



TOGETHER
for a sustainable future

OCCASION

This publication has been made available to the public on the occasion of the 50th anniversary of the United Nations Industrial Development Organisation.



TOGETHER
for a sustainable future

DISCLAIMER

This document has been produced without formal United Nations editing. The designations employed and the presentation of the material in this document do not imply the expression of any opinion whatsoever on the part of the Secretariat of the United Nations Industrial Development Organization (UNIDO) concerning the legal status of any country, territory, city or area or of its authorities, or concerning the delimitation of its frontiers or boundaries, or its economic system or degree of development. Designations such as “developed”, “industrialized” and “developing” are intended for statistical convenience and do not necessarily express a judgment about the stage reached by a particular country or area in the development process. Mention of firm names or commercial products does not constitute an endorsement by UNIDO.

FAIR USE POLICY

Any part of this publication may be quoted and referenced for educational and research purposes without additional permission from UNIDO. However, those who make use of quoting and referencing this publication are requested to follow the Fair Use Policy of giving due credit to UNIDO.

CONTACT

Please contact publications@unido.org for further information concerning UNIDO publications.

For more information about UNIDO, please visit us at www.unido.org

19822

24p

Final Report

Project: DP/RER/87/036
Contract: 91/180/VK

Summary

According to the contract - McDonnell Douglas performed following activities in your project:

1. Delivery of seven (7) software packages to the Lead Focal Point.
2. Five days training held in Prague (11.-15. November 1991) COURSE SEPW and SEP4
3. One day additional presentation in Budapest

Table of Contents

	Page
Contractual Information	3
Technical Information	5
Product Descriptions	
ProKit*WORKBENCH	8
PRO-IV	19
Training Descriptions	
ProKit*WORKBENCH	23
PRO-IV	24

Final Report

Project: DP/RER/87/036
Contract: 91/180/VK

The Final Report for UNIDO is structured in 2 sections, the contractual- and the technical-section.

'Contractual Information'

Ref. 3. d) i) McDonnell Douglas ability and readiness to do the job

McDonnell Douglas has many years of experience in handling international accounts. As one of the leading system integrators McDonnell Douglas has been involved in international projects. McDonnell Douglas started its operation in 1978 with a cooperation with IST (Information Systems Technologies), the former company of Chris Gane and Trish Sarson, the leading initiators for structured systems Analysis and Design. In 1982. McDonnell Douglas acquired the IST company. During the following years our company has developed several CASE-Tools, the first in 1982 called STRADIS DRAW. In 1987 we introduced the PC-based CASE-Tool ProKit*WORKBENCH. Since then several enhancements have been made. The actual Version 2.1 was announced in March and delivery started in May 1991. ProKit*WORKBENCH 2.1 is sold in two versions, the 'single-user' version and the 'multi-user' version. The 4.GL PRO-IV has been bought in 1987 by McDonnell Douglas. Since then it was integrated into the McDonnell Douglas Enterprise Engineering Environment E3. The E3 concept contains the Information Systems Engineering I.S.E. products from McDonnell Douglas, the methodology STRADIS, the CASE-Tool' ProKit*WORKBENCH, the 4.GL PRO-IV, the reengineering Tool Legacy Systems Rebuild and supporting training and consulting for all the above mentioned products. To UNIDO the products ProKit*WORKBENCH and PRO-IV were offered.

Ref. 3. d) ii) References from McDonnell Douglas about similar contracts.

Names and works done

McDonnell Douglas has worked with several UNO organizations around the world including the UNO and WHO offices in Geneva, where a library system has been installed.

The I.S.E. department of McDonnell Douglas has experience with several multinational accounts and state organizations, as for example the State of Mississippi USA and Czechoslovakia, where a very big collaboration agreement was closed recently. Multinational accounts as AMERITECH, ERICSON, 3M, NATO and TNT are satisfied customers round the world. In most of the mentioned accounts McDonnell Douglas has provided support for the total software development life cycle including products, training and consulting.

Ref. 3.d)iii) Understanding of services.

McDonnell Douglas understands its role with regard to the services to perform as follows: Help to set up the training environment in Prague Train the representatives from the 7 countries during 5 days. Give support to the representatives in case of problems by telephone or on request on site. Give free update to the customers within 1 year after delivery.

Ref. 3. d) v) Statement of our intention to be efficient and to assign qualified personnel.

The training and consulting personnel of McDonnell Douglas is highly trained and qualified for the work they have to do. All involved persons have good knowledge of English.

'Technical Information'

Ref. Documents title: UNDP/UNIDO Regional Project
RER/87/036
"Industrial Computerized Management System"

The project is basically addressed to CASE-Tools and 4. GL. These tools help to increase the software development capabilities, skills and working experience.

The products from McDonnell Douglas are rated as professional, highly efficient and user-friendly. ProKit*WORKBENCH and PRO-IV will be described thoroughly in the next section.

Ref. B. The Output of Contracting Services Development Environment for CASE and 4.GL

The software development environment for using McDonnell Douglas tools can be chosen between IBM PC/AT or 100% compatible systems for single-user development workstations or between different PC-network platforms as for example on Novell/Ethernet, PC-LAN/Token-Ring. All IBM PC processors are supported including 80286, 80386 and 80486 and variants like SX types. PRO-IV can be used on other hardware, however. Programs developed on a PC can be easily ported to different operating systems, databases and hardware platforms.

As an integrated part of the "Enterprise Engineering Environment" from McDonnell Douglas the tools are well adjusted to each other for supporting - the total life cycle from analysis to design as well as development and maintenance.

Ref. B. Technical Specification

Implementation of Conceptual and / or Data Modeling

The data-modeler module supports you on both the conceptual and the data modelling. Special features are included to help in data normalization. 3 different notations are supported, extended Bachman, extended Chen and the relational model IDEFIX.

Application Generators

The CASE Tool ProKit*WORKBENCH module "Prototyper" allows easy prototyping and application navigation. Defined masks and Forms can be used in PRO-IV for screen and report generation. The prototyper can automatically generate 7 different 3. GL screen definitions.

Within PRO-IV fast development prototyping can be done. Since PRO-IV is table driven no time intensive compilations are necessary. Even in big applications the tables are linked in a few seconds.

Powerful Query Language and SQL Report Writer

The report module of ProKit*WORKBENCH is divided in two parts, the macro report and the AD-HOC query reporting part. In the macro reports over 150 standard predefined macros are prepared. Each macro can be customized and stored under a new name. The macro reporting facility covers the day to day reporting business. In the AD-HOC query reporting all other reports are covered. Every object and related attributes can be selected and be used for the SQL based report. The report paint facility allows to design reports including headers etc.

Central Data Dictionary

The object orientated repository of ProKit*WORKBENCH is a highly sophisticated data dictionary. As nucleus of the tool it combines the different modules and there objects into one repository where information is stored one fact one time. Several synchronization routines are built in to guarantee the consistency of the CASE project.

Import / Export Modules and Interface Concept

Via the import / export module it is possible to work with data from and to other sources. Data can for example be loaded from a host based dictionary, then modified in the CASE environment and reloaded into the host.

Existing interfaces with other products as Ventura Publisher

are features of the tools. All information from the repository can be put on a ASCII - file for application of the data in every environment which can use ASCII source.

The high level interfaces from ProKit*WORKBENCH are built to load the CASE project into different generators and 4.GL products The following generators and 4.GL are addressed:

- Telon Cobol-Generator
- Transform Cobol-Generator
- CSP Cross System Product from IBM
- PRO-IV the 4.GL from McDonnell Douglas

Reverse Engineering

Reverse Engineering is sold as a service within the McDonnell Douglas organization. The customer provides McDonnell Douglas with a floppy disk of his source programs. McDonnell Douglas will then reverse engineer the source and provide a project disk for ProKit*WORKBENCH. Today this service is offered for COBOL using different dialects.

Product-Descriptions

ProKit*WORKBENCH

Prokit*WORKBENCH Overview

Prokit*WORKBENCH is a set of advanced, integrated development tools for the analysis, design, and development phases of the software life cycle and based on McDonnell Douglas Software Development Methodologie STRADIS. The application of this Computer-Aided Software Engineering (CASE) provides an effective and efficient means for translating strategic information objectives into software that satisfies stated requirements. As a CASE tool, ProKit*WORKBENCH provides numerous important benefits:

Project Effectiveness

The effective development project successfully translates strategic business plans and objectives into a viable tactical software system that has a favorable cost benefit profile. ProKit*WORKBENCH tools provide the mechanism for a structured, controlled and effective development process. It aids the developer in performing and integrating the fundamental techniques of structured analysis, datamodeling, prototyping, and structured design. The repository integrates the ANALYZER, PROTOTYPYER, DATA MODELER, and DESIGNER functions allowing you to build an information base that establishes continuity in the project from the first analysis to the last design activity. The application of ProKit*WORKBENCH in the development process can make a major contribution to the goal of creating a system that satisfies business and system objectives and is cost effective.

Productivity

ProKit*WORKBENCH's design maximizes the output of the development team. By providing order and integration of operations, it offers the potential of controlled directed activity to eliminate redundancy. The use of regular review and approval based on the ProKit reporting system increases efficiency. Another reason for productivity is the ability to refine the graphics. The diagram editors are linked directly to the repository to ensure completeness and accuracy of data. From repository information, full reporting provides a tool for checking completeness and accuracy. Automated and highly integrated in its functions, ProKit*WORKBENCH is a proven productivity tool.

Project Documentation

The repository contains extensive detail of every symbol and image field that is captured in the graphics of ANALYZER, DATA MODELER, DESIGNER, and the images of PROTOTYPED. System development is fully documented from beginning to end within the repository. In addition, the repository extender function allows you to add attributes to system objects that can handle any special project information requirements. The information contained in the repository is used for completeness and accuracy checking, as well as for balancing data flow diagrams. In addition, this documentation allows the reporting of major project deliverables including final system documentation.

Communications

As a dynamic interactive tool, ProKit*WORKBENCH provides the basis for communications throughout a project. Data flow diagrams, data models, prototype images, and structure charts created by ProKit*WORKBENCH functions can be produced on numerous printers and plotters for presentations and reviews. In addition, presentation diagrams allow you to use 30 symbols to prepare diagrams for presentations and management reviews. Predefined standard reports and ad hoc reporting document all aspects of the project. These deliverables can be used for informal project activities, professional presentations, or major project deliverables. They are vital to the review and approval activities. They form the basis for consensus on the project direction and achievement.

Project Continuity

To promote a structured, cohesive progression through the development life cycle of a project, all major functions of ProKit*WORKBENCH are united through the repository. All activities reference the same repository object data which is the basis for common understanding and control for all participants. For example, a data element definition established and approved in early analysis is valid throughout the life cycle. Progressive models of the developing system can be migrated through the life cycle with confidence that the fundamental data view remains the same throughout.

Data Accuracy, Completeness, and Consistency

ProKit*WORKBENCH contains built-in checks for data accuracy and completeness. The interactive link between graphic editors and the repository synchronize information for accuracy. Reports indicate incompleteness of necessary information. Additional reports show balancing errors on diagrams. Throughout, data input formats ensure that consistent use of

data specifications and conventions are maintained.

System Quality

The use of ProKit*WORKBENCH can contribute substantially to the effectiveness of a development project. It provides the tools by which the project satisfies its original intent - to produce the right system in a cost effective manner. The diagram editors and repository are instruments for accomplishing the necessary tasks of system development. Comprehensive reports provide for checking, balancing and completing the repository information, diagrams and images. Applied in a structured way, ProKit*WORKBENCH is an excellent set of tools for generating a high quality system that is well documented and easy to maintain.

ProKit*WORKBENCH Functions

The Main menu of ProKit*WORKBENCH can be divided into three main categories:

Major functions include ANALYZER, PROTOTYPED, DESIGNER, and DATA MODELER. ANALYZER, PROTOTYPED, and DATA MODELER functions relate to the analysis phase of the project. The DESIGNER function relates to the design and maintenance phases of the project.

Support functions include the REPOSITORY, DIAGRAMS, REPORTS, PROJECT, and VERSIONS. These functions directly support the major functions listed above.

Maintenance functions include IMPORT, EXPORT, BACKUP, HELP, and INTERFACE.

ANALYZER

ANALYZER is a major function in ProKit*WORKBENCH that assists you in developing a logical model of a system. You gather and document system requirements using data flow diagrams (DFD's) and the supporting repository. Graphic symbols are available for depicting data flows, data stores, processes, and external entities. You can identify certain data elements and data structures on data flows as keys which can be used to access information from a data store. You can also add - free-form text to the diagram for more detailed annotation; and define boundaries for grouping symbols. The system supplies a legend for the diagram.

Using structured analysis techniques, the DFD progresses from high-level system view into lower level diagrams that represent explosion of processes detail. ProKit*WORKBENCH supports system level diagrams with up to 11 levels of explosion to show progressive detail.

As you draw the diagram, the system places pertinent information automatically in the repository. This repository is integrated with all of the ProKit*WORKBENCH graphics tools (except for presentation diagrams). If you add a diagram symbol that has been defined previously, the repository supplies the defined labels. You can use function keys to switch between the diagram editor and the repository. By using these keys, you can obtain a directory listing for objects or document additional repository information about the symbols. When you make a change to a repository object, those changes are reflected in the corresponding symbol on all diagram surfaces. The system updates the repository objects when changes are made to the corresponding data flow diagram symbol.

ProKit*WORKBENCH validates each data flow diagram against proven diagramming techniques as it is being drawn. Below are a few of the integrity checks made as symbols are drawn and labeled:

- o Improper source/destination for data flows
- o Mixing of public and private relationships
- o Invalid process identification

In addition, you may use reports that are designed to check that data flowing through a diagram is in balance both horizontally and vertically between diagram levels. This check helps to ensure validity and assists in catching errors quickly without manual checking.

DATA MODELER is a major function in ProKit*WORKBENCH. It assists you with gathering and commenting data requirements and their interrelations. E/R (Entity/Relationship) diagrams are the primary deliverable of data modeling and are graphic representations of the data and its interrelationship. Data modeling defines for an organization:

- o What data is needed.
- o Interrelationship of that data

This data is needed by the organization to support decision making and provides rules to govern the use of the data. At the minimum, an E/R diagram must:

- o Represent all needed data
- o Correctly represent the interrelationships of that data based on business rules
- o Be able to support all the end user views of the information

ProKit*WORKBENCH's, E/R Diagram uses data entities and relationships to meet these requirements. Specific information about these objects is maintained at the diagram level and in the repository. The diagram combines with the repository to represent the data model. The data model should accurately define the business rules, data requirements, and user access needs.

Modeling data is essential at both application and corporate levels of an organization. Many different user or information Views can be created using the same data. Therefore, data can be shared within application functions and among all application Systems that support the organization. This approach emphasize the management of data as a resource.

ProKit*WORKBENCH supports three data modeling techniques:

- o Extended Chen Entity Relationship (E/R) diagrams
- o Extended Eachman Entity Relationship (E/R) diagrams
- o Relational Entity Relationship (E/R) diagrams (such as IDEF1X)

Additional features of the ProKit Data Modeler provide:

- o Data store/data entity synchronization which integrates data flow diagrams and data models
- o Normalization which aids in reducing data redundancy and provides the basis for an efficient and flexible database design
- o Generation of the DB2 schema definition statements from information collected in DATA MODELER

PROTOTYPER is a major function in ProKit*WORKBENCH that allows you to simulate the system being developed or document an existing system. When using this function, you can create menus, screens, reports, and forms to construct a prototype of the system at any stage of development. You can identify and make changes early in project planning prior to a significant investment of effort and money. It is also an excellent tool to refine, validate, and document these requirements.

PROTOTYPER is fully integrated with the Repository and has many different features and capabilities to support you. These include:

- o Image Painting allows you to paint menus, screens, reports, and forms. You can tie an image to a data flow which automatically provides the contents of

the data flow to you for painting. This unique integration helps to validate the contents of a data on a data flow diagram. Variables placed on images can be added as data elements to the repository if you wish.

- o State Specification allows you to define states which provide navigation through transfer criteria. It allows you to tie static images together which gives reality to the prototype system.
- o Keyboard and display configurations allow you to experience the prototype system with different man/machine interaction and to determine the best user interface. You can simulate keyboard and display configurations when the system being prototyped runs on different machines and operating environments.
- o Execution of the prototype allows the state specifications to control the images displayed and the navigation between images. The starting state is defined, which gives the user the capability of breaking in at various points within the prototype system. In addition, the combination of the state specifications and appropriate keyboard/display configurations allows you to experience various "what-if" scenarios.
- o Utilities provide deliverables such as reports, copylibs, maps, and on-line help. Several output media are available to create program specification reports, copylibs in seven languages, and CICS BMS and IMS/VS MFS maps. It also allows you to resequence an images cursor movement and browse all reports on-line.

Using PROTOTYER, you can document and simulate a system's data, navigation, and behavioral requirements.

DESIGNER is a major function of ProKit*WORKBENCH that allows you to transform the results of analysis and strategic design into a tactical blueprint for implementation. Structure charts are graphic representations of physical design. The structure chart diagram editor allows the designer great flexibility in producing and maintaining structure charts.

The following module type designations are provided to align the needs of program logic and language requirements:

- o Computing Resource (CR)
- o Function (FN)
- o Load Module (LM)
- o Paragraph (PA)
- o Program (PG)
- o Sub-program (SP)

o Undefined Module

The linkage between these modules may be fully characterized with linkage variables symbolized as control or data; and public or private. Recursion loops and decision points may also be represented. In addition, the ability to break high-level modules into separate structure charts increases the flexibility of design modeling.

A special structure chart symbol shows modules lexically included in others. It also provides a boundary function for grouping modules as a subset allowing the software designer to emphasize functional or organizational unity.

As with the other graphic editors, information on the diagram surface is automatically captured and stored in the ProKit*WORKBENCH repository. Information documented during the analysis phase is available for review, reuse, and refinement. For example, a process and its narrative can be associated with a module that implements the solution. This process narrative can be included with the logic specification area of the module. The designer can then expand and refine the narrative to reflect the complete logic definition of the module.

The following four design objects support the design effort:

- o Subsystems
- o Design Units
- o Modules
- o Database/Files

Support Functions

REPOSITORY

The Repository is the collection of basic facts and relationships about objects captured, described, and used for decision making during software engineering activities. It provides the supporting detail for the graphics functions and is the basis for all reports provided by ProKit*WORKBENCH. The repository links all functions of ProKit*WORKBENCH into one cohesive body of information. In addition, the repository extender function allows you to add user-defined attributes to existing repository objects. A maximum of 25 attributes can be defined plus one additional text field.

The development of a project is documented as it progresses, providing the continuity that results in an implementable system specification that properly reflects business objectives and user specifications.

REPORTS

The REPORTS function is highly versatile and provides flexible control over the reporting process. Specific information can be selected from the full domain of repository information by using system available or user-defined macros. In addition, Ad hoc query/reporting provides ease in quickly designing customer formatted reports. By using selection qualifiers, REPORTS provides the capability to broaden or narrow the scope of information for your reporting purposes.

VERSIONS

ProKit*WORKBENCH supports the creation of eight independent versions for each system development project. This multiple version support allows you to:

- o Model a number of systems within a project.
- o Move from a general project view to a more specific or detailed view.
- o Freeze a project at a major milestone or after management review and then continue with refining the project.

DIAGRAMS

The DIAGRAM option allows you to access all the diagram editors in ProKit*WORKBENCH. Presentation diagrams allows you to prepare graphic and detailed diagrams for presentation to management and other review groups. You can prepare diagrams that:

- o Use 30 symbols including frame, text, boundary, and lines to prepare your diagrams.
- o Use familiar symbols that will give your audience a clear view of the system.
- o Prepare any type of diagram you wish, such as state transition, architecture schematic, and screen navigation.

Data flow diagrams are graphic representations of the flow of information through a system. E/R (Entity/Relationship) Models are graphic representations of data and their interrelationships for a system. Structure charts are graphic representations of the physical design of a system.

PROJECT

PROJECT is a support function designed to give the system or project manager effective control over the ProKit*WORKBENCH environment. It provides the capability to maintain basic information about the project as well as to control the creation, deletion, archiving, and restoring of project versions. This function also allows the establishment of project security through user ID's and passwords.

Maintenance Functions

The following functions provide both necessary and useful activities for maintenance of the project.

IMPORT

This function allows you to retrieve project data from exported ProKit files or external sources. This project data can be imported from another system, project, workstation, or corporate data dictionary into selected project versions.

Import rules govern whether objects, diagrams, or images can be imported. You may choose to accept or reject any objects that will replace objects already existing on the importing workstation.

EXPORT

This function provides a flexible means of transferring repository objects and their relationships (including diagrams and images) from a workstation running ProKit*WORKBENCH to an ASCII file. This file may be used to:

- o Allocate project work among two or more workstations in a multi-workstation environment.
- o Share information between projects.
- o Reuse information from a completed project information needed for all projects °
- o Update your corporate data dictionary after project completion.

BACKUP

This function allows you to copy repository information, diagrams, and images to an external storage medium for a selected project version. This function:

- o Prepares you to restore damaged or accidentally deleted files associated with a project version.
- o Prepares you to restore a project version to a previous condition after "what-if" scenario experimentation.
- o Assists you in distributing project version information in a multi-workstation environment after synchronization has taken place at the master workstation.

HELP

The HELP function gives access to all the HELP screens in ProKit*WORKBENCH system through a menu structure that corresponds to the normal operational menus. The help screens provide information on how to operate ProKit*WORKBENCH and can be accessed at function, subject, or detail level.

INTERFACE

This function allows you to execute external programs without leaving the ProKit*WORKBENCH environment. You can use the user-defined interface to predefine frequently used software in ProKit*WORKBENCH and access it in much the same way as other utility functions. In addition, it allows you use of other formal interfaces such as PRO-IV, TELON, and TRANSFORM.

Life Cycle Application

ProKit*WORKBENCH functions allow you to apply the basic techniques of system development. You will be able to use them in different ways to meet a variety of needs. System development begins with business analysis which identifies business objectives, and the subsequent definition of objectives for the system that will be developed.

Structured analysis focuses on the analysis of the logical system, data entity relationships, and the processes which result from the system analysis. You can apply ANALYZER and PROTOTYPER to the development of system models. You can apply DATA MODELER to data analysis and the subsequent generation of data models. Processes are identified on the system model and their logic is entered into the repository. The basic components of structured design are process logic, design units, and design models.

Structured design transforms the inputs from analysis into a physical blueprint for implementation as a software system. You can add more details to process logic in the repository for identification of modules for input into program design. You develop database design from data models by using DATA

MODELER.

You can use PROTOTYPER to further refine the human interface with screens, reports, menus, and forms. DESIGNER - allows you to facilitate the creation of structure charts. Output from the design phase to development includes detail process logic, structure charts and final database design.

ANALYZER supports the following phases in the system life cycle:

- o Strategic Planning
- o Analysis
- o Maintenance

DATA MODELER supports the following phases in the system life cycle:

- o Strategic Planning
- o Analysis
- o Design
- o Development
- o Installation
- o Maintenance

PROTOTYPER supports the following phases in the system life cycle:

- o Analysis
- o Design
- o Development
- o Maintenance

DESIGNER supports the following phases in the system life cycle:

- o Design
- o Development
- o Installation
- o Maintenance

ProKit*WORKBENCH provides the tools necessary for the key activities in software engineering. It assists you in many vital ways to achieve the goals set forth in your business analysis, no matter what individual application you are developing.

PRO-IV

PRO-IV Overview

Unlock your potential to increase productivity, greater efficiency and improve user satisfaction with PRO-IV, the professional application development language from McDonnell Douglas.

With PRO-IV, you can complete the circle of full life cycle system development because PRO-IV handles the development, installation and maintenance phases. Our front-end CASE tool ProKit*WORKBENCH provides analysis and design solutions, and PRO-IV delivers back-end CASE solutions - while you deliver targeted business applications on time within budget.

Applications developed with PRO-IV are hardware, operating system and database independent. An application can be transported vertically, from microcomputers to minicomputers to mainframes. IBM, DEC, and UNIX-based applications are compatible when they are developed with PRO-IV. And your existing files and databases remain accessible when you use PRO-IV.

What is PRO-IV?

PRO-IV is a complete structured environment in which you develop, execute and maintain business applications. This structured environment guides the developer through a series of fill-in-the-blank screens, which define application function specifications, greatly simplifying application development and maintenance.

PRO-IV Portability

With PRO-IV, developers increase their flexibility because they can concentrate on creating applications that meet user needs, rather than wrestling with the restrictions of a particular processing environment. PRO-IV consists of a library of more than 150 memory-resident, optimized re-entrant modules written in native machine code. These modules form the PRO-IV kernel, which contains all the PRO-IV operating functions required to develop and process business applications. Each module performs a specific function such as "read a record", "open a file", "display data on CRT" and more.

To create an application, PRO-IV guides the developer through a series of fill-in-the-blank screens. Next, PRO-IV analyzes the entered information to create a linkage table, thus producing the desired application. Then PRO-IV applications can be transported vertically to a wide range of micros, minis and mainframes. You can develop and test the application on a micro to use on a mainframe - or develop on a mainframe to use on a micro. The application can be transported horizontally to

diverse systems such as IBM, DEC, and UNIX-based environments, and ported to any supported system without changes.

Productivity with PRO-IV

PRO-IV significantly increases productivity. Menu screens and fill-in-the-blank screens coupled with a comprehensive logic language, promote swift development of sophisticated application solutions. PRO-IV applications can be developed up to 20 times faster than conventional programming languages. Even greater productivity increases in maintenance can be realized - up to 300 times over conventional languages.

Development Environment

PRO-IV is completely menu-driven, organized in an inverted tree structure. These menus allow the software engineer to:

- o Define application data files
- o Define the functions, including input screens and logic functions
- o Produce system documentation
- o Define system security
- o Support a variety of utilities

Data and File Definition

Data and file definition is the first step in creating a PRO-IV application. You define the logical file record layouts and the relationship to physical databases or files. Once defined, logical files can be referenced from any application, any number of times. It also establishes the placement order and attributes of data fields on the screen.

Function and Screen Development

To form an application, PRO-IV creates four types of functions:

- o Menu function - the user selects a function from a list of choices. This eliminates numerous commands for the user.
- o Screen function - controls screen operation and guides the user to enter, view or modify system information.
- o Report function - prints reports based upon database

information and provides updates to the database.

- o Update function - transfers information from one file to another.

The screen, report and update functions support multiple file read and write capabilities. PRO-IV contains powerful logic syntax for manipulating and testing data to support these functions.

Documentation of PRO-IV

PRO-IV automatically prepares and maintains on-line documentation for applications, files and data definitions. Comprehensive information about each application step is collected during the development process. Specification printouts are concise, consistent and easy to read. Documentation is automatically updated and always reflects the current system.

Security

During all phases of a PRO-IV session, the security system is active. PRO-IV uses existing security systems and provides security at all levels: company/division, terminal, operator, function, file and field. Encryption of data fields is also provided.

Special Features

- o On-Line help messages for beginners; expert mode for experienced developers
- o Interface to front-end CASE-Tool ProKit*WORKBENCH
- o Efficient resource utilization. Occupies only 330 Kbytes of Memory
- o Supports multiple DBMS and file systems including DB2, IMS, and DL/I
- o Offers full functionality across all platforms
- o Contains user-oriented features such as screen highlighting, color and functions for screen development
- o Prototyping allows you to preview applications at any stage of development to ensure the right fit for your requirements. This eliminates typical reprogramming delays. The prototype is the actual working application.

Unlock Your Potential

PRO-IV unlocks the potential of your development resources. You can quickly and efficiently create software solutions and deliver applications on time and within budget - and those applications will meet and exceed user expectations.

PRO-IV minimizes development and maintenance costs. Hardware and database dependency is reduced. PRO-IV is your key to a better way of developing and maintaining software solutions.

McDonnell Douglas software engineering solutions support and enhance the complete application development life cycle with methodology, training and consulting.

Training

COJRSE SEPW

APPLIED ProKit*WORKRBENCH

This is a hands-on course which is designed to teach students how to use this advanced development tool during the analysis and design phase of the software development life cycle. The course shows how each module of ProKit*WORKRBENCH approaches systems development as a step by step building process, and how development techniques are incorporated into this automated tool.

SPECIFIC TOPICS

- * "Analyzer"
How to use this function to carry a model of the current or proposed system through progressive analysis stages until it is ready for detailed design.
- * "Data Modeller"
How this function documents and integrates an organisations data resources.
- * "Prototyper"
Which is used to define, document and simulate a proposed system before its development.
- * "Designer"
How to use this function to transform the analysis results into a blueprint for computer system implementation.
- * "Repository"
How the object-orientated repository is used to unify and store information gathered at all stages of the system life-cycle.

WHO SHOULD ATTEND

All developers working in a ProKit*WORKRBENCH environment would benefit from attendance on this course.

COURSE SEP4

PRO-IV INTRODUCTORY

The Course is designed to introduce programmers and analysts to PRO-IV. It is based around an example application which the student creates during the course.

The applicatton requires the use of all the fundamental areas in the PRO-IV development process, from file design to dynamic function linkage.

Students will learn by hands-on experience and detailed examples.

The course will equip the students with the relevant knowledge and information to enable them to begin application development in PRO-IV.

SPECIFIC TOPICS

- * Develop an understanding of the key features of PRO-IV.
- * File creation and maintenance.
- * Function design, creation and maintenance.
- * Learn and use the development process.
- * Discuss efficiency and file linkages.
- * Use of logical screens.
- * Gain a sound fundamental knowledge of the timing cycle.
- * Gain an understanding of logic commands and usage.
- * Produce a sample application using PRO-IV.
- * Prepare for actual product use with guidelines on how to develop further knowledge of PRO-IV.
- * Use of PRO-IV utilities.

WHO SHOULD ATTEND

This course is designed for programmers and analyst programmers who have no previous experience of PRO-IV.

Students receive a course manual, quick reference guide, worked examples of the exercises used in the course and a disk of their own completed course application.