specification of requirements, which describes the functional and non-functional properties of the system and its environment, can be distinguished from the specification of the design that describes the system alone for the benefit of software engineers. This proposed research is concerned with the study of formal methods for building requirement's specifications. Three levels of analysis can be identified in the study of such methods: (1) specification product - description of the desired system and its environment; (2) specification process - set of activities by which the specification is produced; and (3) specification rationale - set of reasons that led to the choice of a particular process.

The project aims to study each of these levels. The prerequisites for Study of Requirements are provided in Table 14.

Table 14. Prerequisites for Study of Requirements Engineering Methods

---

- Practically, as methodological concerns in RE have not so far been addressed, investigations are restricted to functional and performance; e.g., real-time requirements
- Study of real practice will be undertaken to get insights into RE process and rationales
- Formal techniques for supporting methods must be defined, which requires a formal system for modeling specification products, processes, and rationales.

Putting RE methods into practice will be ensured by developing an integrated RE environment prototype actively supporting such methods, and by performing realistic case studies. The environment will be based on a process-driven RE integrating active analyst guidance, consistency and completeness checking, specification visualization, and prototype generation. A qualitative assessment of the formal concepts and tools developed will

be achieved by performing case studies on selected families of real applications. Industrialization will proceed incrementally, according to the particular procedures of each participant.

Atmosphere (ATMOSPHERE, 2565)

This project aims to develop standardized facilities supporting systems engineering activities. The facilities to be developed comprise a framework and associated integrated tools and services. These form a basis for the development of the application specific, integrated systems engineering environments. These environments are to be constructed in turn from integrated tool sets hosted on the framework and will provide engineering management functions in addition to the systems engineering functions relevant to the application domain.

The project views and addresses systems engineering as composed of at least three stages: (1) a technologically independent stage concerned with developing the system requirement and developing high-level functionality; (2) a stage concerned with transforming the functionality into technologically dependent implementable designs; and (3) a stage concerned with integrating the application system components and maintaining them through configuration and version control. The project is targeted to support systems engineering for applications in the information technology field, covering switching systems, manufacturing systems, office administrative systems, and embedded systems.

Throughout ATMOSPHERE, a standards-based approach is to be adopted. This approach is bidirectional in that existing standards--international, national, and industrial--will be adopted in the development of the ATMOSPHERE framework, tools, and services, and their interfaces. Conversely, it is expected that through exploitation and marketing activities, these interface standards will be adapted to enable them to encompass a wider range of strongly related technologies.

In addition to being standards-based, the project follows a technology integration approach by bringing together and re-engineering only where necessary, many of the tools, methods, techniques, and environments developed within the European and other international information projects. The project team recognizes the complex issue of large application domain, systems engineering project management, and the relevance of this complexity to its own highly distributed and diverse functionality. For these reasons, a strong project management structure is followed. This structure is also required since the project is to draw extensively on the results and to integrate and coordinate the outputs of several international and ESPRIT projects.

Maintenance Capability for Software (MACS, 2570)

The MACS intends to support software maintainers in error diagnosing and debugging, program porting, improving performance and enhancing and evolving sys-

tems. The MACS will do so by providing a system for assisting the analyzing of applications. The project objective is to define and implement a prototype system to assist in software maintenance. Design graphs will be incorporated in the system to meet the software maintainer's requirements for information concerning applications design decisions and their intent. Application designs will be portrayed at different levels of abstraction to provide structured information to use in software maintenance as seen in Table 15.

Table 15. Maintenance Capability for Software Work Program

● Developing a knowledge representation system for software design graph recording and use

● Developing application and program design analysers for design graph construction

● Applying human factors principles to the system conceptualization and to the design of the man-machine interface

● Developing a prototype software maintenance assistance system

● Integrating both the knowledge representation and the software evolution expert system with a configuration management system

● Validating experimentation to existing software applications.

The project started in January 1989, and the first review was held in July 1989. Available documents indicated progress in all the main work packages. The combination of knowledge and software engineering skills in the project gives some promise of useful results.

Acquisition of Knowledge (ACKNOWLEDGE, 2576)

The aim is to improve the efficiency of the process of acquiring knowledge at all phases of KBS development. The objective is to construct a knowledge engineering workbench (KEW) that will provide an integrated environment for knowledge elicitation and acquisition. Methods for integrating several knowledge acquisition (KA) techniques will be developed and an environment that integrates a number of existing automated and non-automated techniques will be constructed.

The key constituents of this workbench are machine learning, knowledge elicitation, and dialogue techniques. Learning mechanisms will support automatic deduction and organization of new knowledge after new information is provided or a particular event observed. This capability is necessary to guide the decisions and reactions of the environment. Elicitation methods will be implemented in the environment and will thus define the protocols of interaction with the system. Mixed initiative dialogue techniques, based on the recognition of intentions and the visualization of knowledge bases, will provide the foundations for integrating learning mechanisms and elicitation methods in the same environment. The project will also

develop comprehensive framework for guidance at all stages in the KBS development.

The ACKNOWLEDGE starts by placing a selection of problem diagnosis and/or classification techniques in a common framework in order to produce a coherent toolkit for experimentation (by the end of the first year). The KEW will then investigate potential integrations and synergies between KA techniques and will deliver a first prototype of the KEW by the end of the third year. The KEW will be validated by several jointly developed demonstrator applications.

Multilingual Speech Input/Output Assessment, Methodology, and Standardization (SAM, 2589)

The objective is to develop methodologies, tools, and databases for the assessment of speech systems in contexts where multilingual performance is required for the same basic equipment. The consortium is broad, with participants from six European Community and two European Free Trade Association member states. The project is to provide techniques for assessing speech synthesizers and recognizers for at least the eight languages of the participating countries.

The participation of a large range of organizations ensures that the final recommendations will be widely adopted. Furthermore, close ties have been established with national projects in the participating countries. During a definition phase of their project, supported under ESPRIT I Project 1541, a first multilingual speech database was established on CD-ROM for assessing and analyzing.

The activities in the main phase of the project will concentrate on

● Assessing speech input, including the setting up of tests, their scoring, and delineating performance factors

● Assessing speech output, including evaluation at the segment and suprasegmental level, and defining overall quality measures

● Enabling technologies, including the specifying uniform formats for collecting and managing speech data, annotating speech waveforms methods, and designing a European workstation with tools for speech labeling, analyzing, and I/O assessing.

A Vision System Development Environment for Industrial Applications (VIDIMUS, 2592)

The aim is to support the quick and efficient development of better quality industrial vision systems. The project is to construct a vision system development environment (VSDE) that can be used for forming vision systems for new industrial applications. The VSDE will have four major components: (1) vision framework, (2) functional representation knowledge-based system, (3) intelligent application mapper, and (4) algorithm performance library, brought together in a single integrated system. A subsidiary aim is to design and specify hardware to demonstrate acceptable processing times for the VSDE.

The vision framework will permit the construction of complex image interpretation systems for object recognizing, object registering, and accurate measuring. The functional representation KBS will permit the conversion of a high-level specification of an inspection process into a series of operations that can be executed by an image interpretation system. The intelligent application mapper will produce a specific vision system based on application information supplied by the user. The algorithm performance library will provide the intelligent application mapper with sufficient information for algorithm selection.

Computer hardware to demonstrate that the proposed VSDE can be implemented at the speeds demanded by the applications emerging from the above four software tasks will be mapped onto existing processors. Where algorithms are found not to be implementable on existing processors, new configurations will be designed and incorporated into the integrated computing environment.

Intelligent Training System in Industrial Environments (ITSIE, 2615)

The aim is to improve the level of training for operating and maintaining complex physical systems and potentially hazardous processes by providing computer-based training (CBT). The ITSIE is to enhance the current generation of CBTs and simulators for industrial applications by using KBS techniques in general, and qualitative modeling techniques in particular.

The main technical objective is to develop a generic architecture for ITSIE that allows the domain to be modeled in several ways and presented in the form most appropriate to the user and the task to be learned. Conventional simulation languages will be employed to represent the numerical models of the systems. Such a system architecture will permit a variety of instructional strategies to be explored within a given system. This provides an enhanced architecture for training systems that will greatly extend the capability of current CBTs.

In two phases, the project will use the techniques of qualitative modeling and, in particular, a component-based modeling language developed in ESPRIT Project 820 and QUIC. The first phase will establish the requirements for the demonstrator applications for ITSIEs, using both qualitative and quantitative models. This will lead to developing a set of tools to implement the identified modules.

The second phase will develop two working demonstrators to validate and refine the tools. The first demonstrator is a support system for current training practice in safely maintaining and repairing the devices of an electrical distribution network. The second demonstrator provides a full range of training in both routine and emergency procedures for fossil fuel powerplant operation. Also during this phase, the tools will be synthesized

and generalized to provide a generic architecture for ITSIEs. Finally, the demonstrators will be critically evaluated and further work recommended.

Front Ends for Open and Closed User System (FOCUS, 2620)

This project is to make software systems for industrial and scientific applications easier to use. The approach is to enable the human/computer interfaces to be enhanced by presenting software usage instructions and application domain knowledge when needed. This facility will be provided by attaching knowledge-based front ends (KBFEs) to the core packages that are left unchanged. Developing the KBFEs is a project objective; generic tools and techniques are required to construct and maintain them.

In phase one, the state-of-the-art prototype systems will be used to generate and define generalized tools, techniques, and methodologies. In phase two, these tools will be used to construct KBFE prototypes for closed user systems to be selected from free-standing software packages available within partner sites for a broad range of application domains. In phase three, prototype user systems will be developed and applied to an existing library of numerical and statistical programs to establish the requirements of open user systems for KBFEs. In all phases of the project, on-site user evaluation will be continuous, and interactive feedback will be given to the developers of the open- and closed-system prototypes and the tools.

Cost Management with Metrics of Specification (COSMOS, 2686)

This project is to support the cost management of software development for complex real-time systems. The objective is to develop a computer-aided software engineering tool for the creation of appropriate management support interfaces (MSI). The MSI will present development managers with graphics output giving information on specification properties that affect cost and resource expenditure. The MSI will also advise on the interpretation of this information in relation to the software system, the development environment and other management tools, and will monitor, review, and analyze management feedback.

Cost-management decision-making will be supported by applying the theory of software metrics using formal and semiformal development methods. Acquiring this information will be facilitated by using an enhanced version of an existing core metrics tool. Selection of the formal specification methods to be used in the observation studies will be based on identifying those methods that have genuine use or potential within the host organizations.

On the other hand, the COSMOS project will study common cost-management principles from different software development managers, in order to define a practical set of core metrics tool parameters and a user interface. The interface will use the formal information from the core metric tool together with nonformal information input by the software manager. The relationship with established cost models like COCOMO, SLIM, and Price/S will be studied, and documents from ESPRIT I projects like SPMSS, PIMS, and PCTE will be used when appropriate. Evaluating and enhancing the COSMOS tool will be achieved by conducting observation studies involving its use on real-time systems development projects; e.g., for telecommunications and industrial applications, selected and monitored by the industrial partners.

Appendices

The projects for Office and Business Systems, Computer-Integrated Manufacturing, Information Exchange System, and Basic Research are provided in Appendices A, B, C, and D, respectively, by number, title, and acronym.

Reference

ESPRIT Project Synopses, Directorate General XIII, Commission of the European Communities, September 1989.

APPENDIX A

Office and Business Systems

APPENDIX A

Office and Business Systems

2001	Storage, Processing, and Retrieval of Information in Technical Environment (SPRITE)
2054	Ultra Wideband Coherent Optical LAN (UCOL)
2058	Intelligent Communication Interface (ICI)
2071	Construction and Management of Distributed Open Systems (COMANDOS)
2082	Harmonized European Concepts and Tools for Organizational Information Systems (HECTOR)
2083	Structured Information Management, Processing, and Retrieval (SMPR)
2100	Metropolitan Area Communication System (MAX)
2102	Standard for Coding of Moving Images on Digital Storage Media (COMIS)
2103	Multi-Environment Advanced System for Color Treatment (MASCOT)
2105	Multiworks (MULTIWORKS)
2109	Telematic Object Oriented Tools for Services Interfaces (TOOTSIE)
2111	Planning of Nonspecific Transportation by an Intelligent Fleet Expert (PONTIFEX)
2114	Large-Size Visual Interface Design for Multimedia Workstation Terminals (LSVI)
2121	Integrated Toolkit for Highly Advanced Computer Applications (ITHICA)
2144	Information Technology Uptake Support Environment (IT-USE)
2148	Dynamically Adaptable Multiservice System (DAMS)
2170	Set of Software Tools for a Document Workstation (SUPERDOC)
2221	Reseau d'Information et de Communication Hospitalier European (RICHE)
2239	Self-Service Facilities Architecture (SESEFA)
2267	Integrated Systems Architecture (ISA)
2294	Tools for Object Based Integrated Administrative Systems (TOBIASI)
2315	Transistor's Workbench (TWB)
2322	IT Support for Emergency Management (ISEM)
2374	Piloting of the Office Document Architecture (PODA-2)
2382	Elusive Office (ELO)
2404	Primary Rare ISDN OSI Office Facilities (PROOF)
2431	Home Systems (HOME)
2455	Large Image Terminals (LIT)
2463	Application-Related Graphics in OSI Standards (ARGOS)
2466	Knowledge Workers Intelligently Collecting/Coordinating/ Consulting Knowledge (KWICK)
2476	Bank 92 (BANK 92)
2484	Spirit-I Workstation (SPIRIT)
2499	CD-ROM Workbench (CDR)
2512	Intelligent Area Communication and Information System (IACIS)
2563	General Distributed Architecture for Unified Communications in Heterogeneous OSI Environments (GAUCHO)
2569	European Workstation (EWS)
2638	Advanced Display Optimization Tools (ADOT)
2649	Visual Arts System for Archiving and Retrieval of Images (VASARI)
2684	Multipoint Interactive Audiovisual System (MIAS)

APPENDIX B

Computer-Integrated Manufacturing

APPENDIX B

Computer-Integrated Manufacturing

2010   Neural Product Definition Database for Large Multifunctional Systems (NEUTREBAS)

2017   Automated Process and Assembly Inspection by 3D Vision

2032   CIM Implementation Addressing Levels of Integration in Various Environments (CIM ALIVE)

2043   Mobile Autonomous Robot in an Industrial Environment (MARIE)

2090   Early Process Design Integrated with Controls (EPIC)

2091   Vision Based On-Line Inspection of Manufactured Parts (VIMP)

2127   Holographic Labeling Techniques for Automatic Identification in CIM Environments (HIDCIM)

2165   Integrated Modeling of Products and Processes Using Advanced Computer Technologies (IMPPACT)

2172   Distributed Intelligent Actuators and Sensors (DIAS)

2178   Revision Advisor--An Integrated Quality Support Environment (RA-IOSE)

2189   Building Industry Project Management System (BIPMS)

2192   Advanced Intelligent Multisensor System for Control of Boilers and Furnaces (AIMBURN)

2195   CAD Geometry Data Exchange (CADEX)

2198   Factory Customer Premises Network (FCPN)

2202   CIM System Planning Toolbox (CIM-PLATO)

2277   CIM for Multi-Supplier Operations (CMSO)

2280   Large Manipulators for CIM (LAMA)

2292   Testing Technology for Communication Network for Manufacturing Applications (TT-CNMA)

2331   Advanced Distributed Environment for Production Technology (ADEPT)

2338   Integrated Manufacturing Planning and Control System (IMP ACS)

2349   Fault Tolerant Control and Management of Production Systems

2415   Distributed Manufacturing Planning and Control

2428   Intelligent Process Control by Means of Expert Systems (IPCES)

2434   Knowledge-Based Real-Time CIM Controllers for Distributed Factory Supervision

2439   Real Monitoring and Control of Construction Site Manufacturing (ROCOCO)

2457   Knowledge-Based Planning and Control in Manufacturing Environments (FLEXPLAN)

2483   Perception and Navigation System for Autonomous Mobile Applications (PANORAMA)

2486   Integrated CAE Techniques for Dynamic Analysis of Structures (ICTDAS)

2527   CIM System and Distributed Database and Configurable Modules (CIDAM)

2588   Design Support for Distributed Industrial Control (DSDIC)

2590   Integrated Product Design System (IPDES)

2617   Communications Network for Manufacturing Applications (CNMA)

2623   Methods for Advanced Group Technology Integrated with CAD/CAM (MAGIC)

2626   Intelligent System for Automatic Processing of Design Codes of Practice (AUTOCODE)

2637   Advanced Robotics Manipulation System (ARMS)

2640   Integrated Intelligent Process Control and Inspection in Robot Finishing (ICI)

2656   Intelligent Drive for Shop Floor Systems (IDRIS)

2658   Advanced Robotics in Flexible Automation Components, Tools, and Strategies (ARTIFACTS)

2671   Knowledge-Based Multisensor Systems in CIM Applications (KB-MUSICA)

APPENDIX C

Information Exchange System

APPENDIX C

Information Exchange System

APPENDIX D

Basic Research

APPENDIX D

Basic Research

3001   Vision System for the Natural Human Environment (INSIGHT)

3002   Categorical Logic in Computer Science (CLICS)

3006   Theories of Concurrency: Unification and Extension (CONCUR)

3011   Compositional Distributed Systems (CEDISYS)

3012   Computational Logic

3014   High Temperature Superconductivity: Concepts, Models, and Methods (HTSC THEORY)

3017   Electrical Fluctuations and Noise in Advanced Microelectronics: Submicron, 2-D Gas and Low Temperature Devices (NOISE)

3020   Integration

3023   Information Systems Correctness and Reusability (IS-CORE)

3026   Heterostructures of Semiconducting Silicides on Silicon: Applications to Si-Compatible Optoelectronic Devices (HESSILSIL)

3030   Acquisition of Lexical Knowledge for Natural Language Processing Systems (ACQUILEX)

3038   Vision as Process (VAP)

3041   Possible Mechanisms for High Tc Superconductivity and Phenomenologic Approaches (MESH)

3042   Performances and Physical Limits of Heterostructure Field-Effect Transistors ((NANOFET)

3043   Lateral Microstructures: Fabrication, Low Dimensionality Effects and Applications to III-IV Devices (LATMIC)

3049   Innovative Architectures and VLSI Implementations for Neural Computing (NERVES)

3050   Development of Representation for Machine Learning from Imperfect Information (ECOLES)

3066   Assimilating Models of Designers, Users, and Systems (AMODEUS)

3070   Formally Integrated Data Environment (FIDE)

3074   The Semantics and Pragmatics of Generalized Graph Rewriting (SEMAGRAPH)

3075   Algorithms and Complexity (ALCOM)

3085   Defeasible Reasoning and Uncertainty Management Systems (DRUMS)

3086   Low Dimensionality Structures for Future Quantum Semiconductor Devices

3092   Predictably Dependable Computing Systems (PDCS)

3096   Formal Methods and Tools for the Development of Distributed Real-Time Systems

3104   Provable Correct Systems (PROCOS)

3105   Models of Human Actions in Work Contexts (MOHAWC)

3109   Programming for MOS Processing Technology (PROMPT)

3121   Conducting Organic Materials as Molecular Components for Microelectronics (MOLCOM)

3124   Semantics-Based Program Manipulation Techniques (SEMANTIQUE)

3125   Mechanizing Deduction in the Logics of Practical Reasoning (MEDLAR)

3133   Nanostructures for Semiconductor Devices (NANSDEV)

3146   Study of the Influence of Impurities on the Properties of High Tc Superconductors (DIRTYSUPRA)

3147   Hierarchical Integration of Logic and Functional Paradigms: Specification, Refinement, and Implementation (PHOENIX)

3148   Design Methods Based on Nets (DEMON)

3149   Multisensory Control of Movement (MUCOM)