NCR CONTROL DATA

MEMO

ADVANCED SYSTEMS LABORATORY

DATE: September 6, 1974

TO: R. S. Banks LOCATION:

FROM: P. R. Fundenburg LOCATION: EXT:

SUBJECT: IPLOS R&G

Persuant to our agreement, I have completed the task of reformatting the IPLOS Requirements and Goals, at least as far as is possible prior to my absence from the office for the next two weeks. The package being delivered with this memo includes the following:

1. A replacement page iv for the Requirements and Goals Table of Contents.
2. A complete new section, 6.1 - Operating System, for the Requirements and Goals Document (including a sub-table of contents for the section).
3. A memo, P. R. Fundenburg to R. S. Banks, on the status of the Action Items generated during the O.S. R&G meeting of 8/7 and 8/8 1974.
4. A memo, G. M. Beaugonin to D. L. Slais, providing a list of product set members requiring O.S. support.
5. A memo, W. Rehling to P. R. Fundenburg, providing response to Action Items 6 and 7.

Additional material necessary for a full understanding of the above items includes:

6. O.S. Requirements Study Group Report - Wagner, et. al. - 3/19/74. (This should already be in your possession; if not, it may be found in the Design Data Base.)
7. Minutes of IPLOS R&G Review Meeting - Fundenburg - 8/13/74. (The comment under 6, above, also applies to this document.)
8. Review and Harmonization of Parents Comments on Task Force Document - Fundenburg (same comment as for 6 and 7, above.)

It was originally intended that proposed requirements, extracted from other sources, be included in item 2 along with the requirements already agreed upon by both parents. These proposed requirements were to be flagged in the document with an asterisk. However, due to time limitations, only one such actually appears in the document. It will therefore be necessary to present these additional proposed requirements (they are numerous) at a later date. In any

E.7.501 B.23

NCR CONTROL DATA

MEMO

ADVANCED SYSTEMS LABORATORY

DATE: September 10, 1974

TO: J. A. Kershaw LOCATION:

FROM: R. S. Banks LOCATION: EXT:

SUBJECT: IPL Operating System Requirements and Goals — 9/6/74

Enclosed herewith for transmittal to the ASL Planning Committee is an update of the IPL Operating System R&G's. This report constitutes a major revision of the original "mini" report of March 19, 1974, resulting from:

1. Review by the parents,
2. A seminar conducted by ASL on August 8-9, 1974, and
3. Addition of new R&G's proposed by ASL.

This report is close to the point where ASL can ask the Planning Committee for formal approval; however, as noted in the attached memos, there are several unresolved issues which must be addressed before that time. After cleanup of these items, I recommend that ASL forward the final document to the parents (via Messrs. Mitter and McHale, respectively) for internal review and recommendation of approval/disapproval to the respective Lead Representatives. In other words, further review will be at the "working level" and the Planning Committee will not be involved until asked for formal action on the final document.

Respectfully submitted,

R. S. Banks

RSB/ds

cc: H. J. Squires w/o attachment
P. R. Fundenburg w/o attachment
R. E. Wagner w/o attachment
R. W. Clough
D. L. Slais

NCR CONTROL DATA

MEMO

ADVANCED SYSTEMS LABORATORY

cc: T. Amochaev
G. P. Hulshult
M. J. McHale
T. S. Mitter
H. J. Squires
R. E. Wagner

DATE: September 6, 1974

TO: R. S. Banks LOCATION: PGAASL

FROM: P. R. Fundenburg LOCATION: PGAASL EXT: 830-6484

SUBJECT: Status Report - IPLOS R&G Action Items

Thirteen action items were generated during the IPLOS R&G meeting at ASL/C on 8/7 and 8/8/74. It was hoped to obtain resolution of these items by 8/30/74; however, this has not proved possible, and the purpose of this memo is to detail the progress made so far and to flag the items still needing attention.

AI No. 1: R. E. Wagner - Amplify R16 with a definition of dynamic linking.

Mr. Wagner has provided the following definition in response to this item:

"Dynamic Linking is a service rendered a user that allows the execution of jobs prior to the completion of a load or binding action. The user appearance of such a service is that a job requiring infrequent use of certain code and/or data segments can elect to execute the job in a partial bound (loaded) state, thereby saving the additional space required for a fully bound state. Also, program development and debugging can proceed prior to implementation of all necessary modules. Dynamic linking to data segments requires hardware assists."

AI No. 2: NCR/CDC - Provide positions on dynamic data binding.

NCR has replied, stating that this is a requirement, with a priority of 3. CDC has not yet replied.

AI No. 3: NCR/CDC - Provide positions on fixed virtual addresses as per NVS and STAR subroutines.

NCR has informed us that a final position on this subject cannot be provided until the larger issue of NVS/IPL compatibility is finalized. If an NVS virtual machine is defined, fixed virtual addresses will not be needed. If, on the other hand, it is required to run NCS/NVS jobs in emulation mode under IPLOS, fixed virtual addresses will be required.

CDC has indicated that they have no requirement in this area, since STAR/IPL compatibility is not required.

case, the unresolved Action Items would create the need for a later revision, even if all else were complete.

If the requirements in item 5, above, can be reviewed by the Planning Committee along with the formal R&G document, it would help speed up the process.

P. R. Fundenburg, Sr. Consultant
Architectural Control

PRF:jmk

cc: H. J. Squires
R. E. Wagner

AI No. 4: NCR/CDC - Expound on performance and storage improvement multiplier by device class: Secondary storage, RMS, magnetic tape, U/R, communication lines; and further by: access time, transfer rate and capacity, through 1985.

NCR has indicated that an improvement multiplier of 10 is desired for RMS, magnetic tape, printers and communication lines. No breakdown was given by access time, transfer rate and capacity; 10X is assumed for each. NCR has no secondary storage facilities at the present and therefore has no feel for a required improvement multiplier. U/R equipment other than printers are not expected by NCR to undergo any substantial improvement in speed or capacity. NCR is using as a baseline in each case presently deliverable equipment, not necessarily state-of-the-art. For example, the baseline for disc is 806KB transfer rate, 100MB capacity and 38.3 ms. access time.

AI No. 5: ASL - Provide a list of languages to be supported initially and at the second release. This list is to be inserted as a requirement.

The attached memo from G. M. Beaugonin to D. L. Slais provides the list of languages which CDC requires that IPL support. It is ASL's understanding that NCR is in agreement with this list. Time-phasing (initial vs. second release) will be determined later.

AI No. 6: ASL - Obtain requirements regarding DBMS Recovery Services, DBMS Security and Data Base Reorganization Tools from the DBMS design team.

ASL has requested these requirements from the DBMS design team; the attached memo to P. R. Fundenburg from W. Rehling indicates that these requirements cannot be provided until a firm decision is taken as to exact methods that will be employed by DBMS in these areas. These requirements will be included when received.

AI No. 7: ASL - Obtain requirements regarding Interlock Support from the various Product Set design teams.

The attached memo to P. R. Fundenburg from W. Rehling outlines these requirements. These will be rewritten in R&G format as soon as possible.

AI No. 8: NCR - Reflect the requirement for IPL internal compatibility throughout the R&G where applicable.

NCR has indicated a need for further discussion with ASL regarding the exact intent of this item. Resolution of this AI is pending such discussion.

AI No. 9: NCR - Define exact minimum configuration workloads.

NCR has provided the following requirements:

A. Dedicated batch processing - three jobs (BDP oriented); e.g. one COBOL compilation and two production jobs.

B. Concurrent online/batch processing - one online and two batch jobs. The online job must support 35 terminals, with a transaction throughput rate of six to eight per second.

C. Dedicated online processing - one online job supporting 100 terminals, with a transaction throughput rate of ten to twelve per second.

The terminals to be used in online jobs are expected to be of the NCR 270 or 280 type, possibly mixed with 260's. (Transaction processing, not time sharing.)

AI No. 10: NCR - Confirm that R42.1 satisfies a real need.

NCR is still awaiting confirmation from R. Hunter. In the interim, NCR's basic position is that the operating system should be able to flaw out the smallest unit of memory that is likely to fail at one time, from a hardware point of view.

AI No. 11: NCR/CDC - Provide figures to fill in the blanks in section 3.1.6.A, RAS parameters.

No figures have yet been provided by NCR or CDC. This item has also been given to the RAS mini-task force for consideration.

AI No. 12: NCR - Investigate security vs. cost tradeoffs for S0 and S1.

NCR reports that this item is still under investigation. This item will also be given to the security mini-task force for consideration.

AI No. 13: NCR/CDC - Clarify positions in the area of optionality of O.S. security measures (R93).

NCR has provided the following position: It must be a site selectable option to enable those O.S. security measures that degrade system speed by more than 5%.

CDC has not yet replied. This item will also be given to the security mini-task force for consideration.

P. R. Fundenburg

Paul R. Fundenburg

PRF:kg

ATTACHMENTS

c) Block interlock on block requests:

DBMS is the only product with an identified need for block locking at this time. This could be done with special lock/unlock requests, or tacked onto position and I/O requests. Wait/reject options are desirable, along with automatic detection of cyclic rejects, multi-job deadlocks, or some time-out algorithm for preventing endless lockout when a user loops while locks are in effect. A separate form of lockout might be required by DBMS if the DBTG keep/free approach becomes standardized. With a keep, the OS does not lock out other users, but records block use by others. This approach avoids deadlocks but puts a logic burden on DBMS and the user, so will not be requested of the OS unless standardized. OS lockout mechanisms should try to avoid precluding future addition of this feature.

d) Block interlock on record requests:

DBMS requires a way to lock the block containing a record when accessing it via record I/O. The preferred implimentation would be to have exact parallel requests or parameters to those implimented for block I/O. The alternative would be a separate request for DBMS to pass a record access key or number to the OS and get returned the block number the OS has mapped it into. Then block reserves could be constructed by DBMS.

e) Record interlock:

No requirements have been identified for locking parts of blocks by the OS.

f) Other forms of interlock:

No OS requirements are expected for other forms of I/O locking. DBMS will control any locking on sets, record types, elements, etc., should that ever be needed.

2) AI No. 6 - DBMS requirements regarding recovery, security, and reorganization tools.

There is no firm decision as to exactly what methods will be employed in these areas, and the role required of the OS is unclear at this time.

Wood Rehling

cc: N. M. Kendall

NCR

MEMO

ADVANCED SYSTEMS LABORATORY

DATE: August 28, 1974

TO: P. R. Fundenburg LOCATION: PGAASL

FROM: W. Rehling LOCATION: ESCASL EXT: 65

SUBJECT: IPLOS R&G Action Items 6 & 7

1) AI No. 7 - Interlock Support for the Product Set

a) File interlock at open time:

Protection for exclusive use of a file, via wait/reject, is required at option by all products. Sorts and recovery do not strictly need more protection mechanisms than this, but will most likely allow those used by COBOL.

Protection for protected use is required by COBOL, RPG, and DBMS, whereby a file may not be shared with another job requiring open for output. RPG does not strictly require more, but will most likely follow COBOL.

Permission for unprotected open of input files is required by COBOL and DBMS, except where it conflicts with exclusive opens. No explicit block/record locking should be required when a user requests this permission, although the DBMS user will have the locking as described below.

Permission for unprotected open of output files (full sharing at file level) is required for DBMS, with DBMS making explicit block/record lock/unlock requests as described below. COBOL does not currently require this level of sharing, although it is a desirable feature for some applications and several proposals are being discussed. Unless there is a standard for such processing, COBOL will not consider extensive decision making or explicit requests by the user. (COBOL might consider a simple implicit rule (such as, the last record/block pointed to or read is locked and a write removes the lock) especially if the OS enforced it as a default to explicit locking and if it could solve a clear class of problems, such as several jobs adding records to the end of a shared file.)

b) File interlock upon request (file reserve):

DBMS will utilize file reserves only rarely, but would consider passing user file reserve requests through to the OS. COBOL and the other products do not anticipate a need for this, although sorts could use this to protect its processes from externally opened unprotected files.

NCR/CDC INTEGRATED PRODUCT LINE

OPERATING SYSTEM

REQUIREMENTS AND GOALS

September 6, 1974

MEMO

DATE: 26 June 1974

TO: D. L. Slais LOCATION: PGAASL

FROM: G. M. Beaugonin LOCATION: HQS11B EXT:

SUBJECT: Action Item #102 - Product Set Support

REFERENCE: My memo of 24 June 1974 - same subject

CDC requires the following product support for IPL:

1. COBOL - ANSI 73 + certain extensions (ATG, DBTG) Virtual Storage
2. SORT/MERGE - with interfaces to COBOL and FORTRAN
3. FORTRAN - ANSI 73 FORTRAN + Extensions
4. BASIC
5. APL
6. RPG
7. PL1 - CDC has not yet determined the viability of a PL1 product, however, CDC wishes the IPL OS to incorporate those features necessary to support a viable PL1 compiler. We assume from this directive that several people will have to be assigned to PL1 compiler development in order to specify compiler requirements to the OS Development Group.
8. ALGOL - currently CDC has no requirements for ALGOL.
9. SIMSCRIPT - products in this area will be developed separately by CDC.

- G. M. Beaugonin -

cc: R. W. Duncan HQW12I
L. E. Jodsaas ARH210
R. E. Nienberg ARH241
M. J. McHale HQS11B

/sf

TABLE OF CONTENTS - 6.1 OPERATING SYSTEM

TABLE OF CONTENTS - 6.1 OPERATING SYSTEM

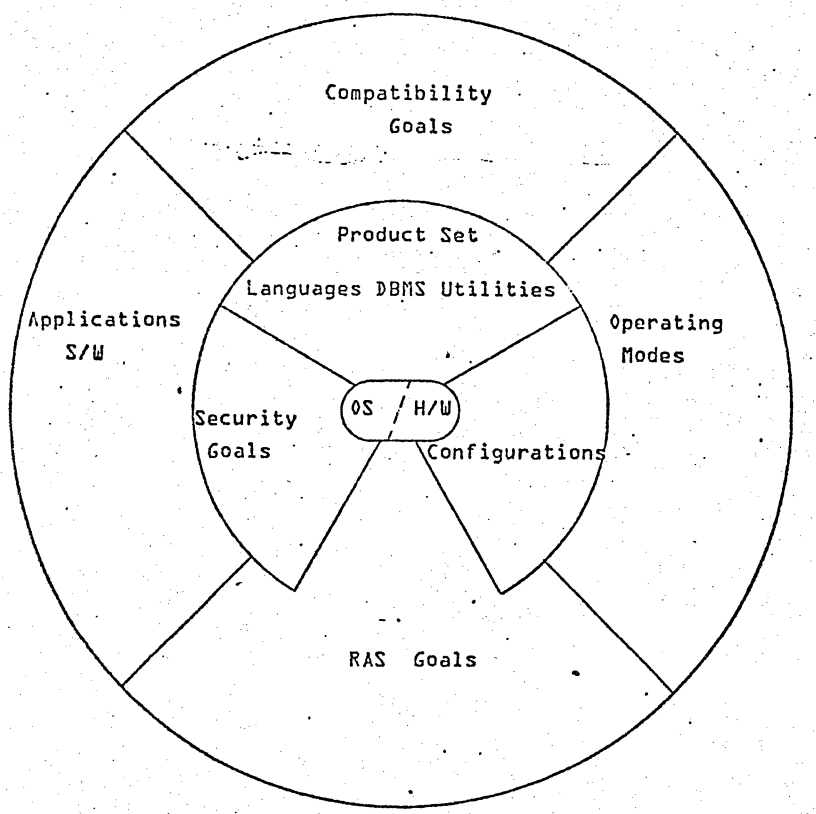

Figure 1

Software Requirements Model

GENERAL

OS requirements papers have been generated many times in the past with various degrees of success. A major reason for failure of previous efforts is the lack of a generally accepted definition of what is and what is not an operating system. An outline model has been defined to be used as a vehicle for deriving OS requirements. The position taken herein is that OS requirements are defined through two major channels:

1. the OS requirements of product set members
2. general statements of requirements and goals of all system software (i.e., OS, Product Set, Application S/W.)

OS Requirements model

For purposes of specifying OS requirements a distinction must be made between those statements made directly to the OS and those that are stated in terms of product set and applications software requirements. This situation can be graphically portrayed according to the traditional "Onion Skin" picture of a Computer System.

Figure 1 represents the fact that OS requirements should be stated largely in terms of product set interface demands and general software goals and objectives. The outline model reflects this view.

GENERIC OS RESPONSIBILITIES (CONT.)

7. Support hardware and software diagnostics and instrumentation.
8. Error detection, reporting and recovery.
9. Execution support services.

Requirements for all the above headings except Nos. 7 and 8 will be found in the immediately following pages. Requirements for Nos. 7 and 8 will be found under the broader topics of RAS (section 6.1.7) and Performance (section 6.1.9), as appropriate.

GENERIC OS RESPONSIBILITIES

Every operating system regardless of size, speed or adequacy will provide a certain set of services with varying degrees of capability. Another way of stating the same position is that if left to their own devices and theories every OS designer and developer will supply a certain minimum set of capabilities with a level of usefulness that is determined by the interest of the designer/developer. These are considered generic OS responsibilities. It is the responsibility first of the specifier of requirements to modulate these generic responsibilities to meet the specific needs of a product line endeavor. It is then the responsibility of developers and their management to implement the correct variations of these generic functions.

The generic OS responsibilities are:

1. Effect physical and logical transfers of information within a system configuration.
2. Communicate with the system operator(s).
3. System work scheduling and the control of logical and physical system resources.
4. Resource usage accounting and general status reporting.
5. Protect system integrity and the isolation of various work units within the system.

Operator Communication

1. Operator communication must be done through standard I/O interfaces.
2. Operator communication must be capable of accomodating multiple and/or remote operator configurations/consoles.
3. The system must allow specialized function consoles, e.g. remote I/O writers to control magnetic tape delivery.

Data Transfer

1. The system must be capable of effecting all data transfers at rates equal to the maximum for the hardware involved.
2. Data streaming capability is required of the system. Streaming is defined as the ability to process two or more consecutive I/O requests for the same device without degrading the device transfer rate. A minimum of four simultaneous user data streams are required to be supported in addition to any system streaming needs.

Accounting and Status Reporting

1. All significant state changes of a job must be able to be indicated to a user. (Including normal job or step termination).
2. Full and complete accounting and logging services must be provided. Level of detail must be selectable as a site option. Billing services are not required of the operating system.
3. The system must provide environment description services, both static and dynamic, sufficient to allow jobs to self-optimize themselves at execution time.
4. Options must be provided that allow site managers and end users to take maximum advantage of available resources for both individual and multi-programmed runs.

System Scheduling and Resource Control

1. With the exception of requirements for physical action, the system must be capable of executing in an unattended manner.
2. Operator override must be provided over all system made operational decisions.
3. The system must provide time initiated, event initiated, program initiated and conditional scheduling.
4. The system must provide exclusive or shared allocation of devices and/or files.

System Load, Restart and Termination

1. The system must provide orderly load and terminate operations under both normal and emergency conditions without loss of job, system, or data integrity. This must be accomplished with minimal operator intervention.

System and Job Integrity

1. The system must be impervious to all errors in operation.
2. Job integrity must be maintained independent of other job or system errors.

NCR/CDC PRIVATE

Execution Support Services (Cont.)

9. One library format must be able to meet all system requirements. (i.e., load modules, object modules, source modules, macros,...)
10. Various levels of object code binding must be provided. (static and dynamic).

Dynamic Linking is a service rendered a user that allows the execution of jobs prior to the completion of a load or binding action. The user appearance of such a service is that a job requiring infrequent use of certain code and/or data segments can elect to execute the job in a partial bound (loaded) state, thereby saving the additional space required for a fully bound state. Also, program development and debugging can proceed prior to implementation of all necessary modules. Dynamic linking to data segments requires hardware assists.

11. The operating system must possess a file management capability that is functionally compatible with current file systems.
12. The O.S. file management facility must provide for a record level interface.
13. The operating system must insulate the user from any peculiarities of a particular device, (i.e., provide device independence) at all levels of I/O.

Execution Support Services

1. Time-of-day service must be provided.
2. Interval timing must be provided.
3. Process timing must be provided for system and user functions.
* 4. Wake-up service must be provided (e.g., to allow activation, reactivation or switching of tasks at a specified time or after a specified time-lapse).
5. A well defined consistent and common interface between the operating system and various subsystems must be provided. It must provide all the capability required for language compilers, system utilities, telecommunications, application packages and user programs so as to eliminate "special interfaces" to meet special requirements of some program.
6. Support must be provided for multiple user and application libraries. These libraries must be individually protected and must be modifiable independently of the system or other libraries.
7. Access to each library must be optionally regulated and full accountability of any library usage provided.
8. The order of library selection and scanning must be a site manager and end user option.

Execution Support Services (Cont.)

21. The O.S. must be designed to enable users to secure and protect their data both from other users and from the site management and operators.
22. Protection of user data from the operating system itself is highly desirable.
23. Users must be able to be provided with a bit copy of data from any external peripheral device or communication path as an option to the normal I/O path. (for device dumping and temporary support of new devices.)
24. The file management system should be considered as a subsystem utilizing basic functions provided by the basic operating system.

Execution Support Services (Cont.)

14. Device class dependencies will be supported (i.e., CRT vs mass storage vs tape) with minimum perturbation to the end user.
15. The O.S. must support all contemporary device types and capacities at a competitive level of performance and security.
16. The O.S. file management facility must be sufficiently modularized and encapsulated within standardized system interfaces to allow total or partial functional replacement to meet the needs of specialized application problems. The OS will not provide special interfaces peculiar to the usage of any product set member.
17. The O.S. must be designed with the capability of supporting new devices having increased performance and storage characteristics as set forth in section 6.1.9.1, Performance Goals.
18. The O.S. file management facility must be designed to support data bases of a capacity ranging to 10^{12} bytes of information.
19. The O.S. file management facility must be designed to allow levels of security down to the element level with various intermediate levels which are selectable as a site manager or end user option at a data base and/or file level.
20. The O.S. must be able to meet NCR/CDC standards as they become specified.

Languages

General

1. Support for the following languages must be provided by the O.S.:
   - Software Writers Language (SWL)
   - COBOL - ANS 73 plus certain extensions (ATG, DBTG, virtual storage, Sort/Merge interface)
   - P1/I - It has not yet been determined that PL/I will be an IPL product, but it is required that the IPLOS incorporate the features necessary to support this language.
   - BASIC
   - APL
   - FORTRAN - ANS 73 plus certain extensions
   - RPG
   - SIMSCRIPT

2. There is no requirement for support of ALGOL.

Specific

3. SWL - requirements to be supplied by the SWL design team.
4. COBOL - requirements to be supplied by the COBOL design team.
5. PL/I - requirements to be supplied by the PL/I design team.
6. BASIC - requirements to be supplied by the BASIC design team.

PRODUCT SET REQUIREMENTS

For purposes of this document a list of anticipated product set members has been identified according to three groups; languages, utilities, DBMS. Within each group the order of inclusion is specifically a statement as to the impact on OS design associated with that member regardless of the requirements for that member. (E.g., COBOL has the most pervasive effect on OS design of the various standard languages while FORTRAN has little additional design impact).

General Requirements

1. The O.S. will be designed such that any product set member can be designed for interactive and/or batch use without restriction or reservation due to the O.S. requirements.
2. Object codes produced by compilers must be in a form that allows loading and execution of programs consisting of modules coded in mixed languages. It is not a requirement to mask language dependencies from such usage.
3. All system utilities must be "callable" from internal program sequences (e.g., Sort/Merge callable from COBOL).
4. The O.S. must provide intercommunication capability for product set members and user jobs such that information associated with one job may be made available to the O.S. or to other jobs.

Utilities

General

1. Utilities must be supported within the system framework that meet all needs of systems operations., there must not be any stand alone or special load operation type utilities. {Possible exception - major crash recovery and dumps.}

Specific

Sort/Merge

2. A sort/merge facility must be supported that is performance competitive at all levels of configuration.

System Generation

3. The facilities to generate, optimize and maintain the operating system and all other software must be provided. This process must allow the site to replace and/or modify any and all of the various components of the system with minimal perturbation to normal running procedures. The integrity of user modified operating systems is the responsibility of the user making the modifications and not of the remaining standard OS components.

Media Transfer and Formatting

4. Requirements to be supplied by the Utilities design team.

Languages {Cont'd}

Specific {Cont'd}

7. APL - requirements to be supplied by the APL design team.
8. FORTRAN - requirements to be supplied by the FORTRAN design team.
9. RPG - requirements to be supplied by the RPG design team.
10. SIMSCRIPT - requirements to be supplied by the SIMSCRIPT design team.

COMPATIBILITY/COEXISTENCE/MIGRATION

This section has been broken down into the following topics:

1. Compatibility Target Identification - to define with which systems IPL compatibility must be achieved.
2. Coexistence Time Relationships - to define the degree of coexistence, i.e., instant replacement, short-term conversion or long-term concurrent operation.
3. Degree of Compatibility - to define how well compatibility should work in each case.
4. Level of Compatibility - to define level at which compatibility will occur, i.e., source code, object code and/or data formats.
5. Configuration Components - to identify any system components which present special compatibility problems.
6. Operational Aspects - to define the degree of compatibility required from the operational viewpoint.

Data Base Management System (DBMS)

DBMS requirements on the O.S. to be supplied by the DBMS design team.

The following areas of impact have been identified, but specific requirements are not yet known:

1. Recovery Services
2. Security
3. Data Base Reorganization Tools

Compatibility Target Identification

CDC (Cont'd)

5. Compatibility with CDC 3000L systems will be achieved through emulation of the MASTER operating system. The IPLOS must support such emulation and its corresponding data formats.
6. The O.S. must support source code compatibility for CYBER 70/170 FORTRAN, COBOL and DBMS.
7. The O.S. must support conversion aids for the CDC 3000L.
8. There is no requirement for support of compatibility with the following: NOS, ALGOL, or COMPASS; CDC 7600, CYBER 76, CDC 1700, STAR, CDC 8600.

IBM

9. The IPLOS must support source code compatibility with IBM 370 FORTRAN, COBOL and a subset of IBM data base code. The reference point for compatible versions is OS/VS Release 2.

Other Manufacturers

10. There is no requirement for compatibility with any other systems.

Compatibility Target Identification

IPL

1. The O.S. must support compatibility between all models in the IPL line, subject to configuration constraints.
2. Full upward compatibility is a must, to allow for a convenient user growth path.
3. Downward compatibility is desired, to allow extracting of network developed applications to smaller dedicated systems, and to permit degradeability of single systems; it is realized, however, that this goal is attainable only within limits imposed by configuration constraints and that these limits will vary for each individual case.

NCR

Requirements for compatibility with NCR systems will be supplied subsequently.

CDC

4. Compatibility with CDC 6000 systems will be achieved through emulation of CPU code. The O.S. must support such emulation and its corresponding data formats.

Degree of Compatibility

IPL

1. Full IPL compatibility (source, object and data) is required upwards and across the product line. Downwards compatibility will be limited by configuration constraints (peripherals and memory.)

NCR

2. Equivalent performance is required on an IPL configuration equivalent to the original NCR configuration. The IPLOS must support replaceability without user visible impact on software or user jobs.

CDC

3. For CDC systems which are replaceable through emulation, the IPLOS should support replaceability without user-visible impact on software or user jobs.
4. For CDC systems not emulated by IPL, no source conversion should be required for supported languages. Recompilation will be required and degraded I/O performance is acceptable.

Coexistence Time Relationships

IPL

1. The O.S. must support instant replacement of one IPL model with another, without user-visible impact on software or user jobs.

NCR

2. NCR systems with which IPL is compatible should be instantly replaceable by an equivalent IPL configuration.

CDC

3. CDC systems with which IPL is compatible through emulation should be instantly replaceable by an equivalent IPL configuration.
4. CDC systems not emulated by IPL must be capable of undergoing a short-term conversion. The O.S. should support conversion aids (utilities).

IBM

5. Long-term conversion and coexistence are required. Compatibility is required between IBM and IPL magnetic tape formats to support coexistence.

Level of Compatibility

Source Level

1. All IPL external source code must be transportable throughout the entire range of the Integrated Product Line without requiring any user conversion.
2. All languages jointly designated by NCR/CDC will be supportable by the IPLOS.

Object Level

3. Object support will be provided for NCR systems designated for IPL emulation.
4. Object support will be provided for CDC systems designated for emulation.
5. No object support will be provided for IBM or other manufacturer's systems.

Data Formats

6. All IPL external data must be transportable throughout the entire range of the Integrated Product Line without requiring any user conversion.
7. B1 & B2 generated external data must be processible throughout IPL without user conversion.

Degree of Compatibility (Cont'd)

IBM

8. No source conversion should be required for supported languages. Recompilation will be required and degraded I/O performance is acceptable.

Configuration Components

1. Programs supporting critically time dependent devices (e.g., MICR) will normally require user conversion.

Level of Compatibility

Data Formats (Cont'd)

8. CDC and IBM external data files whose content, type and format are statically defined will be processible on IPL without requiring additional user conversion. (Statically defined means via fixed definition as opposed to being defined at execution time.)

CONFIGURATIONS

Memory

1. The OS design must be capable of efficiently supporting main memory sizes from 256K bytes up to 10^8 bytes and larger through time. The basic design must not require user interface changes to adapt to these larger memories.
2. Required workloads for minimum memory size will be specified at a later date.
3. The OS must organize memory in such a manner as to maximize usability, availability, performance and security, in that order (see 2.2.1, Joint Objective 23).
4. Memory must be degradable in steps no greater in size than 25% of normal real memory capacity.

Processors

5. The OS must allow all processors to access "Common Memory".
6. All processors must access memory through the virtual addressing mechanism.
7. The OS must accommodate multiple types of processors in one configuration. At least two types (with and without I/O) are presently specified; future specification of "virtual machines" may increase the number of processor types to be supported.

Operational Aspects

1. No operational compatibility requirements exist. In the case of NCR B1 and B2 upgrades operational changes must be restricted to additional parameters outside existing job streams rather than alterations to the job streams.

OPERATING MODES

1. The IPLOS must be tunable, via system parameters and generation options, in order to maximize performance in any one or a combination of the following areas, stated in order of relative importance:
   a) local batch
   b) remote batch
   c) time sharing
   d) transaction processing
   e) time critical processing

2. The level of performance achievable in any particular area is not required to be at the level of specially developed dedicated application systems but the standard IPL OS must be able to supply the majority of code that would make up such dedicated special systems.

Environmental Modes

Local Batch

Requirements to be supplied.

Remote Batch

Requirements to be supplied.

Time Sharing

Requirements to be supplied.

Transaction Processing

Requirements to be supplied.

Time Critical Processing

Requirements to be supplied.

CONFIGURATIONS (Cont'd)

8. Multiple processors of each type in any hardware configurable mix must be supported.

Peripherals

9. Peripheral requirements on the OS remain to be specified.

Data Formats

10. The O.S. must remain insensitive to hardware data formats.

OPERATING MODES

Internal Modes (Cont'd)

Multiple Virtual Machines

8. The IPLOS must support multiple virtual machines (e.g., for direct execution of NCR CENTURY B1 and B2 programs in an IPL environment.)

Concurrency of Environmental Modes

9. The IPLOS must support any combination (in any proportion) of mixed environmental modes.

Multiprogramming

10. The IPLOS must support multiprogramming.

OPERATING MODES

Environmental Modes (Cont'd)

Network Coupled

3. The basic OS design must be such that network processing and/or network coupling subsystems can be optionally included without alteration to the OS. These network processing subsystems must allow the system configuration to function as a central computing source, as a front end message switcher, as a remote user terminal, as a data base controller or other similar usages.

Dedicated Application Systems

4. The IPLOS will not specifically address this area. OS design should support this use in special development cases.

Internal Modes

Centralized vs. Distributed Operation

5. Support for distributed operation will only be made as explicit user requests, not as a normal, transparent OS function.

Centralized vs. Distributed Data Bases

6. Support for distributed data bases will only be made as explicit user requests, not as a normal transparent OS function.

Multiprocessing

7. Support for multi-processing (like and unlike processors) must be provided.

RELIABILITY, AVAILABILITY AND SERVICEABILITY (RAS) (Cont'd)

RAS Parameters (Cont'd)

Mean Time Between Interruptions (no user visible destruction)

Processing

Requirements to be supplied.

Data Base

Requirements to be supplied.

Mean Time to Recover After Interruption

Processing

Requirements to be supplied.

Data Base

Requirements to be supplied.

Mean Time to Repair Error Causing Interruption

Requirements to be supplied.

Diagnostics

1. A complete history of all system errors, both hardware and software, must be recorded for subsequent analysis by maintenance and enhancement personnel, with enforced execution by the OS. In-Line diagnostics will be provided, and On-Line and Off-Line diagnostics will be supported, by the OS.

RELIABILITY, AVAILABILITY AND SERVICEABILITY (RAS)

RAS Parameters

Mean Time Between Failures (some user visible destruction)

Processing

Requirements to be supplied.

Data Base

Automatic Reconstruction

Requirements to be supplied.

Manual Reconstruction

Requirements to be supplied.

Mean Time to Recover After Failure

Processing

Requirements to be supplied.

Data Base

Automatic Reconstruction

Requirements to be supplied.

Manual Reconstruction

Requirements to be supplied.

Mean Time to Repair Error Causing Failure

Requirements to be supplied.

RELIABILITY, AVAILABILITY AND SERVICEABILITY (RAS) (Cont'd)

Error Detection, Reporting and Recovery (Cont'd)

Job/User

6. Checkpoint/restart facilities must be provided for both system use and individual job use.
7. The system must be impervious to user errors.

Operational

8. The system must be impervious to operator errors.
9. The console operator must not be a necessary component in error reporting or action procedures except as physical action requires.

Growth Techniques

Hardware

10. The OS design must be capable of efficiently supporting main memory sizes from 256K up to 10^8 bytes and larger through time. The basic design must not require user interface changes to adapt to these larger memories.
11. The addition of peripherals of increased performance or of new characteristics must cause a minimum perturbation to a running system configuration. (i.e., no down time). This requirement does not imply the ability to plug and unplug peripherals while the system is running.

RELIABILITY, AVAILABILITY AND SERVICEABILITY (RAS) (Cont'd)

Error Detection, Reporting and Recovery

2. All deviations in the normal or expected processing of hardware or software functions must be detectable and reportable at the option of the site manager and/or end user. Recovery sequences must be provided by the system for all types of system interruption.
3. Error reporting conventions must be direct and explicit and should optionally not require the use of additional explanatory materials.
4. A complete history of all system errors must be recorded for subsequent analysis by maintenance and enhancement personnel.

Hardware

Requirements to be supplied.

Software

5. System and job recovery must be provided for all levels and types of system failure. Multiple levels of system recovery must be provided which recover from various levels of system loss.

System

Requirements to be supplied.

RELIABILITY, AVAILABILITY AND SERVICEABILITY (RAS) Cont'd

Testing and Debugging

Hardware

Requirements to be supplied.

Software

16. Full symbolic testing and debugging services, both preplanned and post mortem must be supplied.

17. Provisions must be included that allow selective installation and testing of upgraded versions of compilers and application packages without affecting normal operations of the standard version. Access to the test version must be selectable by site managers and/or end users. The OS must be able to supervise another IPLOS (user modified version or different releases) for such purposes as upgrading and testing.

Operational

Requirements to be supplied.

RELIABILITY, AVAILABILITY AND SERVICEABILITY (RAS) (Cont'd)

Growth Techniques (Cont'd)

Software

12. Site and user developed subsystems and programs must be installable and maintainable in the OS in a running environment. User-developed operating systems and new releases of an IPL OS must be installable and usable under the regular IPLOS running environment.

Support/Maintenance

13. Failing system configuration components, hardware and software, must be individually able to be removed from the operational configuration for checkout and repair without downing the operational system.

Hardware

Requirements to be supplied.

Software

14. The facilities to generate, optimize and maintain the operating system and all other software must be provided. This process must allow the site to replace and/or modify any and all of the various components of the system with minimal perturbation to normal running procedures.

15. The major OS components should function as subsystems which may be selectively replaced by all levels of system users.

SECURITY (Cont'd)

User programming in any language. User level programs may be written in any programming language available on the system. This does not imply that every user should have access to every language, but rather that protection is not a function of the language used.

Multiple levels of data classification and user clearance. Various levels of users and data are able to coexist in the system without threat of security compromise.

Concurrently shared resources. Where appropriate, resources such as data, programs and physical devices may be concurrently shared.

Large data collections having complex relationships. The system provides the flexibility for handling large data bases in an arbitrary, but secure, manner.

SECURITY

The term "secure operating system" is not clearly defined since the idea of computer security is itself still being developed. However, a working definition is presented as follows:

> A secure operating system offers users a flexible set of features preventing the theft of data and programs, preventing unauthorized computer use, preventing disruption of system operation, and preventing unauthorized alteration of stored information.

In defining operating system security requirements, the following assumptions have been made regarding system features:

The system is a multi-use general programming system with the following characteristics.

Multiple simultaneous users. Many individuals may be using the system at the same time, with system resources being appropriately multiplexed among them.

Multiple concurrent modes of operation. Individuals may use the system interactively from a terminal device, may submit batch-type jobs either locally or remotely, or may interactively start a batch-type job execution.

DEFINITION OF TERMS (Cont'd)

The security matrix is that part of the security data base which describes exactly the manner in which each user (i.e., process) may access each object.

The basic protection mechanism, or security structure and mechanism, is that portion of the total system which has responsibility for maintaining a secure state within the system. Generally, it includes the following:

Any hardware device or feature the use of which could pose a security threat. In particular, the follwoing are included:

1. The hardware addressing structure and mechanisms; for example, the virtual memory system, memory address bounds checking, etc.
2. Mechanisms which change the security state of an executing program; for example, user/supervisor mode control, the protection ring mechanism, etc.
3. Error detecting mechanisms; for example, I/O device controllers, parity checking mechanisms.

Any software the use of which could pose a security threat. The following are included:

1. The security data base and any software which directly accesses it.

DEFINITION OF TERMS

The following definitions apply to terms used in this section.

Within a computer system, security is that state which exists when users of the system have an acceptable assurance that they can neither affect nor be affected by the state of other users, except by a known, well defined set of rules. Note that this definition recognizes that absolute security is not attainable, and that the term may validly mean different things to different people.

A security object, or simply object, is an entity to which access is controlled by the system.

A data object (itself a security object) is the basic unit of protection for collections of data within the system. The particular definition may vary from system to system; for example, it may be a file in one, and a segment in another.

A process is a program, or a collection of programs, in some state of execution.

Intra-process control is the means by which the security state may change within a single process. The change from user state to supervisor state is an example.

An access request is an explicit attempt by a process to gain access to security object.

The security data base is the total collection of data which is critical to maintaining security within the system.

PROCESSING PROTECTION

System from User

1. It is a mandatory requirement that operating system processes and data be protected from damage or unauthorized access by intentional or accidental actions of any user process.

User from System

2. It is desirable to protect user processes and data from damage or disclosure due to intentional or accidental actions of the operating system. This should be available as an option; speed penalties are acceptable in obtaining this feature.

System from System

3. It should not be possible for any operating system process to damage any other OS process, or data owned by any other process. Each OS process should be impervious to damage due to errors in shared data, or in data being passed to it by another process. This is a goal, to be compromised only by speed requirements. Provisions must be made to add features implementing this goal as technology permits.

User from User

4. No user process shall be capable of damaging or disclosing another user process or its data. This includes global code which is shared by two or more processes.

DEFINITION OF TERMS (Cont'd)

2. Intra-process control not wholly covered by hardware; for example, program calls, aborts, termination and errors.
3. Access requests not wholly arbitrated by hardware; for example, file opens, requests for I/O, etc.
4. Critical processes; for example, system start-up, recovery, the handling of access denials, etc.

CONSTRAINTS

General

1. It shall be considered impossible for the OS security apparatus to be more secure than that enforced by operational management (e.g., if user identifiers are considered unique then it is an operational requirement to maintain that uniqueness).

Federal Requirements

Requirements to be supplied.

Commercial Requirements

Requirements to be supplied.

Compromises/Tradeoffs

Requirements to be supplied.

Security Mechanism Failures

Requirements to be supplied.

DATA BASE PROTECTION

Private Usage

1. Within the limits imposed by site management, IPLOS must guarantee private data bases to be accessed only by those users designated by the data base owner, and only in the stated manners.

Public Usage

2. Within the limits imposed by site management, IPLOS must guarantee public data bases to be accessed only by those users designated by the site management and only in the stated manners.

Shared Data

3. Access to shared data shall be regulated by IPLOS. All necessary interlocking services for user level control must be provided.

4. There is no requirement to prevent malicious destruction of shared data by an authorized user of that data.

HUMAN ENGINEERING

Requirements to be supplied.

PERFORMANCE

Performance Goals

Requirements to be supplied.

Performance Monitoring/Measurement/Reporting

1. The operating system must provide complete and detailed self-measuring facilities and must support utilities which present this data in a meaningful fashion thereby allowing tuning of the system by the system vendor and the site managers. Individual hardware components, individual software components, and total system performance must be monitored.

Performance Tuning

2. System tuning must be available at system generation time, initial start-up load time and at system execution time.

Operating System Parameters/Algorithms

3. System scheduling parameters must be tunable by site managers and operators.

Configurations

4. Performance tuning options must be provided that allow site managers and end users to take maximum advantage of available resources for both individual and multi-programmed runs.

Performance Modeling

Requirements to be supplied.