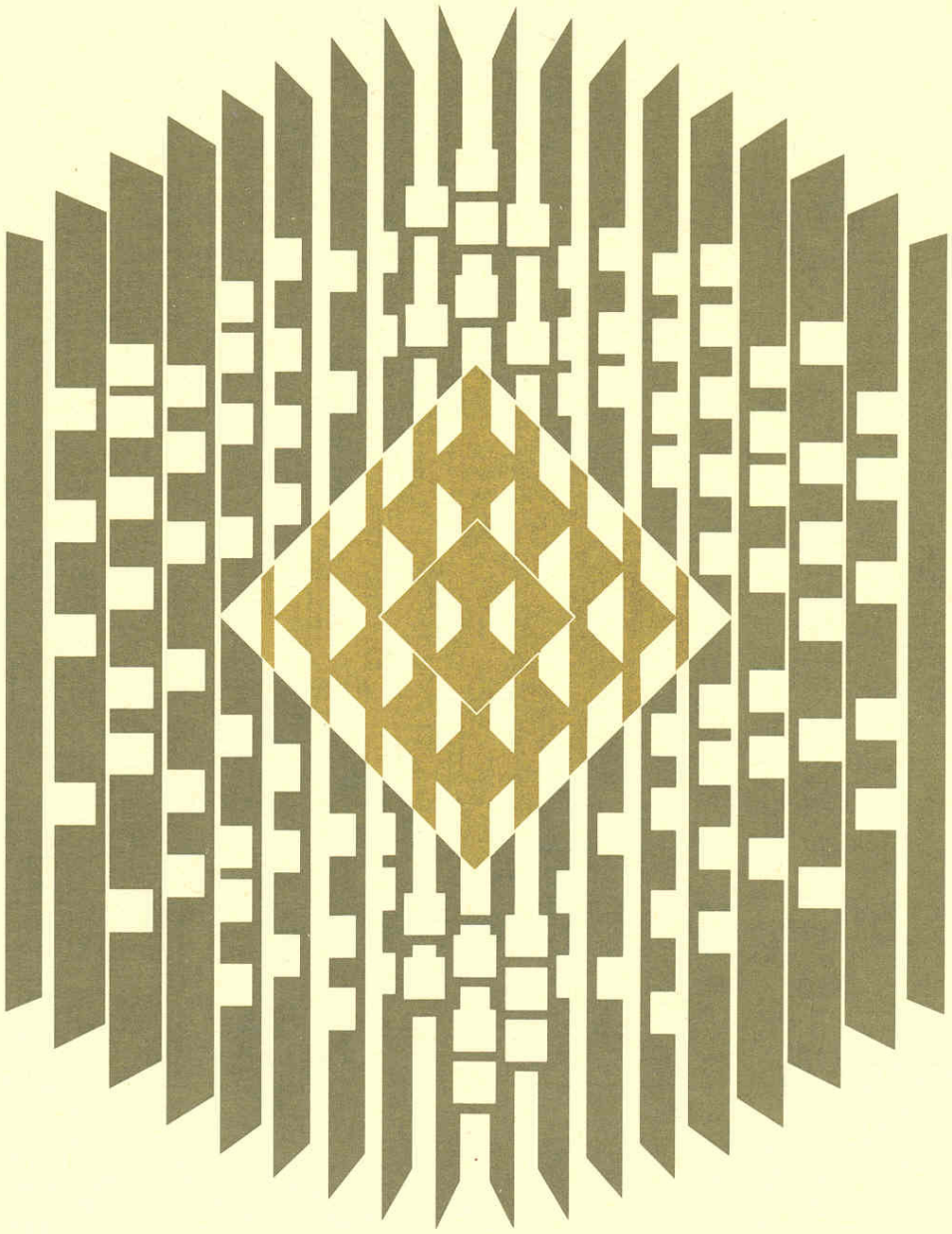
SPERRY UNIVAC

1100 OS

Facts and Figures



SPERRY UNIVAC

The SPERRY UNIVAC 1100 Operating System

Introduction

The 1100 Operating System is the culmination of literally thousands of man years of development and millions of hours of operation. The results of those years of effort provide facilities for achieving highly cost-effective solutions to a wide range of problems. The evolution of the operating system's control structures and algorithms results in both efficiency and stability.

Two main points of SPERRY UNIVAC software policy are of particular relevance:

Continuing Change and Enrichment

The 1100 Operating System, which is the single operating system for all SPERRY UNIVAC Series 1100 hardware, is continually undergoing a controlled evolution. New hardware is supported, new software capabilities are developed, existing software is reviewed, enhanced, and made more efficient. All of these changes are integrated into the operating system and given extensive live systems test before release for customer use.

Continuing Stability and Maturity

This process of change is carried out on the foundation of the stable and refined base which has been achieved through many years of extensive use. Each developmental modification of the system not only contributes new capabilities, but also contributes to an increased level of stability and flexibility of use. Careful analysis of the many years of customer experience provides the framework for ensuring and validating each new software component.

Main Components of the 1100 Operating System

The Executive System

At the heart of the 1100 Operating System is the Executive, which has been tailored to take full advantage of the capabilities of the new Series 1100 hardware. The Executive provides the user program processing environment. Integral to the design of the Executive is support of batch, transaction, and interactive modes in both multi-programming and multi-processing environments.

Input/Output Control

As one result of Sperry Univac's commitment to enrichment of the Series 1100, a major effort represented in both hardware and software was undertaken to provide improvement in the efficiency and flexibility of system input/output management. This has resulted in a new, very general software control and management structure which controls all I/O operations for all 1100 Systems. In addition, Series 1100 hardware provides new flexibility and throughput improvement through the availability of both word channel and multiplexor channel capabilities. There are two main efficiency benefits in the new software system. First, relatively few machine instructions are executed by the Executive to carry out each I/O function. Second, the time which elapses before the next I/O request can be initiated after completion of a previous I/O operation has been minimized.

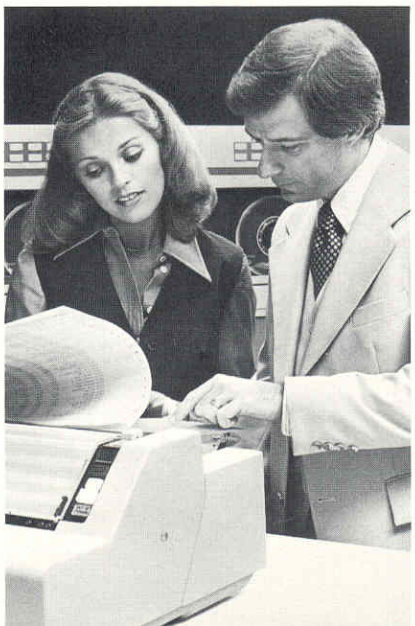
System Management

Due to the increasing size and sophistication of Series 1100 installations, there has been a corresponding increase in the requirement for effective tools for Executive system management, upgrade and testing. In response to this need, Sperry Univac has developed a comprehensive set of tools for the Executive system's management staff. Specifically, the main components of this set are:

The symbolic declaration of the 1100 system configuration, including any attached communications networks, is supported by a very powerful general-purpose symbolic stream generator (SSG). As a part of 1100 development, a new set of SSG programs is provided to more fully support and facilitate the management of large and dynamically changing system environments.

The QUOTA System is particularly useful in that a site administrator may limit the time, memory, mass storage and other system resources available to users, thereby insuring that no single user can preempt the entire system. The QUOTA system provides a method for controlling the entry of RUNS into the system and of controlling the RUN itself thereafter. Several levels of access control, accounting control and quota enforcement are provided.

An entirely new system facility (the FLIT process system) has been developed to provide the capability to execute, and diagnose the Executive while running as a normal user under Executive control. Thus, a new version of the Executive, or a planned new configuration, can be studied



and tested in a "virtual" environment, prior to its use as the production Executive system.

Due to the increasing importance of communications networks, the Series 1100 Communications Simulator (CS 1100) has been developed to provide the ability to run any communications configuration in simulated mode within the 1100 central system. This allows investigation of communications configurations and environments without the need of communications hardware and real communications traffic.

Finally, the 1100 Executive is itself fully instrumented so that it can dynamically monitor its own activity. This Software Instrumentation Package (SIP) provides a very important tool for system throughput and response optimization.

These enhancements constitute a major commitment in terms of development effort. They are part and parcel of the "evolution" found in SPERRY UNIVAC Series 1100 systems.

System Processors

The next level of SPERRY UNIVAC Series 1100 software consists of the various systems processors which are available to all users. These processors have followed an evolution paralleling that of the Executive. New hardware or Executive facilities and new user needs result in new enhancements to these products. Included in this category of software are facilities to modify and maintain files, provide system and terminal security, produce back-up file copies, and provide diagnostic information on the status of the system and programs upon termination.

Terminal Security System (TSS)

It is appropriate to start with the Terminal Security System as this system provides the initial level of system security and validation of user access. The TSS processor allows each site to select and tailor the features in the security system which are appropriate. The Terminal Security System may be used to simply validate the USERID/PASSWORD information provided by each terminal when it is logged on. However, in more secure or restricted environments TSS may be used to impose strict controls over the kind of access allowed to individual terminals, or terminal users. Thus, for instance, a given user may only be allowed to use one specific account number and he may be restricted to a specific set of files. Control over the TSS environment is provided through privileged use of the TSS processor by the site security officer. He may dynamically modify any of the security controls at any time.

Sentry

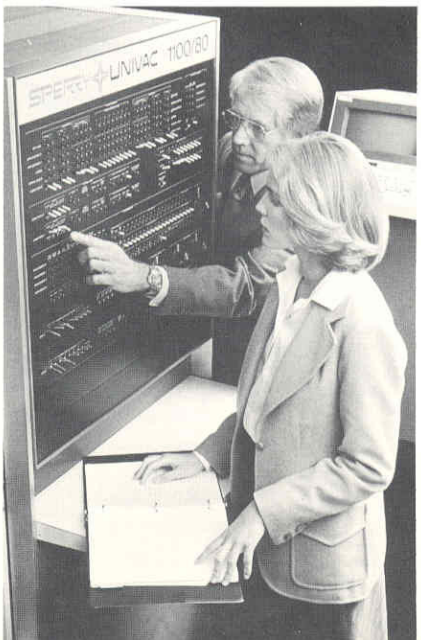
The Security Control Processor is used to create and maintain a user security profile data base. This data base is then used by the Series 1100 Security System to control user access to files and certain privileged functions. Validation of statements is provided as well as the capability to obtain various security type reports.

Quota Input Processor (Quip)

Quip provides the installation manager with an easy-to-use interface to control the utilization of system resources by individual users or groups of users. It defines users to certain accounts, any privileges granted, limits of system utilization and accumulated usage of the system to date.

Checkpoint/Restart System (CKRS)

CKRS has been developed to provide assurance to the user that his run or program can be safely restarted due to hardware, software or program failure. It "snapshots" a run or program and creates a checkpoint that may be used for restarting at a later time if desired. The checkpoint may be initiated within the user's program or at the Operators Console. Full recovery includes positioning of data files.



Define File Processor (DFP)

DFP provides a data file description *external* to the program processing the file. This allows program flexibility, in that file information usually imbedded within a program is now kept as a separate entity. Using DPF, programs written with processors such as FORTRAN, COBOL, PL/1, APL, SORT and RPG are now file-format independent and can thus share common files. It can be used when changing complete files or program data file descriptions without the usual requirement to recompile the program.

DFP may also be used in conjunction with the tape labeling system to enter information such as file identifier, expiration date, accessibility code and date structure.

File Administration Processor (SECURE)

The SECURE processor protects the physical security of cataloged files, which reside on mass storage, by producing tape backups. At periodic intervals, each file is examined to determine whether any write operations have occurred since its previous backup was saved, in which case a new backup must be made.

Integrated Recovery Utility (IRU)

Integrated Recovery Utility is an integral part of the Series 1100 Integrated Recovery System. Simple English commands are provided for the user to initiate a variety of integrity features and capabilities involving data base roll back and data base reconstruction.

Reliable recovery techniques include the Save and Restore of TIP or TIP/DMS 1100 files.

IRU can be used to control user access to selected TIP or TIP/DMS files, legs of duplex files or even partial file access. It also may be used to compare complete or partial records between files or files with like attributes.

Memory Allocation Processor (MAP/COLLECTOR)
The collector provides for collecting and interconnecting relocatable elements to produce a program in an executable form. This form is called an absolute element. Optionally, the collector can be used to create a single relocatable element from a collection of several relocatable elements. These relocatable elements may be a mixture created from Assembly language or any of a number of higher level language processors or library routines retrieved and linked in the program by the collector.

Postmortem Dump Processor (PMD)
The Postmortem Dump Processor serves as a user debugging aid. As such, it produces edited dumps of the contents of main storage if the program terminates in error or, as an option, when a program terminates. Dumps produced dynamically during program execution are automatically printed. Individual program parts are identified with the assistance of diagnostic tables produced with the absolute program by the Collector.

In addition, PMD can be used to produce a changed-word list that points out any word in memory changed during program execution.

Element Processor (ELT)
ELT is used to insert symbolic, relocatable, absolute or omnibus elements into a program file from images in the runstream. ELT will also update a symbolic element already in a program file and may be directed to produce a new symbolic element with a new name or an updated element with the same name.

Procedure Definition Processor (PDP)
PDP processes symbolic elements which may contain Assembler, FORTRAN, or COBOL procedures and it produces entries in the table of contents of a program file. These entries are used by the assembler and the FORTRAN and COBOL language processors to locate the individual PROC in the program file.

PROCS may contain table definitions, commonly used code, or Assembler procedures (macros). This provides a powerful tool for effective program structuring.

Symbolic Stream Generator (SSG)
The SSG processor is a general purpose generator manipulating a variety of symbolic streams. Directions and models for building the desired stream images are conveyed to SSG through a skeleton (program) written in SYMSTREAM. The resulting symbolic output streams may be placed in a user specified file and/or printed and/or dynamically added to be executed after SSG terminates. SSG also helps maintain special symbolic input files which may be printed, corrected and/or updated for later use. The primary purpose of the special symbolic input files is to maintain corrections to be applied to a symbolic element. By using SSG, these sets of corrections may be changed (updated), merged with another set of corrections printed in the symbolic output stream, and/or inserted back into the original input file for future use by the programmer.

File and Program Utility Processor (FURPUR)
FURPUR consists of a set of file maintenance routines which provide the flexibility in management and manipulation of cataloged or temporary files containing data or programs.

Data Processor (DATA)
The DATA processor provides data handling capabilites, at the file level. DATA is used to create, update and list System Data Format (SDF) files which reside on mass storage or tape.

Text Editing

Edit 1100
Edit 1100 is a modern, easy-to-use, full-screen text editor to be used with IPF 1100. It allows the user to intermix editing commands and IPF 1100 commands. It takes advantage of full-screen buffering and standard editing features inherent with

all of Sperry Univac display terminals. It will utilize intelligent display terminals to further enhance ease-of-use and performance.

In addition to the full-screen editing capabilities, Edit 1100 provides for general editing commands to be used on either screen devices or non-screen devices, such as teletypes.

The editor also functions as a stand-alone processor and it can be called from within a program. Thus application programs may also make efficient use of the editor.

The commands and display forms of the editor are easy to learn and remember. Assistance in recalling the editor's features may be gotten by using the HELP command.

Utility Processors

CULL Processor
The CULL Processor produces an alphabetically sorted, cross-referenced listing of all symbols in a specified set of symbolic elements. Each symbol processed by CULL may contain up to 12 alphanumeric characters plus the dollar sign. Provisions are included, via options to selectively include or exclude defined symbols from the output.

Documentation Processor (DOC)
DOC is an automatic documentation system which produces formatted documentation output from specifically prepared input data.

DOC accepts input from images in the runstream or from a symbolic element in a program file. The output is a formatted document and optionally an updated symbolic element in a program file.

Fault Location by Interpretive Testing (FLIT)
The FLIT processor is a program which provides the system user with the ability to test and debug absolute program elements on an interactive basis. In much the same manner as would be used in small system installations, FLIT makes console type features available to users through simulation and interpretive techniques. Its design was aimed at deriving all the benefits of console debugging while at the same time removing the high cost factor of machine block time.

Two modes of operation are provided: System mode and User mode. The first provides for the testing of operating systems in a simulated system configuration.

In the User mode, absolute program elements can be executed, traced, debugged, and edited.

FLIT provides a programming capability to make it possible for commonly used functions to be programmed and saved in libraries.

FLIT also provides a program analysis capability which accumulates data about frequency of use and time spent in each basic program block. A FLIT post-processor prepares a summary report concerning the program execution. This report identifies high-usage program areas and provides a clear guideline for program optimization.

To provide high degrees of control, FLIT executes all instructions interpretively. The results of all instruction executions are kept in a set of pseudo registers and not directly in the corresponding actual registers.

Simulated memory is maintained through a paging algorithm on a mass storage file. Consequently, the size of FLIT is constant and does not depend on the size of the program being simulated or tested.

SORT/MERGE
In addition to the sort/merge subroutine, there is a free-standing sort/merge program, SORT/MERGE 1100. SORT/MERGE 1100 provides three sort options and a standard merge option. The sort options are:

- Record Sort
- Selection Sort
- Tag Sort

SORT/MERGE 1100 provides the following features for all sort/merge options:

- Up to 26 files may be merged
- Up to 40 keys may be specified
- User own code exits
- Tapes may be labeled or unlabeled
- Automatic checkpoint-restart capabilities
- Free format parameter cards
- Optional sequence checking of input/output data.

Log Analyzer (LA)
The LA is an easy-to-use, easy-to-install user program that will assist a customer in monitoring the resource utilization of a Series 1100 system. System and user activity data analyzed by LA is dynamically collected by the 1100 Operating System. LA will output this data in the form of comprehensive reports to pinpoint bottlenecks to substantiate possible configuration adjustments.

Performance Analysis Routines (PAR)
PAR is a reporting system for data collected by the Software Instrumentation Package (SIP) embedded in the operating system. SIP is a measurement tool that provides a total system performance analysis capability, gathered optionally for different levels of analysis. SIP does this with minimum impact on system performance.

PAR is a stand-alone software package provided for analyzing and generating reports from the data gathered from SIP. PAR provides a selectable set of predefined reports, as well as the capability to be programmed in a user language to perform arithmetic operations on the data and produce additional reports. It also has the capability to produce performance graphs.


SPERRY
UNIVAC
1100/80
SYSTEM CLOCK
PROCESSOR 0
PROCESSOR 1
PROCESSOR 2
PROCESSOR 3
OFF LINE
APPLICATION 0
APPLICATION 1
CHECK
POWER CHECK
MAINT MODE

Communications Processing

Communications Management System (CMS 1100)

CMS 1100 is the software that manages all communications for the SPERRY UNIVAC Series 1100 systems. It provides common support for all on-line user modes—transaction, interactive and remote batch—in either centralized or Distributed Data Processing networks.

One of the principal features of CMS 1100 is the independence of the terminals in a network. The terminals interface to CMS 1100 and not to an application or a particular function. A user may interface to the time-sharing system at one moment and the next moment interact with the query/update package.

Maximum flexibility—with ease of use—is provided by generating the CMS 1100 network separately from system generation and initialization. This independence allows the entire terminal network configuration to be generated, checked and corrected, without generating a full system. This method provides the capability to add, subtract, or replace entirely the network configuration dynamically, reinitialize CMS 1100 and proceed with the new configuration in effect. Another ease of installation and use feature is that multiple copies of CMS 1100 may exist in a given system. This allows support of the system's production environment with the "standard" CMS 1100, while allowing live testing of a new "development" level of CMS 1100. Finally, CMS 1100 maintains a logging facility to provide information about the communication environment it is saving.

By maintaining a user transparent interface to CMS 1100, the user can move the network control from the central processing unit to a front-end processor as communication requirements expand. In addition, networking capabilities will be provided to terminals connected through the non-programmable communication controller. For example, a terminal connected to an 1100 through a non-programmable communication controller (GCS) can establish a session with a terminal connected to the 1100 through the front-end processor (DCP) network. This will provide uniformity in operation and functionality.

CMS 1100 utilizes a modular, top-down, structured design following the basic structure design defined by the Distributed Communication Architecture (DCA). A prime objective of CMS 1100 is to provide the Series 1100 communication interface for DCA. It provides the I/O interface for the Distributed Communication Processor (DCP) as a "node" in a Telcon network.

DCA is the communication network base for Distributed Data Processing (DDP). CMS 1100 is the communication interface for the Series 1100 system to a DCA based distributed data processing Telcon network.

Processor Common Communication System (PCCS 1100)

PCCS 1100 provides a means by which application programs developed in high level languages such as COBOL and PL/I may utilize the Series 1100 communications system. This capability enables COBOL and other languages that interface with PCCS 1100, to meet the ANSI standard for message communications.

Programs using this mechanism may communicate to/from other communication programs, terminal users, remote batch systems and certain host computers. Terminal users also have the capability to communicate with each other.

Transaction Processing

Series 1100 Program Products for transaction-oriented data base data communications processing include the following:

- DMS 1100
- DDS 1100
- QLP 1100
- RPS 1100
- TIP 1100
- DPS 1100
- MAPPER 1100

Data Base Management System (DMS 1100)
This is a CODASYL-type implementation for data base processing. The structure of the data base is a network described by the administrator with schema and sub-schema languages. It is also maintained with utilities for reorganization and recovery.

Manipulation of the data base is a separate process from the definition phase. Programmers use data base manipulation commands embedded within COBOL, FORTRAN or PL/I programs to access the base. This separation permits re-structuring for system throughput optimization without requiring the applications programs to be re-compiled.

Accessing techniques used in DMS 1100 include sequential, index-sequential, direct, CALC and set relational.

Data base protection is provided through several levels of recovery. A run-unit can recover from an individual command receiving an error indication, by invoking a "roll-back" of that command, thereby, cancelling its effect on the data base. A run-unit can also establish recovery points, so that a series of commands can be rolled back. In the event of a system error, all run-units executing at the time of the error are rolled back and re-started from these recovery points. Should an individual device fail, containing a portion of the data base (such as a disk), that device is recovered, leaving the rest of the system available for accessing. Total recovey can be made using back-up and audit trail of the data base.

Data base security is provided by system, file and data base access keys, sub-schema design and individual item access and coding techniques.

DMS 1100 provides users with a comprehensive set of data base management capabilities. It has been used in a production environment since 1972 and is presently installed at many sites throughout the world.

Data Dictionary System (DDS 1100)
The Data Dictionary System provides a means for centralized description, location and control of the various elements within a user data base environment. The DDS provides the user with facilities which can be used to describe data so that its representations and its intended use in the real world is clear. It provides through one mechanism a means to describe and locate the relationship between data users and the data base by:

- Descriptions of the user data entities (files, records and their component items, and relationships).
- Descriptions of the application programs (run units and transactions) and processes (runstreams) which access the data entities.
- Descriptions of the user entities and relationships in the environment which employ the data entities, application programs, and processes, providing control and improving productivity.

The Data Dictionary System consists of:

- a data base of information (called the meta-data base) about the entities in the user data base environment
- a set of processors which access the meta-database for the purposes of creating, updating, and reporting information.

Query Language Processor (QLP 1100)
QLP 1100 is an interactive facility with generalized features selected and specified through the use of a natural, English-based command vocabulary. The vocabulary is both simple and rich in content, meeting a range of needs.

QLP 1100 provides access to data using either data base files (DMS 1100) or traditional files. Traditional files are those created by language processors, i.e., COBOL and FORTRAN, and are in forms such as sequential, indexed sequential, or multiple indexed sequential. The use of traditional files allows the use of existing files without restructuring or reprogramming.

The facilities in QLP to access data are summarized as follows:

- Display information within a record or a set of selected records
- Update information within a record or a set of selected records
- Create new records
- Deletion of records.

Procedures may be parameterized so that one procedure may serve many kinds of requests. QLP 1100 can be a convenient alternative to writing programs.

Meaningful information is provided by the use of the table translation feature. Instead of codes being displayed that have no inherent meaning, you can specify that you want QLP 1100 to translate the codes into information understandable to you.

A comprehensive list of simple commmands are used to specify what action is to be taken (list, count, change, delete, etc.). The criteria to be used for selecting the action is given through an added clause (WHERE). If desired, a series of commands may also be pre-stored and recalled, providing a procedural language facility.

QLP 1100 also provides a comprehensive report writing facility. Reports are defined by the user, complete with headings, columns of data, and margins conveniently arranged in a meaningful and useful form. Multiple varied reports can be produced with a single pass of a data file.

Remote Processor System (RPS 1100)
This general-purpose data management and file processing system offers a "menu" of standard functions to a user at a display terminal. Interfacing with the system, users can browse through files, sort and search them, build new files, update existing files and perform other types of traditional file processing.

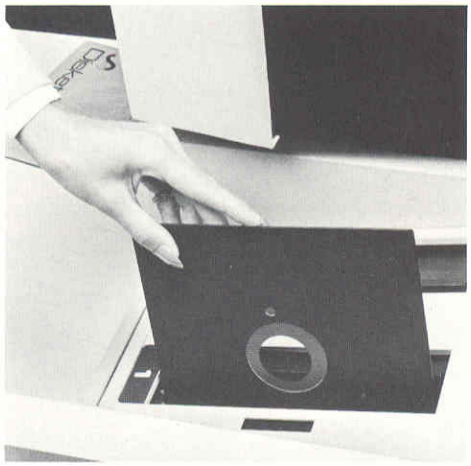
RPS 1100 guides the user at each step, soliciting information such as search identifiers or sort criteria, as required.

An important capability of the system is the Tutorial Processor. It permits functions to be stored in a sequence to accomplish any application comprised of the standard functions. This sequence can then be invoked when needed. In this way, clerical personnel can interact with the application without having any knowledge of the functions or the sequence. The sequence itself is described to the system through the terminal; the application designer does not "program" in the normal sense of the word.

Time and costs are dramatically reduced by using this form of applications development.

Transaction Interface Package (TIP 1100)
TIP 1100 provides the Series 1100 systems with an efficient on-line communications-oriented capability that operates under control of the Executive system. Transactions can be processed concurrently with demand and batch operations, thereby minimizing computer use by equalizing system loading. TIP 1100 is extremely valuable for the inquiry, on-line data entry and communications-oriented user. The package is streamlined, modularized, application-independent and fully compatible with the Executive. For users now operating primarily in batch mode, TIP 1100 provides a simple vehicle for expanding a system into an on-line transaction-processing environment. The flexibility of TIP 1100 lets it interface readily with DMS 1100.

TIP 1100 is proven, operational and available.



Information Management System (IMS 1100)
IMS 1100 is a file processing transaction interface for programs developed in COBOL or RPGII. IMS 1100 is compatible with the IMS 90 interface available on SPERRY UNIVAC Series 90 systems, providing functional equivalency and source language compatibility. It is an alternative useful in migration to the Series 1100 from Series 90.


SPERRY UNIVAC

Display Processing System (DPS 1100)

DPS 1100 increases programmer productivity and provides a measurable improvement in the flexibility of program design, development, and maintenance. DPS 1100 is a multi-functional product, providing in the first case interactive development of screen formats for interactive programs. DPS 1100 will generate the screen program source code for the programmer, streamlining the programmer's workflow and speeding up task completion.

DPS works with high level programming languages, such as COBOL. At execution time DPS 1100 continues to provide valuable services, since it intermediates between the terminal and the program, essentially making the program independent of terminal type and features.

DPS 1100 will support TIP 1100 and demand programs concurrently, or support a program which moves from one environment to the other. By providing security features and pre-processing error checking, DPS can improve overall system efficiency and integrity.

Maintaining, Preparing and Producing Executive Reports (MAPPER 1100)

MAPPER 1100 for the Series 1100 systems is a general purpose, on-line report processing system that uses UTS 400/4000 or UNISCOPE 100/200 Display Terminals with auxiliary print/storage devices.

MAPPER 1100's form generation capability allows implementation of databases and related report processing without applications programming.

MAPPER 1100 uses a report-structured database. The system is capable of free form and column-formed report entry, storage, retrieval, real-time update/change, display, and hardcopy output. Report processing capabilities include searches, sorts, matching, and computation (totalizing). Functions for data analysis, character string location and change, and FORTRAN-like equation solving are also available.

With MAPPER the user can conveniently begin and modify reporting formats as his experience grows. In addition, complete database management, recovery, and historical access capabilities exist for the user.

Where repetitive relationships of functional use exist in MAPPER 1100, a custom MAPPER application (Run function) can be designed and used. The application may be designed to be used in a tutorial mode allowing sequential interruption for display and interactive control. The selected functional sequence of a given application may be interspersed with logical decisions. The decisions may affect functional jumping, branching, or data content. A MAPPER 1100-to-MAPPER 1100 remote application capability makes intersystem data linking possible.

Other MAPPER functions include a station-to-station message delivery service and a start and retrieve interface to the 1100 batch environment.

The power, ease of use, and flexibility of MAPPER 1100 make this program product adaptable to the real-time report processing requirements of almost any industry or institution.

Interactive Processing

Four modes or levels of Interactive processing are supported on all 1100 systems. These are:

- Demand
- CTS 1100
- HVTS
- IPF 1100

Demand Interface

The 1100 Operating System will interface directly with a user at an interactive terminal. All system functions, all processors and the job control language available to the batch user are available to the demand user. The system automatically provides a higher processing priority to the demand user, and directs card reader and printer I/O to the demand terminal. In this manner, a program may execute in batch or demand mode with no modifications. All system processors and utilities may be utilized in the demand mode.

Conversational Time Sharing (CTS 1100)

CTS 1100 has two goals:

- Providing a man/machine interface for the interactive user in a language to which he is accustomed.
- Disciplining the interactive user to efficiently use the system with respect to storage and processor utilization.

The main module, essentially becomes a monitor under the 1100 Operating System and controls the activities within the interactive environment, calling specific processors as needed or stipulated by the job control. The time-sharing monitor includes the English-language interface for total text editing capabilities on programs or data.

The command language includes all the attributes of text editing, file handling and other systems-oriented commands.

A total interactive debugging capability is incorporated, including pause and testing variables. A procedure file, or subroutine capability, enables a user to fully program at the command structure level. The user can declare variables, develop expressions looking similar to FORTRAN, effect decision branching (including looping) and perform input and output . . . all at the command level.

High Volume Time Sharing (HVTS)

The HVTS product provides an alternative to the conventional time-sharing system, CTS 1100. The benefits derived from this system lie in its orientation towards concurrent handling of a large number of active terminals. This specially designed high performance system optimizes resource utilization by the selection of critical control elements for retention in resident storage, and by forcing the scheduling and time allocation elements to operate in a real-time mode.

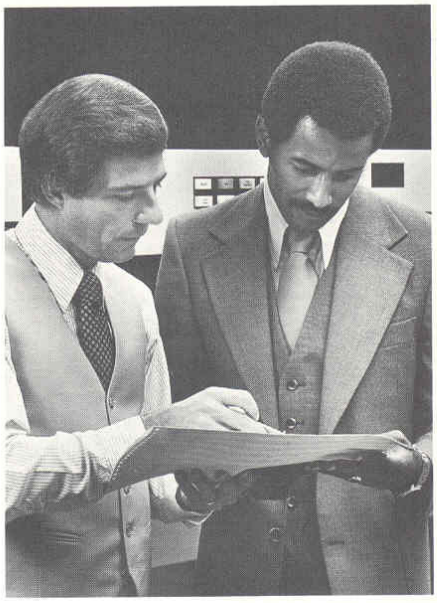
HVTS has been designed to handle configurations in the range of 50-2000 terminals, each operating in any of six modes. Language modes include APL, BASIC, and FORTRAN. These allow the development and execution of programs within that language mode.

Interactive Processing Facility (IPF 1100)

IPF 1100 is a full capability user interface that supports both time-sharing and batch operations. It was developed to support user requirements for functionality, performance and ease-of-use. IPF 1100 interfaces with the system according to the user's level of sophistication. In addition to a Command/Response language, an execution environment is provided to support all language and system processors as well as user developed programs.

With IPF 1100, the user can fully utilize the system. The command language is keyword oriented, increasing both user and system performance. Simple but powerful commands permit building, editing, and executing a program *concurrently* with a task or tasks previously started.

Further increasing user performance, IPF 1100 uses an efficient file access method used by the full screen editor, and provides access to compilers, system processors and application programs.



The Interactive Processing Facility provides the functionality of Demand and CTS processing combined with the large network capacity of High Volume Time Sharing.

There are eight major modules of IPF 1100

- DDP 1100
  The Distributed Data Processing module of IPF allows file transfers and job submissions from Series 1100 to Series 1100 systems.
- COMMAND LANGUAGE
  The primary interface for using IPF is a command language based on the CODASYL Common Operating Systems Command Language specifications.
- USER ASSISTANCE
  The module of IPF which manages both reponses to the terminal user and HELP and explanation processing.
- PROCEDURES
  The development of command language subroutines and macros is accomplished through the use of IPF Procedures, which provide a high level programming capability.
- CONSULT MODE
  This mode permits an IPF expert to consult or teach another IPF user by linking the terminal of the expert with the terminal of the user who requires assistance.
- MAIL
  IPF provides a user to user mail feature based on a store and forward design.
- CTS COMMAND ANALYZER
  To provide for ease of migration for users of the CTS user interface, IPF provides an optional command language interface that is compatible with that of CTS.
- EDIT 1100
  The input and update editor is EDIT 1100, which provides access to a variety of file formats, works in an easy-to-use full screen mode, and can be used from a terminal or called from a program.

Language Processors and Libraries

Sperry Univac has been in the forefront both in development of high-level languages and in participating in a number of committees chartered to reach agreement on industry-wide standards. We are committed to the concept of industry standards because we feel our customers benefit from them, and because adherence to industry standards simplifies our development effort as well. As a result, PL/I is available to our customers, and a Macro Processor has been provided to extend and create new facilities for languages such as PL/I, APL, and ASCII FORTRAN.

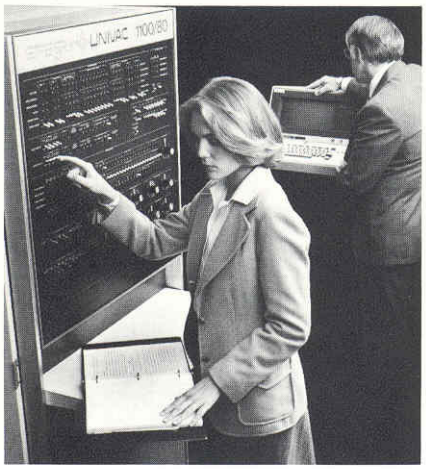
To provide the best systems for the Series 1100, we have formulated an internal standard for all new language processors with respect to the characteristics of the products and the interfaces with other parts of the Series 1100 environment. This insures compatibility, commonality and parallelism across all new language processors.

Some aspects of the Series 1100 standard within which new products have been developed are these:

- ASCII CODE
  The emphasis on code sets has switched from FIELDATA to ASCII in our new processors. The byte instructions and operating system modifications all complement this feature in the new processors.
- RE-ENTRY AND REUSEABILITY
  The new processors are designed to fully capitalize upon all the features of the 1100 Operating System. This means that the processors are re-entrant and reuseable and generate re-entrant code. These features contribute to significant improvement in the total 1100 Operating System efficiency.
- COMMON I/O and Libraries
  The capability of interchanging data files between different language processor executable elements is highly desirable.

  Our approach is to encourage this interchangeability between as many language processor elements as possible. Not only does this mean common data formats and access methods, but the full advantage of common banks as employed by the operating system. Common banks are also used in the formation of a common universal math library.
- OPTIMIZATION
  All compilers for new languages contain an optimization module which will insure efficiency code generation. Smaller and faster run-time systems are the result. The area of optimization has during the last few years received much research and study at Sperry Univac. The techniques we have developed for FORTRAN and other languages provide the best optimizing compilers in the industry. Since the optimization phase of the compilation processes is expensive, compilers are parametered to include optimization only as an option specifiable at compilation time.


1
2
SPERRY UNIVAC
UNIVAC



□ CUSTOMIZATION

Not only is optimization by selection of parameters included at compilation time, but the same capability in other areas of the compilation processes is also considered important. Our new processors are used on systems which can vary greatly in availability of memory and other computer facilities. The use of the systems and the relative importance between batch and demand, real time and batch, etc., varies greatly as well.

Therefore, instead of tailoring the processors to certain types of systems or users, we have designed the processors so that—through customization—they can be tailored to specific installations and users.

The following product provides the capability of customizing processors.

The General Syntax Analyzer (GSA 1100)

GSA 1100 is a front-end tool that can be used in the construction of a broad range of processors, text editors and other applications that require the scanning of a language-type input. This is accomplished by the language builder supplying language and structure specifications to the different processor building blocks of GSA. The resultant tables are then merged with the GSA modules to produce the actual user processor.

GSA 1100 can also be used to update or customize existing processors for increased functionality and compose error, warning and other messages that are normally printed upon user termination from a processor.

Following are brief descriptions of the languages, debugging aids and libraries available:

COBOL

COBOL processors provided with the SPERRY UNIVAC Series 1100 Operating System are:

□ Series 1100 American National Standard COBOL (ASCII).

□ BCOB (COBOL Syntax Analyzer).

American National Standard COBOL (ASCII)

SPERRY UNIVAC Series 1100 American National Standard COBOL (ASCII) is in compliance with the American National Standard COBOL X3.23—1974 with major CODASYL and SPERRY UNIVAC based extensions, and with FIPS-PUB 21.1. The compiler complies fully with the highest levels of all modules of the American National Standard COBOL and recognizes ASCII characters as the standard data mode at both source and object time. Fieldata code handling capabilities are available by option.

COBOL Syntax Language (BCOB)

A COBOL language syntax prescan processor, BCOB runs as a fully reentrant CTS 1100 (conversational time sharing) sub-module. As such it also supports the full CTS command set. The purpose of BCOB is to aid the timesharing user in constructing, editing and syntax debugging COBOL programs from a terminal. Its syntax analysis is compatible, at user option, with either the Filedata or ASCII compilers, and includes the respective data manipulation languages as well. The syntax scan is performed first conversationally line by line as the program is entered, and then again on the program as a whole for a more complete analysis.

FORTRAN

FORTRAN processors provided with the Series 1100 Operating System are:

□ ASCII FORTRAN

□ BFTN (FORTRAN Syntax Analyzer)

Other FORTRAN processors are supported on Series 1100 systems for compatibility purposes.

ASCII FORTRAN

The ASCII FORTRAN system is an extension of FORTRAN V. The system supports the existing FORTRAN standard language, most FORTRAN extensions of the standard as well as extensions proposed in the forthcoming FORTRAN standard. The term 'ASCII' is applied to this system since it provides the full ASCII character set for use in FORTRAN applications.

ASCII FORTRAN for the Series 1100 Operating System consists of an extended FORTRAN language, a library of subprograms, and a compiler designed to provide an extended, industry and standard compatible, FORTRAN system utilizing the features of the Series 1100 system in an efficient and complete manner.

The FORTRAN language has proven particularly useful for programming applications that involve mathematical computations and other manipulation of numerical data. ASCII FORTRAN significantly extends the capabilities, efficiency, and usefulness of the FORTRAN language, providing extensive manipulation of both numeric and nonnumeric data.

Prescan FORTRAN (BFTN)

BFTN is a reentrant FORTRAN syntax analyzer used in conjunction with SPERRY UNIVAC Conversational Time Sharing (CTS). BFTN, like all the source language prescan units available with CTS, gives CTS users the capability of programming in a high level source language, such as FORTRAN, in a conversational environment.

Programming Language I (PL/I)

PL/I is a programming language designed to cover as wide a range of programming applications as possible. The language is designed to reduce the cost of programming, including the cost of training programmers, the cost of debugging, and in particular, the cost of program maintenance.

Two basic characteristics of PL/I are intended to reduce the need to rewrite complete programs if either the machine environment or the application environment changes. These characteristics are the block structure used in the language and its machine independence.

The variety of features provided by PL/I, as well as the simplicity of the concepts underlying them, demonstrate the versatility of the language, its universality, and the ease with which different subsets can be defined to meet the needs of different users.

PL/I is in compliance with ANSI standard X3.53—1976.

Processor Common Input/Output System (PCIOS)

The Series 1100 Processor Common Input/Output System (PCIOS) is used with the ASCII COBOL, ASCII FORTRAN, PL/I, SORT, APL, RPG, and QLP processors to produce compatible data files. PCIOS supports a variety of access methods including sequential (for tape and for random access mass storage), direct, indexed sequential (single key and multiple key). Each access method is provided by an I/O module which is common to all languages which support that particular access method. Therefore, files written by an application developed in a selected programming language may be read/updated by applications developed in different languages.

The Define File Processor (DFP) is a stand-alone processor developed in conjunction with PCIOS as a means of providing a data file description external to the program that is processing the file. The Define File Processor produces a define-file-block which contains the data file description and enters this information as an element into a file for subsequent reference at run time. The program actually processing the data file references it through an access method supported by PCIOS. When this program opens the data file, the Processor Interface Module associated with the processor will retrieve the define-file block and use its contents as the file description.

The Define File Processor allows alterations to file descriptions to take place without recompiling or recollecting the program(s) which access those files.

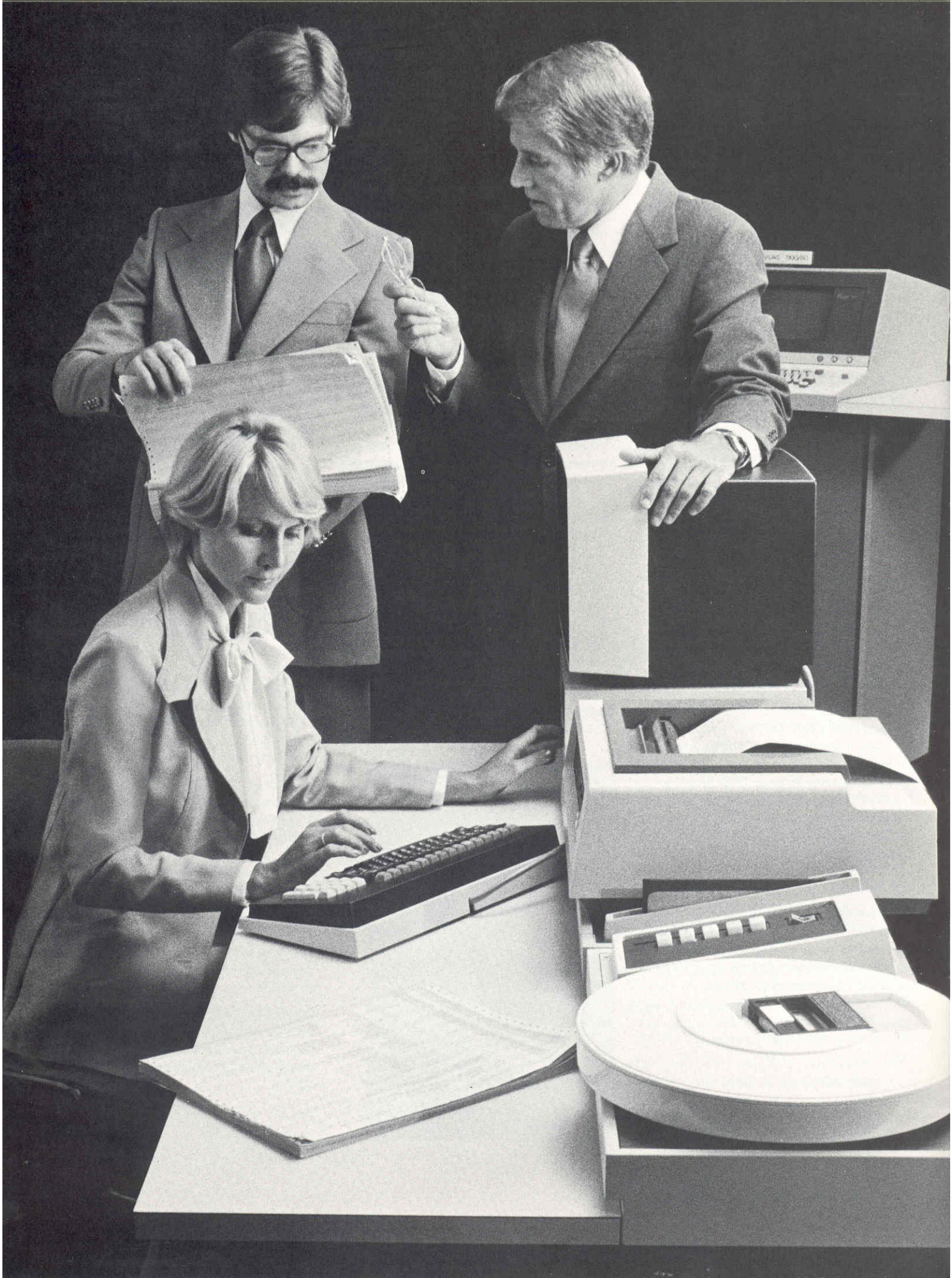
Meta-assembler (MASM)

Assembler code processing on the SPERRY UNIVAC Series 1100 is carried out by means of a meta-assembler (MASM). This processor allows the programmer to define the language to be assembled and to define the resulting object code produced by MASM. Thus, while MASM is normally used to assemble Series 1100 instructions to be executed on the Series 1100 system, it may also be used to assemble other machine languages for execution on other processors.

In addition, MASM has substantial high level constructs which facilitate efficient and well-structured programming. MASM recognizes conditional constructs, variable types (including strings), controlled scope of definition of identifiers, and includes powerful diagnostic tools and full micro and macro capabilities.

The usual character set recognized by MASM is ASCII; however, other standard or user defined character sets may be specified.

Special facilities are available in MASM to provide very high efficiency for large program systems with many modules which share common procedures and tables.



MACRO

MACRO is a general purpose macro processor designed to extend nonspecific host languages through the ability to process character strings. Its capabilities include, but are not limited to, a powerful text translation language suitable for extending programming languages, translating programming language dialects, text generation, text editing, and text validation.

The MACRO language is easy to learn and use, yet powerful with generality and simplicity being emphasized. Its definition is algorithmic and higher level. It is, therefore, particularly easy to learn by those programmers who previously have had experience with such languages as FORTRAN and PL/I.

MACRO consists of two separate processor modules, one which compiles macro programs into an internal library format, and a second which uses the stored output of the compilation module to perform the actual text transformations. This structure allows an immediately usable form to be generated and stored in a program file, but yet allows the efficiency of separation of the debug/use functions.

UNIVAC BASIC (UBASIC)

The SPERRY UNIVAC Series 1100 BASIC Processor (UBASIC) is designed primarily for conversational use. Written with the novice programmer in mind, the UBASIC language consists of instructions which are largely self-explanatory, and the syntax is closely related to normal English. The UBASIC Processor can be used for batch processing, and is powerful enough to solve a large class of problems.

The UBASIC Processor reads statements (or lines) and immediately examines each one for syntactic correctness. On remote devices in conversational (demand) mode, syntactic errors in a line cause a diagnostic message to be printed immediately, so that such errors can be corrected before the next line is accepted. When a complete program has been processed, then a RUN command may be given, at which time the program is translated to machine language (compiled), and execution of the program is initiated.

The prescanner BBASIC works under control of CTS. When appropriate prescan action is complete the UBASIC processor continues the processing of the program. BBASIC is closely associated with UBASIC. The BASIC language syntax is prescanned in a conversational mode by BBASIC under control of CTS. BASIC source language programs are prescanned for syntactical errors before they are submitted to UBASIC for processing.

APL 1100 (A Programming Language)

APL 1100 is a reentrant, interpretive processor which uses 9-bit ASCII code throughout and functions as part of the conversational time sharing (CTS) system.This processor provides a superset of the ad hoc industry standard APL language. The processor specially provides all language features of the Iverson notation and extends capabilities in the areas of I/O and operating system related functions. APL 1100 uses the facilities of CTS for text editing and file manipulation, thereby avoiding redundant code in the system.

The APL 1100 language provides a powerful tool for describing and communicating array-oriented algorithms in a time sharing environment.

Wide ranging applications presently provide a marketplace for APL 1100 in education, business, and scientific communities.

NUALGOL

NUALGOL is a language for communicating scientific and data processing problems to the Series 1100 systems. The basis for this language is the Revised Report on the Algorithmic Language, ALGOL 60 (Communications of the ACM, Vol. 6, January 1963, 1-17).

The NUALGOL language allows the mathematician or engineer to prepare programs for the Series 1100 system without the necessity of becoming familiar with the details of the internal machine operation.

Report Program Generator (RPG 1100)

The Series 1100 Report Program Generator (RPG) performs two general functions, program generation and data processing, to produce user designated reports and generate output files. Common report features such as input data selection, editing, calculation, multiple report files, summarizing, control breaks, and file updating are provided by RPG. Sequential files, indexed sequential files and table files can be processed.

Programmers Advanced Debugging System (PADS 1100)

The Programmers Advanced International Debugging System (PADS 1100) is a language independent debugging tool. It was designed mainly for debugging of programs written in high level languages such as COBOL, FORTRAN, and PL/I. However, it may also be used for programs written in Assembler or any other compilers which generate the symbolic dictionary tables according to its required format. Assembler language users may only use a subset of the features provided.

There are many significant advantages of using PADS 1100 such as:

- It requires neither debugging statements to be inserted into the source program nor collector directives to be used at collection time.
- Common command language provides a uniform way of gathering debugging information independent of the source programming language being used.
- It may be included within production programs and results in no execution overhead on successful production runs. It is automatically activated at program failure without any prior usage or action.
- It supports both batch and demand jobs, as well as programs utilizing TIP, DMS or LOAD and GO facilities.
- The command language is easy-to-use and learn for the novice yet provides considerable functionality for the more experienced user.

PADS 1100 commands are divided in three functional areas:

- Monitoring and controlling flow of contract within the program—The commands for monitoring and controlling the executing program's flow of control provide four basic functions. These commands establish execution steps within the program, transfer control from PADS to the program, terminate the program and the debugging session, and gather information about the executing program.
- Examining and modifying the program environment—The second basic type of commands within the PADS language deals with examining and modifying the program environment. This includes information about the contents of instruction and data locations and registers. Whenever the user has control in PADS, the contents of program variables may be interrogated.
- Controlling the debugging session—The third basic type of command within the PADS language provides control within the debugging session. These commands provide (1) alternate sources for PADS command; (2) declaration and manipulation of debugging session variables, procedures, and formats; (3) conditional and iterative statements; (4) help available in the form of command explanations; and (5) information about the execution environment.

When program testing is required the interactive user simply loads his executable program into memory, enters the PADS commands required to monitor execution at selected program locations, and begins execution. PADS permits the user to step through the program logic, stopping at specified locations to display key areas within the program. With AIDS commands, the user can modify program data fields and indicators to test all paths of the program logic without having to create elaborate test data. When errors are found the interactive user simply makes the necessary corrections and continues the testing process. PADS both speeds up the program debugging process and also—because PADS commands are so powerful and easy to use—insures that the finished program is more thoroughly tested.

LIBRARIES

The Series 1100 Operating System includes the following libraries:

- System Library
- MATH/STAT-PACK

System Library

The system library contains relocatable subroutines provided by the operating system. Subroutines referenced by user programs are automatically included in the absolute element constructed by the collector. The system library contains routines in the following general categories:

- System Procedures and Definitions
- Dynamic Dump and Program Trace Routines
- Editing Routines
- Processor Interface Routines
- Collector Interface Routines

Mathematical and Statistical Package (MATH/STAT-PACK)

MATH/STAT-PACK is a comprehensive library of mathematical and statistical routines coded in the FORTRAN language. They are designed to furnish many of the most frequently used tools of numerical and statistical analysis in an easily useable form.

MATH/STAT-PACK is entirely self-contained, requiring only the standard FORTRAN Library subroutines for implementation.

The subprograms have been designed to facilitate usage by the inexperienced as well as the sophisticated FORTRAN programmer. The programmer need be concerned with only a small number of familiar terms, not the internal design of the subprograms. The object has been to minimize the program preparation required prior to calling a subprogram.

The subprograms are completely independent of any specific input or output device.

Applications

UNIS 1100 Industrial System

UNIS 1100 Industrial System is an integrated, computer based system for manufacturing control. The system is designed to handle problems common to manufacturing companies... from the creation and projection of a demand schedule based on customer orders and forecasts... through the development and evaluation of a master production schedule via material and capacity requirements planning... the procurement, production and shipment of products.

Utilizing a comprehensive data base for the integration of manufacturing control data, the UNIS 1100 System provides multiple solutions to manufacturing control problems through a series of integratable, modular, application software functions and executable programs oriented to production and inventory control applications.

The structure of the UNIS 1100 functions, the executable programs and the comprehensive data base, provide superior flexibility to implement a manufacturing control system or to enhance and expand a company's existing system.

UNIFACS 1100

The SPERRY UNIVAC Financial Accounting System includes a General Ledger/Budgeting System, an Accounts Payable System, an Accounts Receivable System, and the Payroll/Personnel System.

Programmed in ANS COBOL 74, the system is designed using a DMS 1100 data base. It can handle a variety of requirements with minimal program changes, by using a control file to customize applications.

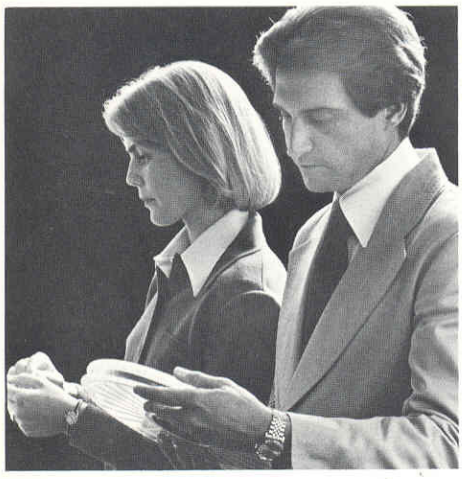
The General Ledger System maintains all necessary General Ledger information and does processing for multiple companies, divisions, and departments. It accepts automated input from the other modules of UNIFACS 1100. Budgeting functions, available from a self-contained subset of the system, may be reported with the Financial Reporter subsystem.

The Accounts Payable System automates the task of accounting for invoices and their related distribution, while still giving the user complete control over cash distribution.

The Accounts Receivable System provides maximum control and flexibility in processing cash payments and in controlling outstanding receivables. General Ledger transactions created by this system are accepted directly by the General Ledger System. A sales history is created for sales analysis functions.

The Payroll/Personnel Accounting System supports a variety of pay categories for employees and performs all federal and state tax calculations. It processes multiple non-tax deductions, manual overrides, handwritten checks and voided checks. Other features include labor distribution, General Ledger distribution, EEO reporting, and personnel history maintenance.

The system includes REPORTER, for customizing reports. A useful feature of UNIFACS 1100 is that each application can be used independently or in conjunction with other modules of the system.



SPERRY UNIVAC Financial Integrated Control System (SUFICS 1100)

SUFICS is not simply a modelling language, but a complete system for the design of planning applications which can interface with existing or planned systems.

Based on an English-like language, SUFICS was designed for ease of use as a tool to be used for planners and accountants rather than computer technicians. Simple, short commands combined with a series of prompts enable the new user to plan quickly and efficiently.

Some of the important features of SUFICS are:

- THE DATA STRUCTURE
  Data can be input in the appropriate format for the application—scaled, geometric growth, arithmetic growth, incremental growth or fitted to a straight line or exponential curve, etc....
- REPORTS
  The English-like report formatting language ensures that reports can be easily formatted to the standard required for the application. These reports can be stored on files for easy access.
- SENSITIVITY ANALYSIS
  "What If" changes can be made both to the rules and the data. Absolute and percentage changes may be made. The system can also work backwards from a specified target.
- DATA BASE INTERROGATION
  Models can be designed to interrogate time-series data bases or interface with existing database systems.
- INFLATION
  Different inflation rates can be selectively applied to input variables without changes being made to the rules of the model—thus optimistic and pessimistic inflation scenarios can be explored.
- SIMPLE AND HIERARCHICAL CONSOLIDATION
  Data can be consolidated in a simple manner (aggregation) or according to a company structure or hierarchy that can be expressed in English-like terms.
- RISK ANALYSIS
  Using the Monte Carlo method, probabilities can be attached to the most sensitive variables in the model to give probability distributions as output for the profitability criteria.
- Forecasting and Econometric Modelling
  Techniques such as regression analysis are available to forecast from historical data and to build regression models to predict microeconomic variables.

UNIDAS 1100

UNIDAS 1100 is an Information Storage and Retrieval System for Documents. Wherever fast access to large numbers of articles, reports, contracts, laws, general directives, or abstracts of publications are required UNIDAS 1100 provides an automated means of achieving this.

UNIDAS 1100 supports and controls document data bases using DMS 1100, and is a general purpose retrieval package, able to operate in an interactive and/or batch mode. It allows the user to:

- create and store data bases from individual documents
- expand existing data bases
- retrieve data from data bases according to specified criteria and by various methods, either through on-line dialogue with the system or in batch mode
- have retrieved data displayed at a terminal and/or printed on a high-speed printer.

UNIDAS 1100 provides a broad spectrum of search capabilities as well as flexible and varied output options for retrieved documents. Privacy is provided at many levels to ensure the integrity of both the system and the data to which it has access.

UNIDAS 1100 can be an invaluable asset to the work of people who are concerned with the storage, retrieval and manipulation of any type of document.

Automatically Programmed Tools (APT 1100)

APT 1100 is a complete contouring system for the computer-assisted programming of numerically controlled devices such as machine tools, flame cutters, drafting machines, and similar equipment. It is production oriented, written to take full advantage of numerically controlled techniques in engineering and manufacturing with the least expenditure of effort, time and money.

APT 1100 enhances most of the advantages usually found in numerical control: reduced load time, greater design freedom and flexibility, lower direct costs, greater accuracy, improved production forecasting, lower tooling costs, better engineering control of the manufacturing process, and simplified introduction of changes.

APT 1100 offers the following features:

- English-like command language
- Full multi-axis contouring machine tool capability, including 5 axis tool control
- Full geometric definitions
- Sculptured surfaces
- Pocketing, including BPOKET
- Regional milling, including BSURF and APTLFT/FMILL
- Basic pattern capability
- Cutter configuration options
- Repetitive programming
- Computational statements
- Postprocessor support routines
- Demand terminal printout options
- Ease of writing, editing, and running part programs by taking full advantage of the power of the executive system and its text editor and file manipulation capabilities.

Functional Mathematical Programming System (FMPS)

The FMPS is an advanced mathematical programming language. Its features include:

- A fast state-of-the-art linear optimizer (SPRINT), expressly designed to solve numerically difficult problems.
- Procedures commonly used to solve linear programming (LP) problems (CRASH, OPTIMIZE, INVERT). The current program will solve a maximum of 8192 rows depending on the amount of main storage available.
- A user oriented FORTRAN-like control language, through which the user can establish the sequence of operations to be performed, control the treatment of exception conditions and adjust tolerances if desired.
- Input procedures used to read matrix data from cards or tape in standard FMPS 8-character name format or in various share formats (ILONA, LP 90/94, UNIVAC 1108 UP, CDC CDM4).
- A state-of-the-art generalized matrix generator/report writer, GAMMA 3. The matrix generator minimizes the time required to create the matrix while providing maximum flexibility in creating the matrix. The generalized report writer displays the solution or input matrix in a user-defined format.
- Selective output by masking operation.
- Postoptimal ranging.
- Procedures for saving the basis, restoring the basis and obtaining error estimates.
- Procedures which can be called during the solution process to perform non-linear adjustments of the matrix coefficients. These procedures are especially useful in problems pertaining to gasoline blending.
- The ability to call user FORTRAN subroutines from FMPS or GAMMA.
- Bounded variable code.
- Recursive capabilities.
- Generalized upper bounded operating mode. This is a particularly valuable feature for transportation, weighted distribution, and multiplant distribution LP problems.
- The ability to call the GAMMA publisher from FMPS.
- Mixed integer programming.
- Parametric programming, including rim parametrics.
- Separable programming.

Interactive GAMMA Data Manipulation System (IGDS)

IGDS is a full screen editing product which facilitates the entry and manipulation of data for GAMMA, including GAMMA tables and lists. The GAMMA tables and lists in turn drive the FMPS matrix generator and report writer. This program product simplifies and speeds up the use of FMPS for mathematical programming applications.

OPTIMA 1100

Optima 1100 is a flexible project management system that will process data concerning networks of activities, analyzes time, cost, and resource utilization, performs resource allocation, and provides detailed and extensive reports.

The overall design is an integrated system comprising three functions: time analysis, resource allocation and cost control, which


SPERRY

can be employed in combination or individually. The parameterization and execution of the Optima 1100 processors are controlled via a command language. The system handles single and multi-network projects of as many as 4,095 activities per network, and up to 4,095 networks per run.

Data is input using fixed formats with variable identified sizes.

Optima 1100 provides activity, predecessor/successor activity, bar chart, activity cost, resource load, and cost reports. Extensive facilities permit the user to tailor his reports.

A network plotter program is available.

Integrated Civil Engineering System (1100 ICES)

ICES is made up of two components: the ICES System, and the various ICES subsystems. The relationships between these components is hierarchical in that the subsystems run under the control of the ICES System. Additional subsystems are developed using facilities provided in the ICES System.

Each of the subsystems is associated with a particular application area. For example the STRUDL subsystem deals with the structural analysis area, while the COGO subsystem deals with geometrics. The user communicates with a subsystem via a Problem-Oriented Language (POL) which allows him to state his problem to the subsystem in familiar and convenient terms.

The subsystems currently available are:

ICES STRUDL II—The *STR*uctural *D*esign *L*anguage

ICES TABLE I—An ICES File Storage Subsystem

ICES COGO—*CO*ordinate *GeO*metry

ICES ROADS—The *RO*adway *A*nalysis and *D*esign *S*ystem

ICES SEPOL—The *SE*ttlement *P*roblem-*O*riented *L*anguage

ICES LEASE I—*L*imiting *E*quilibrium *A*nalysis *S*lopes and *E*mbarkments

ICES TOPOLOGY I—The Structural *Topology* Subsystem

ICES URBAN COGO—The Urban Geometric Heuristics Subsystem

The SPERRY UNIVAC 1100 ICES is comprehensive—and growing. It's offered as a completely supported software system operating on SPERRY UNIVAC Series 1100 computers.

Graphics-oriented Interactive Finite-element Timesharing System (GIFTS 1100)

Gifts 1100 is a series of programs used primarily by mechanical engineers in the graphic design of structures. With GIFTS 1100, you design a visual image that you can alter and test with ease. The model can be viewed from any angle and allows enlargement of any portion of the model for closer scrutiny.

There is no need to program structure analysis routines, as GIFTS 1100 interfaces with many existing products such as ICES/STRUDL.

GIFTS 1100 reduces time spent on calculating and checking of coordinates. By issuing a few simple parameters, the system generates a mesh. The display of the resultant model may be edited, the analysis re-run and the model displayed again.

A "HELP" command is available if an explanation of a particular command is necessary.

Author System for Education and Training (ASET 1100)

ASET enables a curriculum development specialist in education and a training specialist in industry to develop "courseware" in either drill and practice or in tutorial Computer Aided Instruction (CAI) strategies.

The ASET system operates under the SPERRY UNIVAC 1100 Operating System on a minimum Series 1100 hardware configuration.

SPERRY UNIVAC Automated Documentation System (UNADS)

UNADS is a powerful and flexible document composition system which meets the rapidly growing technical documentation requirements of industry, government, and the military. UNADS provides a composition-oriented command language and a powerful macro definition facility. Together, these are used to establish a set of high level composition and control commands which are consistent with the documentation style guide of the user.

UNADS consists of a single processing program and two coordinated sets of model macro definitions. The first set is oriented to the production of general format documents using a SPERRY UNIVAC high speed printer. The second set is oriented to the production of general format documents as camera-ready copy on sophisticated devices such as phototypesetters.

Data entry of documents is performed using 1100 OS text editing facilities such as CTS.

UNADS:

- performs complete page makeup
- composes in multicolumn formats with vertical justification
- provides automatic page and paragraph numbering
- provides automatic headings
- provides six tables of contents and two level indexing
- provides right justification with optional algorithmic hyphenation
- reserves space either in-line or floated out-of-line for artwork
- provides a proof list on high speed printer
- provides a macro capability for user tailorability of data entry and final document standards
- provides for insertion of device handlers to adapt system to final output device of the user's choosing.

Spell 1100

The Spelling Checker is a processor designed to check for spelling and typographical errors in computer based documentation. It is used with input text for various Sperry Univac documentation processors as well as for accommodating plain text in a language other than English. Spell 1100 also offers the option to complement the provided dictionaries with private dictionaries and allows rebuilding of the provided dictionaries.

Spell 1100 provides an integral tool for producing a document of high integrity.

Requirements and Development Processor (RDP 1100)

RDP 1100 is a productivity application for software development. With a non-procedural language called RDL, a single source entry for all aspects of a software development project can be described, including:

- Requirements specification
- Design specifications
- Implementation details
- Testing analysis
- Management control.

The integrated RDP data base, built from the RDL commands, receives all data associated with the development project, and it is from this data base that appropriate reports are generated.

RDP 1100 provides concurrent services for all project personnel including: the analyst, as a vehicle for structure and design; the programmer for a coding guide; the project manager for status, control and performance analysis; and the eventual user of the system.

RDP 1100 has a built-in security system that allows only authorized users to access and/or update the information. This security system also allows RDP 1100 users to apply their own philosophy of data base access through user construction of the security directory.

To complement its functionality RDP 1100 interfaces with the Sperry Univac UNADS document processor. Complex documents may be built by tracing relationships and/or attributes. Objects, properties, and narrative text entries may then be selected to form such things as system requirements documents or unit design specifications for a programmer. UNADS also provides the capability to create pictorial representations or relationships such as HIPO charts, Nassi-shneiderman charts, flow charts, and even organizational charts from the information stored in the RDP data base.

Migration Software

SPERRY UNIVAC provides a variety of conversion software aids to simplify and automate application transition to Series 1100 systems. These software products perform file and data conversion, application program translation, and Job Control Language translation, as well as many support activities incidental to planning, implementing, and managing a conversion project. Using this software can save new users considerable time and effort in bringing up their applications on Series 1100 systems. Software products exist to handle applications written for most vendors' systems.

The key conversion software product is the UCAP/IMPACT System (UNIVAC Conversion Assistance Process/Integrated Management, Planning, Analysis, and Conversion Tool System). This powerful group of processors analyzes a new user's application, both programs and JCL, establishes an inventory of programs, jobs, and files to be converted, then effects the transition to the Series 1100 environment. Any statements which cannot be directly translated are highlighted in a special report and flagged in a translation listing.

UCAP/IMPACT offers planning and scheduling aid for the conversion project. The inventory is the project base line. When resource, skill, and time parameters are added, IMPACT processors develop a plan for the conversion, based on an IMPACT data base.

Other UCAP/IMPACT processors will track the project, identifying deviations from the schedule. This is an important tool for management, aiding project control.

UCAP/IMPACT SYSTEM COMPONENTS

Automatic Translation System (ATS)

ATS is a set of routines that serve as the basic interface to IMPACT. The user directs IMPACT in a conversational mode through ATS, selecting the IMPACT processors required and identifying the input and output files to be used. Tutorial sections of ATS identify the options available to the IMPACT user, and simplify the tasks of selecting the appropriate processor and options for the needed function.

Program Analysis Software (PAS)
PAS is a multiphase program analyzer which reads the source programs to be translated and analyzes them. PAS identifies program problem areas, processes record descriptions, calculates a program difficulty factor and builds a syntax correction file for CSC, the COBOL Syntax Converter.

COBOL Syntax Converter (CSC)
CSC translates other systems' COBOL (ANS '68 or 74) to SPERRY UNIVAC ASCII COBOL (ANS '74). Corrections and changes are based on PAS analysis and CSC source language analysis.

FORTRAN Syntax Converter (FSC)
FSC analyzes programs written in a variety of FORTRAN versions and translates them to SPERRY UNIVAC Series 1100, ASCII FORTRAN.

PL/I Conversion Aid (CPL)
CPL translates application programs written in PLI for non-Series 1100 systems to 1100 PL/I.

RPG Conversion Aid (RPGCON)
RPGCON translates application programs written in RPGII to RPG 1100.

Control Language Analyzer (CLA)
CLA reads job and task control streams. CLA analyzes these streams, identifying programs, program steps, files, and recording job/program/file relationships.

Control Language Generator (CLG)
CLG generates the 1100 Executive Control Language runstreams to replace the source JCL streams input to CLA. CLG also generates Compile and Map and Unit Test runstreams to assist the conversion analyst in compiling and testing the translated programs.

Directive Generator (DGEN)
DGEN uses the PAS record description information to create the File Converter output directives.

File Converter (FC)
FC is a parameter driven generalized data conversion processor. FC converts files from a variety of non-Series 1100 systems to 1100 file formats.

File Compare Processor (FCM)
FCM is a parameter driven processor which compares non-Series 1100 files to Series 1100 files.

Test File Generator (TFG)
TFG is a parameter driven processor which creates data for program testing.

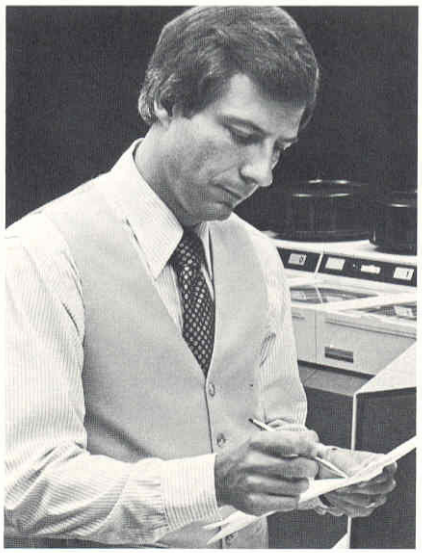
UCAP/IMPACT REPORT COMPONENTS

Register Printer (REGPRT)
REGPRT provides the conversion planner and conversion analyst with a variety of reports to support the planning and preparation phases. These reports or "Registers" are produced from the output of PAS, CLA, and other components.

- PROGRAM REGISTER details programs and files in the inventory.
- FILE REGISTER identifies file use, cross-referencing internal/external file names.
- JOB REGISTER lists the JCL streams in the inventory.
- PAS REGISTER displays program information which is available when only source programs have been input to SAS without accompanying JCL streams.

Planning and Scheduling System Data File Build (PSSLOD)
PSSLOD prepares the information derived by PAS, CLA, and RCP for loading on the IMPACT data base.

Conversion Project Definition Processor (CPD)
CPD is used by the conversion planner to define the project parameters. The data developed by the planner from the analysis of the report is input to CPD. CPD relates this data to the information already on the data base.



Scheduling Processor (SCHEDULER)
The Scheduler applies specific scheduling parameters to the data stored on the contractual data base and generates a conversion plan. The conversion planner may vary the parameters until a plan which best meets the project criteria is produced.

Dynamic Rescheduler (DRS)
DRS defines new project parameters, skills required, task sequences, schedule dates, and scheduling parameters. It allows the project manager to input changes modifying the original plan to determine the effect the altered parameters have on the plan.

Other Reports
In addition, other IMPACT components will produce reports on personnel time records, cross reference listings of files and programs, completed component summaries, exception reports, active work summaries, and other essential data.

Summary

The UCAP/IMPACT system is a tested system, proven during numerous conversions to Series 1100 systems. We have a reservoir of experience developed from that success. The new user of Series 1100 can be confident that transition to the new system is supported by a comprehensive set of programs and methods covering every need. UCAP/IMPACT incorporates a high level of automation, speeding up the conversion process and increasing productivity. The system is custom tailored to the requirements of management.

SPERRY UNIVAC

The computer people who listen.

U7077

8111 Printed in U.S.A.