-- --

IEEE P1003.0 Draft 13 – September 1991

5.3 Information System Management

Responsibility: Don Folland, Neil Croft D

DEF : The following is outline material as place-holder for more substantive text.

5.3.1 Overview and Rationale

The following issues should be addressed in subclauses 5.3.1 and/or 5.3.2:

- — should support a variety of system management support scenarios (central management, dispersed management, or hybrid)
- — this section should address both distributed systems and standalone systems.
- — could be applied to Application Software or software components of the application platform.
- — support automated management and operation of the IT infrastructure
- — should address a wide variety of licensing scenarios

5.3.2 Scope

This category includes services and policies that address the administration of the overall information system required by any organization, including:

- — Information Management
- — Processor Management (e.g., Add new user)
- — Network Management
- — Configuration Management
- — Security Management (e.g., Authentication, Key Management)
- — Accounting Management
- — Performance Management

Administration services accessible from the API may have Programming Language or Language Binding service specifications associated with them.

These services are defined to provide system and network administrator portability.

5.3.3 Reference Model

< Consistent with Draft 11 Section 3.0 >

5.3.4 Service Requirements

The following services are required:

5.3.4.1 Software Installation and Distribution

The main types of software to be installed and distributed are application programs developed in-house, bought-in applications, and utility software and personal computer software packages. All software needs to be managed effectively from development or purchase through to the live environment. Unless the distribution and implementation process can be controlled automatically, or from the center using software tools, procedures must be in place to ensure that distributed software arrives when expected and is checked for authenticity in whatever way is practical, and that the software is brought into use when required. The main procedures involved in software distribution and installation are:

- — Central Software Control and Distribution (SC&D) staff to inform remote staff when to expect distribution software to arrive.
- — Recipients to report to central SC&D staff when the distributed software has arrived successfully.
- — Central SC&D staff to check that all software is received as expected at locations.
- — Central SC&D staff to issue clear instructions about when the software is to be implemented.
- — Location staff to report to SC&D when the software has been implemented. The release record on the Configuration Management Database will state which installations are to receive the release. This database must be updated to reflect the receipt and implementation of the release at each site.

5.3.4.2 License Services

The terms and conditions relating to the supply of software may place legal restrictions on the organization (e.g., no unauthorized copies to be made). It is particularly important therefore that the Configuration Management Database is updated with details of who holds copies of software items. This assists the organization in discharging its legal obligations and assists auditors in checking for the existence of unauthorized copies.

All authorized copies of licensed or purchased software that are made by SC&D staff must be allocated a unique copy number and recorded in the Configuration Management Database together with where they are located and who is responsible for them. Procedural restrictions should be introduced to prohibit the unauthorized copying of software, and regular software audits should include a check for any unauthorized copies.

5.3.4.3 Print Output and Distribution Services

Output and distribution packages control output production and distribution from the moment the output is planned to the time the user receives the print. The working criteria need to be set up first; e.g., define who receives the report and how much of the report the user gets.

The main functions are

- The report can be limited to parts wanted by the user.
- Multiple copies of the entire report, or of selected sections can be produced.
- Reports are grouped by recipient within delivery location.
- Reports for each job are spooled as a group when the job is complete.
- The number of whole reports and individual pages received by each user are recorded.
- Report production can be monitored and managed efficiently.

Output and Distribution packages should contain the following minimum characteristics:

- Printing and distribution of whole and part reports
- Status (queued, printing etc) of the report tracked
- Online viewing of reports
- Ability to archive report files
- Ability to support a wide range of printers
- Costing and charging functionality
- Security facilities

By using an output distribution package, the delivery of reports to the correct person at the correct location can be ensured. Paper, time, and IT resource are saved as the users receive only the parts of reports that they need, and can also view the reports online. The number of pages printed can be controlled. Reports can be tracked from the time they are created to the time they are delivered to the user, allowing good security monitoring.

5.3.4.4 Office Media Management and Backup/Restore

The main functions of magnetic tape and data cartridge management systems are:

- To provide automated support for tape housekeeping and maintenance including:

  - Allocating tapes and releasing them for reuse helping
  - To ensure even patterns of use where appropriate
  - Constructing and triggering cleaning schedules
  - Maintaining the security of data

— To help automate archiving (vault management) for offsite storage

— To help identify growth requirements

Vault management is concerned with controlling the movement of tape cycles from one storage location to another. As a tape cycle is used, the tape management system automatically logs a different vault identifier against each tape.

A backup strategy is required to control the frequency of backups and the way in which they are created; e.g., whole volumes to cartridge or individual files to tape.

The backups and restores of system and application software should be separate from the backups and restores of data. Software and library backups should be explicitly scheduled and the complete software item or library backed up. The schedule for backing up files must be fully documented, properly maintained and adequately safeguarded as the contents of the schedule are required for disaster recovery purposes.

5.3.4.5 Online Disk Management

The operation of disk management systems requires that they take account of a range of factors such as retention period, recovery, space fragmentation, disk overflow, file and record activity levels, and channel use. Some systems merely report against values or thresholds set, but increasingly they invoke corrective action. Typically, the corrective action is file and disk reorganization or file and data archiving.

If a disk management system is used, the constant monitoring and actioning of requests for disk space can be minimized. Disk space may be collectively pooled and unused space constantly reclaimed.

5.3.4.6 Job Scheduling

Scheduling involves the continuous organization of jobs and processes into the most efficient sequence, maximizing throughput and utilization to meet the targets set in service level agreements (SLA). Jobs are scheduled to ensure:

— SLAs and user requirements are met; e.g., certain jobs need to be run by a certain time

— Available capacity is used effectively; e.g., the workload run at any given time does not exceed the practical capacity.

The minimum functionality of a scheduler should include:

— A high upper limit for the number of relationships allowed between jobs

— The ability to schedule by calendar and criteria

— Workload balancing support

— Levels of security

— Ability to restart jobs

— Operator override capability

— Capability to model future workloads.

5.3.4.7 Processor Configuration Management

Configuration management consists of four basic functions: identification, control, status accounting, and verification.

Identification involves specifying and identifying all components of an IT infrastructure.

Control implies the ability to agree and ''freeze'' configuration items (CIs) and then to make changes only with agreement of the appropriate named authorities. Control is concerned with ensuring that none of the CIs shown is altered or replaced, and that no CIs are added without appropriate authorization.

Status accounting involves the recording and reporting of all current and historical data concerned with each CI. Status accounting maintains records of the current, previous and planned states and attributes of the CIs and tracks these states and attributes: for example, as the status of a CI changes from ''development'' through to ''test,'' ''scheduled to go live,'' ''live,''and through to ''archived.''

Verification consists of a series of reviews and audits to ensure that there is conformity between all CIs and the authorized state of CIs as recorded in the configuration management database (CMDB). It is concerned with checking that the physical CIs actually match the authorized system as described in the CMDB.

5.3.4.8 Network Configuration Management

To ensure the viability of network services the configuration of systems and services must be controlled and managed. Effective configuration management will produce a minimum risk environment.

Configuration management procedures must ensure that details are provided for network equipment and systems covering:

— Configuration activities—how to configure the network equipment

— Security controls

— Access controls

— Configuration history log

— Configuration authority

— Build details

— Fall-back and test records

— Management reporting requirements.

5.3.4.9 Distributed System Configuration Management

— authentication services for a distributed system environment

— distributed Naming Service Configuration

— distributed Time Service Configuration

— X Window system configuration

— Window/Session Manager configuration

5.3.4.10 Accounting

An effective cost management system should:

— Provide assistance in developing a sound investment strategy which recognizes and evaluates the options and flexibility available from modern technology.

— Provide targets for performance and measure that performance against targets. It should also enable management to monitor actual costs against budget.

— Facilitate prioritization of resource usage.

— Ensure the sound stewardship of all the assets employed in the organization, including the maintenance of appropriate records for control purposes.

— Assist management in their day to day decision making and enable them to make considered decisions as regards forward plans with the minimum of risk.

— Be flexible and capable of providing a fast response to changing business circumstances.

The essential objectives of apportioning costs are to:

— Recover from users the full cost of (IT) services, including the cost of capital.

— Ensure that IT users are aware of the costs which they impose on the IT systems.

— Ensure that IT providers have an incentive to deliver an agreed quality and quantity of economic and effective service.

5.3.4.11 Performance Management

— ensure that users have satisfactory performance, workload throughput, and terminal response times in accordance with their service level agreements

5.3.4.12 Capacity Management

An effective and efficient capacity management function contains at least the following elements:

— Performance management to monitor and optimize the use of current systems.

— A capacity management database that contains current and historic data of technical and business related interest. This database forms the basis for the provision of both tactical and strategic reports on performance and capacity.

— Workload management to identify and understand the applications that make use of the system. The understanding of workloads has both a technical and business related nature. This involves application sizing to accurately predict the performance and required capacity of new applications.

— Capacity planning to accurately plan the required hardware resource and associated cost for the future and to predict the effect on performance and capacity of both tactical and strategic plans.

5.3.4.13 Availability Management

— measures taken to ensure availability and reliability

— automated or manual measures taken to recover a failed system

5.3.4.14 Security Management

— configuration of appropriate ACLs for System, User Interface, Storage, Network, and application software services.

5.3.5 Standards, Specifications, and Gaps

To be provided.

5.3.6 OSE Cross-Category Services

— Security for remote print jobs

5.3.7 Related Standards

To be provided.

5.3.8 Open Issues

5.4 Fault Management

Responsibility: Rich Bergman

The trade-offs in this subclause involve:

— Testability and Verifiability vs. simplicity and time

— Confidence and reliability vs. cost vs. risk

— Downtime vs. availability

— Maintainability vs. higher system cost

These services allow the system to react to the loss or incorrect operation of system components at various levels (hardware, logical, services, etc.). The classical model of fault tolerance has a three-step approach. The three steps are fault detection, fault isolation, and fault recovery. Typically implementations divide these steps into multiple steps or integrate them into one or two steps. Additionally, fault diagnosis services support the other steps in the treatment of a fault.

Various fault tolerance strategies, such as checkpointing and voting, are implemented as a collection of services comprising one or more of the steps in the fault tolerance classical model. For example, services involved in implementing a three-node voting scheme will include a vote comparator service (fault detection), vote analyzer service (fault isolation/fault diagnosis), a service to pass the majority ‘‘answer’’ through (fault recovery) as well as a service to disable the faulty resource and reconfigure the voters (fault recovery/reconfiguration).

Fault Detection

Fault detection services are concerned with determining when a fault has occurred in the system. Fault detection services are both passive and active. Active services are those that attempt to determine the status of various system components by testing those components. Passive services, on the other hand, try to ascertain system components by passively gathering information and watching the behavior of the system.

Fault Isolation;

Fault isolation services attempt to determine the component at fault and segregate the faulty component from the rest of the system. Services may be shared between the fault detection and isolation service library in that they perform both functions.

Fault Recovery

Fault recovery services attempt to bring the system into a consistent state. These services may be very interrelated to the scheduling services, network services, and data base services, depending on the recovery scheme used.

Redundancy of resources is many times needed to support fault recovery. Resources may include data, process, processor, disk drive, etc.