U. S. Railroad Retirement Board

Distributed Operations Management Domain Architecture

Distributed Operations Management Domain Architecture
Table of Contents

Distributed Operation Management Domain Definition

The distributed operations management domain defines the hardware, software, data, and procedures necessary to manage and support RRB automated business systems.

This domain identifies controls, processes and components for the various technologies. It also includes the tracking and auditing of all technologies to ensure better performance and consistent delivery of service.

Domain Technology Categories

- Disaster Recovery
- Storage Management
- Operations Management
- Performance Monitoring
- Testing
- Change Management
- Asset Management
- Automated Software Distribution
- Job Scheduling
- Help Desk

Distributed Operations Management Domain Principles Summary

1. Establish a centralized, coordinated repository to document and manage all of the functionality and capabilities of tools and technologies. (Asset Management and Disaster Recovery)

2. Integration testing of all hardware, software, and procedures will be completed at appropriate levels following unit testing and before being deployed in production environment.

3. We will use standard change management practices and procedures to minimize the impact of changes.

4. All hardware and software documentation must be kept up-to-date and easily accessible to interested parties.

5. Disaster recovery planning must be addressed in each system during design and/or acquisition.

6. Wherever possible we will automate the scheduling and execution of processing jobs

7. Performance monitoring will be used where feasible to measure efficiency and effectiveness, capacity planning/forecasting to validate the results.

8. Help Desk software will be the repository of RRB “IT” problems and support the technicians and Help Desk staff in resolving the problem in a timely manner.

9. During development consider the best or most appropriate and cost effective storage media to satisfy the business need.

10. Automate software distribution for faster deployment to end points and consistent configuration across the user base.

11. Capture and use information on fault and event management to continually improve RRB operations

Domain Relevant Trends

- The RRB has initiated a policy with the intent to conclude business with one handling, "One and done"
- There will be an increase in telecommuting
- There will be an increase in customer choice of interaction with the RRB (Internet/web access/telephone etc.)
- Internal developers and vendors will present the RRB with evolving / multiple versions of software
- There will be greater mobility of workforce/virtual office
- There is an expectation of service - 24x7x365
- RRB will experience a changing workforce- loss of experience due to retirement, knowledge transfer
- Organizational reorganizations will continue
- The RRB budget will continue to shrink and impact the technical budget
- There will be a proliferation of tools for Distributed Operations
- Volatility in vendor market is expected to continue.
- Customer expectations will continue to rise.
- Outsourcing must be considered (OMB, A76)
- Pressure to integrate systems and share data will increase.
- NARA requirements will continue to impact disaster recovery plans.

Background of Distributed Operations Management Related Technologies at the RRB

Disaster Recovery

The Bureau of Information Services has been performing Disaster Recovery Testing since the mid 1970's. Currently, in our disaster recovery testing we attempt to duplicate the environment needed to run our daily activity. In our earliest disaster recovery planning our focus was limited to assessing the capability to duplicate the Mainframe batch processing.

In July 1981, we initiated a study that was performed by the MITRE Corporation, on Contingency Plans for the Railroad Retirement Board Computer Center. This study provided RRB with recommendations for improving their testing procedures and policies. This resulted in a contractual agreement with Comdisco for our Disaster Recovery Testing

As a result of the study we expanded the scope of our recovery scenarios to include the capability to simulate our business needs. This extended the efforts to include the reproduction of:
Hardware,
Software,
Data Communications (both Internal-at Headquarters and External-Field Service,
Other agencies, and Vendors),
Storage Management (data and programs), and
Manual procedures.

In the late 1980's, the Railroad Retirement Board contracted to use Comdisco's Wood Dale, Illinois mainframe facility for our offsite testing. This provided a more compatible environment with our Hardware and Software needs. During these tests we started a minimal operating system and did limited testing.

Over the years our Disaster Recovery tests have expanded to incorporate more of our critical business processes. We duplicate much of our mainframe, communications and LAN Hardware and Software. We restore our mainframe Operating Software and do selected testing.

For LAN and communications support we have added Comdisco's Rosemont, Illinois facility site for testing. So far we have done minimal testing at the Rosemont site - bringing up one of each type of server and enough capability that designated personnel can determine the critical needs and get funds, equipment, and space allocated for further expanded capabilities.

The user community is only involved in a limited fashion in the actual test scenarios and the environment created is not necessarily real time (in terms of the moment of the disaster but at a point in time corresponding to the last back-up; we cannot guarantee that they will be able to recreate the environment from the last back-up to the moment of disaster without more user input and involvement and specifications/saving of needed files…).

Fault Event Management

In the 70's through the present we have handled fault event management by recording problems and/or events using paper documents called problem reports. This was a totally manual task. Hardware errors would be displayed on a console and required an operator to research the hardware error and call the appropriate vendor and report the problem. The Operator would fill out a initial problem report, work with the user, developer, system programmer or vendor to solve the problem and document the solution. Often times the problem was solved and the solution wasn't always documented. This posed a problem for solving future error events.

During the late 80's to present we became a little better at our recording and documenting of problems. Likewise, hardware vendors developed a "Phone Home" feature which would automatically capture an error message, and place a call to the appropriate vendor.

Automated Operations Software also includes a messaging facility which eliminates most informational messages that do not need operator intervention from displaying on the console. This allows the operators to deal with critical messages that require operator intervention.

However, we do need a total solution by obtaining a problem/change management software package that everyone at the RRB could use. By doing so we would have a centralized repository of information on events, and or problems and this would allow us to produce more meaningful reports whenever reporting on problems or changes to any system both software and hardware.

Performance Monitoring

Performance monitoring of the mainframe system is relatively new at the RRB. Only with the last few "systems" running OS/MVS did we actively do performance monitoring. The early systems did not have the sophisticated multi-tasking that we now have and all jobs were run in a batch mode and very few files were kept on disk.

With the advent of the Time Sharing Option (TSO) and a dramatic increase of user files on disk, system performance became more visible to users at a TSO terminal. IBM added performance parameters to various operating system components, which controlled how time was shared between online users and batch jobs. Along with that came techniques that "System Programmers" could use to optimize file placement on disk, physical disk location in relation to disk controllers and dispatching priorities for both batch and online tasks.

At first, computer manufactures were reluctant to publish performance numbers for their equipment. Increased performance was accomplished by buying bigger systems.
As "System Programmers" dug deeper under the covers of the operating system, user performance enhancements entered into the picture. Users groups were established and lobbying began to change the

operating system. One of the 1st users enhancements was the “7 dwarfs”. These were 7 user modules that modified the operating system to increase performance.

Shortly thereafter, software companies sprang up to address different performance problems. These companies provided the much needed support to extend the life of the mainframe and the operating system. System Programmers could now monitor and track performance factors and became sophisticated at system performance analysis. However, this almost meant that you had a full time position dedicated to enhancing performance.

In the 80’s we purchased a package from Boole and Babbage called CMF (Computer Monitoring Facility). Initially we determined various “High Water Marks” for disk and tape usage, TSO usage, and overall system usage. The package extracted information from the IBM System Management Facility (SMF) records and recommended changes to the different operating system parameters. A special unit was established to review these reports and recommend changes to the various system components. Disk files would be relocated and other system parameters would be changed each month. We then purchased a package called TSOMON. This package dealt specifically with TSO. Eventually we would run out of things to change and it was inevitable that we needed to get a new system. Many of the reports produced by CMF were used to justify buying a new system. Eventually TSOMON outlived it’s usefulness and it was discontinued.

As a result of many of the early user recommendations and today’s microprocessors , we now have more efficient systems. Today there is a new type of performance monitoring that is done. We still use CMF although it is now distributed from BMC Software. It now has an online component that can be used in real time if problems arise. CMF extracts performance records (Type 70 SMF records) from the operating system and passes them to the “analyzer” component and highlights problem areas.

IBM has added the Workload Control Manager (WLM). This is a sophisticated set of programs that automatically change parameters in the System Resource Manger (SRM) based on how we want our system to perform.

IBM has also added System Managed Storage (SMS). This program controls disk usage and where files are placed on disk. As disk space gets saturated, files are moved from Management Level 1 (ML1 – uncompress) to Management Level 2 (ML2 – compressed and eventually Management Level 3 (ML3 – Tape storage). This has greatly reduced the time needed to monitor disk space and performance.

An additional tool we recently purchased from BMC software is RESOLVE. This product will work with SMS to better manage file placement.

Our Virtual Tape Storage Subsystem has its own performance software that monitors and optimizes the use of our virtual tapes.

Since mid-1995 the Data Communications section has used NetView to monitor the performance of the IBM communications environment.. Since 1997 the Data Communications section has used CiscoWorks to monitor the router activity in the network environment. Since 1999 the Data Communications section has used HP OpenView, and Visual-UpTime to monitor the activity in the wide-area network environment.

In the LAN environment, all NOS (Network Operating System) for Microsoft and Novell have performance monitoring capabilities that can be turned on. They are usually not active because of the computing power they consume.

In the Data Communication area we use a program call RMON. This program measures terminal time.

Testing

Initially, starting in the 60's, automated data processing consisted basically of batch processing. There were no standard rules for testing. In some cases, production data was used for testing.

As new technology came into use in the late 70's – early 80's (terminals, Data Base Management systems, On-line program development, on-line applications processing, PC's emulating terminals, etc.) the testing process became more complex. Testing tools were purchased to help manage this complex environment. Government-wide mandates required that testing be done using test data.

With the advent of the PC and PC applications in the 90's and multi-faceted platform interfaces (mainframe/network), the testing process became even more complex.

The testing process for both hardware and software now entails:

1. Performing a product evaluation(pilot study) to test for compatibility with our current systems. This includes peripheral equipment, such as printers, scanners, etc.;
2. Installing the product on a test-bed which includes designated test servers and desktops;
3. Performing parallel testing;
4. Performing integration testing;
5. If pass test, fully implement;
6. If fail test, troubleshoot the problem and re-test.

Additional software testing tools, such as, Compuware Abend-Aid, Ads Alive, Viasoft Workbench, came into use.

Though, even now as the test process is more sophisticated, there still remain gaps. More consideration should be given to using more consistent testing practices and training of personnel using those practices and techniques, such as including testing at all levels, i.e., unit testing, integration testing.

Change Management

"Change management" as defined by the RRB means (1) changes that are mandated as a result of a change in the law and (2) changes that are initiated by the agency and are planned and managed; they are aimed at implementing new methods and/or systems as a result of changes in the (IT) environment.

Based on the above definition, in the early 60's, the RRB has taken two separate approaches to change management. If the change was the result of legislation, the RRB followed an orderly policy of reviewing and analyzing the change, training of personnel to implement the change and, finally, implementing the change. If the change was initiated by the agency (improvement or modification), there was no real structured approach. For example, if a decision was made by an organizational head to automate or purchase new equipment, the organizational head would meet with other organizational heads to sell his proposal; the matter would be taken up the management hierarchy to the Board Members.

Eventually in the early 70's, the ADP Steering Committee was formed to take on new proposals (Changes). The ADP Steering Committee consisted of bureau heads from each of the major bureaus. They met once a month to officiate approval of major ADP purchases or procedural changes. They were responsible through the leadership of Systems Development Chiefs in Data processing for developing a set of standards (ADP Standards) that outlined the steps (System Life Cycle Phases) that must be followed when developing or making changes to automated data processing systems. Those standards (with current modifications) are still followed today.

Starting in the early '80's, the change management practices became more refined. The project management system, PAC II, was introduced. This tool was a planning tool used to manage technology related projects in the Bureau of Data Processing. Projects were broken down into steps or milestones with planned target dates. The project management system called WINGS replaced the PAC II system.

In this same ‘80’s time period, the project team concept was introduced. Projects were worked on in teams consisting of a project leader, a systems and/or program analyst and a user bureau analyst.

As the RRB moved into the late ‘80’s and early 90’s and broadened its technology, the RRB (as an agency requirement) set its IT objectives and goals in the RRB Strategic Plan and the IT Capital Plan.

With the fast changes in technology in the mid-90’s, change management practices became somewhat more involved. The ADP Steering Committee is still involved in making decisions; though the Chief Information Officer (CIO) is the responsible head of the ADP Steering Committee. Once a major investment is approved, an elaborate series of steps are followed, which include:

1. Preparing a detailed Request for Proposal;
2. Doing a pilot study.
3. Doing tests to see how the change interacts with other RRB systems. (Change management tools such as, CA ENDEAVOR, CA IDMS IDD, CA PANAVALET, Microsoft SOURCESAFE, IBM SMP-E, and RRB TECH(NICAL) CHANGE are used to identify, organize and manage changes to software and hardware).
4. Training appropriate personnel which also includes procedures; and communicating the change via E-Mail or through the RRB Intranet to all personnel affected.

Asset Management

The Bureau of Supply and Service has always had the responsibility of managing the agency's fixed assets. The Bureau of Information Systems has always provided a tool in the form of a mainframe database software package to help them manage their data.
Before 1980 an IDMS database product (AMIS) was used. Currently and for approximately the past 11 years, a CICS-VSAM based product (FFS) has been used.

BSS primarily tracks information relating to the organization or bureau that requested purchase, the purchase price, the date purchased, the actual purchase order number, and a description of the equipment generally limited to the manufacturer, model, brief description, and serial number. This equipment is tracked at purchase, during transfer from one location to another, and when the equipment is finally surplussed or excessed.

About 10 years ago, as the number of personal computers begin to expand, the User Computer Services section of BIS which is responsible for installation and support of desktop PC's and networks, determined it was necessary to develop it's own database. The purpose was originally not to track fixed assets, but as a tool to provide statistical, technical, and warranty information for personal computer specific hardware and software. The database was originally created using the DOS based Dbase software package, and later converted using a Windows based Paradox software package. An initial audit was conducted by U.C.S. to capture the desired information for all personal computer hardware and software in the agency. The data was then entered into the Paradox database.

It became evident that the best way to capture the data for newly purchased hardware and software, was to complete the data entry process as it was received and before it was deployed. The Bureau of Supply and Service generally receives all equipment (including software). But since they did not need to, nor did they have the technical background to track the data required by U.C.S., it was agreed by both parties that all PC related hardware and software would be "funneled" through U.C.S. before deployment.

U.C.S. completes the data entry for the Paradox database, and then prepares a form which includes the required fixed asset data. This form is forwarded to Supply and Service where there data entry procedure is completed.

A new software package called Wise Track has been purchased for the purpose of consolidating these two packages. The software uses an SQL database that will reside on a Windows NT server. As of this date, the package has been received and is under review for installation.

Automated Software Distribution

The Railroad Retirement Board has used several methods to distribute software to desktops and servers. These methods have included:

- Installation on the desktop from a diskette or CD
- E-mail of an executable file that the recipient opens and executes
- Systems Management Server (SMS)

The first two methods are heavily dependent upon manual intervention.

The first method, installation on the desktop from a diskette or CD, requires first the availability of the diskette or CD. In the instance of commercial software, the installation materials are provided with the purchase of the software. In the instance of RRB developed software, staff must also create the necessary diskette or CD. This method also requires that BIS personnel visit each workstation to install the software directly to the desktop PC. These visits are time consuming. They must be scheduled and coordinated. They use the same resources that could be used to resolve Help Desk problems. They are disruptive to the work environment because users must stop their work to allow the installation to occur. In addition, this method prolongs the installation of software because each installation requires a BIS installer and only a limited number of installations can occur at a single time.

The second method, e-mail of an executable file, requires that the users read the message and successfully execute the file. This method eliminates the need for the time consuming, special desktop visits but it has other problems. It is still a manual process for the user to open the message and execute the file. There is no assurance that the users will complete the installation in a timely manner. If the user is unsuccessful with the installation, the Help Desk must become involved in correcting the resulting problems. With this method it is difficult to determine when all of the installations have been completed successfully.

The third method, Systems Management Server (SMS), "pushes" the software from a server located at 844 Rush Street to each of the field office servers. An "advertisement" is sent to each of the individual office workstations advising that there is new/updated software. Opening the advertisement and clicking on the "click here to install" button moves the software from the server to the workstation. SMS allows for remotely monitoring to ensure that all installations were successful. Currently, there are two shortcomings with SMS. RRB is experiencing problems with the SMS server and SMS was only intended to be used to distribute software to field offices. SMS was not purchased for or intended to be used to distribute software to workstations at 844 Rush Street.

Job scheduling

In the early 1970's, job scheduling was a manual process done by computer operators. Their basic tool for the task were run sheets which documented the number of tape drives that would be needed for the job and any job-dependencies. If the right amount of tape drives were available the operator would be able to release the job to run. Back then multi-processing was under the control of the operator. The Operator would look for and run as many jobs as he could at a time based on dependencies and the number of tape drives available. This process was time consuming, tedious and error prone.

In the late 1970's, we purchased a scheduling package by Value Computing Corporation. Because this scheduling package was not an automated system, it was not very flexible. This package required a “scheduler” to define all the system hardware and job profiles to the schedule in order to build a database of what each job needs to run. Then on a daily basis the “scheduler” would have to define each job to run for any given day to the schedule. This task took much time to complete. The product of this process would be given in hardcopy form for the scheduling of the 2nd and 3rd shift work.

This tool was an improvement on run sheets being used to determine which jobs could run together, but did not always provide a good job mix. Also, when a problem occurred on 2nd and 3rd shift it would invalidate the

schedule. Because there was only one “scheduler” on 1st shift, no revision schedule could be run for the rest of the night. This process wasn’t as useful as it could be because it was not an automated schedule giving the flexibility needed to produce revised schedules quickly.

In the early 1990's, we purchased a scheduling package called Control-M. This system is still in use today. This scheduling system allows you to build job profiles and it schedules the jobs based on job dependencies, time restrictions, calendar dates, etc. This provided the operators with a truly automated scheduling system. Job are scheduled and tracked by the scheduling system. This system is working fine and we are getting more sophisticated in the way we schedule jobs using logical statements that are placed in the JCL to either schedule jobs or flush jobs depending on processing outcomes.

Help Desk

In the 1960’s automation at the RRB was just beginning and all equipment was under the Bureau of Data Processing and Accounts. Any problem calls (help calls) were the responsibility of that Bureau. BDPA staff including Computer Operations, Tech Support and Systems Development handled calls.

In 1978 the Data Communications Section and the Database Administration Section were established as separate entities from Technical Support and handled calls related to those areas.

In 1980, the Data Communications Section and Technical Support were merged. In 1984, the Data Communications Section was again established as a separate unit. At this time, the Board established a network connecting all ninety-four remote offices with Headquarters. The Data Communications Section was responsible for this wide area network as well as the ‘in house’ network of terminals, printers, and other communications equipment.

As the Board began acquiring PC’s in the early 1980’s, a new section was established to handle PC related issues and problems. It was called the Customer Information Section (CIS). CIS eventually evolved into the present day User Computer Services (UCS). This group handled calls related to PC equipment they were responsible for.

In early 1990 user bureaus established ‘PC Specialist’ positions to handle some PC issues within these bureaus.

In 1995, the Office of Programs established the Systems and Technology Development (S&TD) section. One of the responsibilities assigned to S&TD is responding to PC/LAN help calls within the office of programs.

UCS is now divided into three sections: LAN/WAN administration, Desktop Administration, and Data Communications/Help Desk.

In the current environment, all help calls are received by Data Communications staff and documented in a help desk tracking and administration software program (MAGIC Total Service Desk by Network Associates). Any help desk issues that cannot be handled and closed by the DC/HD unit are assigned to the appropriate section for correction.

Detailed Domain Principles

Domain Principle 1

Establish a centralized, coordinated repository to document all of the functionality and capabilities of tools and technologies and that will allow us to manage them.
Rationale:

Reduces cost by avoiding the purchase of products that duplicate functionality/capability (better buy/build decisions)
Maximizes the value of investments by identifying the functionality of currently owned products that may address current needs.
Speeds project initiation by simplifying research on functionality and capabilities of our existing tools and technologies.
Improves support of established products and leads to greater expertise in those products and reduces training requirements.
Enables expanded use of current products leading to a smoother running IT and user organizations
Reduces integration complexity
Help track license assignments and observe agreements

Implications:

Requires that the repository be used in design and procurement processes
Need to determine the responsibilities for managing and administering the repository
RRB may have a tendency to use a less-than-ideal product or tool because “we already have it”
RRB may have a tendency to use older or out-dated technology
Requires communication and education of repository users/contributors
Requires that information about product functionality would have to be determined, captured and maintained.

Domain Principle 2

Integration Testing of all hardware, software, and procedures will be completed at appropriate levels following unit testing and before being deployed in production environment.
Rationale:

Minimizes surprises (compatibility issues, access problems, potential help desk calls, etc)
Allows for correction of mistakes and problems before deployment

Avoids ‘chain reaction’/ cascading of problems
Avoids loss of productivity
Allows more time to assess and correct the problem (avoids crisis thinking)
Provides more consistent testing

Implications:

- Need a 'separate' testing environment
- Need consistent definition of testing
- Need consistent approach
- Need a comprehensive set of tools for testing
- Helps avoid overlooking testing in system development life cycle, especially in non-applications development
- You can't test everything so will still need 'contingency plan'
- Need to include appropriate technical, user, and subject matter expert staff
- Need user appropriate technical and/or user sign-off before deployment
- Need to consider versions and configurations
- May extend the period to implement and deployment
- Need for training in testing, especially in the PC/LAN environment

Domain Principle 3

Use standard change management practices and procedures to minimize the impact of changes.

Rationale:

- Saves dollars in the long term
- Able to use lessons learned (both mistakes and successes).
- Greater coordination equals fewer problems in all areas.
- Better informed customers increases the likelihood of acceptance of the changes.

Implications:

- Requires effective communication between all parties.
- Need to bring the right knowledge base together
- Need to develop the right sponsorship
- Need proper change management support tools
- Need a holistic view of impacts of a change
- Non IT initiated changes can affect IT
- Increased cost in short term
- May increase deployment time
- Enables the Help Desk to be more effective because they'll have a record of the changes that have occurred.

Domain Principle 4

All hardware and software documentation must be kept up-to-date and easily accessible to interested parties.

Rationale:

- Makes installation and support easier
- Enables Help Desk to be better equipped to handle problems
- Aids in debugging and upgrading software and hardware
- Decreases dependency on one individual
- Provides a road map of where you're going
- Enables you to identify interdependencies

Implications:

Documentation must be acquired or created.
Adds time and cost to initial development
Helps to establish requirements for solicitations for contractual services.
Enables communication (knowledge transfer to outsourcers and new employees)
Enables more accurate auditing
Need a common repository with appropriate security
Need to establish a structured approach for all documentation (application programs, installation procedures, requirements for contractual services, etc)
Requires enforcement of all procedures
Need appropriate documentation before deployment

Domain Principle 5

Ensure that disaster recovery planning is considered in each system during design and/or acquisition.

Rationale:

Design/Acquisition decisions support disaster recovery making it easier to plan, test and execute disaster recovery activity
More timely identification of additional requirements (contractual, hardware)
Minimizes down time and recovery time
Enables better understanding of TCO (disaster recovery is considered in design to prevent the need to retrofit after deployment)

Implications:

May alter some of our design decisions
Develop understanding of pros/cons of various recovery approaches
Best done using industry standards
Consider implications on business partners
May increase /change test times and requirements
Need to establish and communicate priorities

Domain Principle 6

Wherever possible we will automate the scheduling and execution of processing jobs.

Rationale:

Better utilization of time and system resources
Improved controls with less reliance on people
Improved customer service
Better utilization of manpower
Improved throughput
Allows for unattended operations after normal business hours (extending the business day)

Implications:

Might push problem resolution to next day
Need to acquire job scheduling tools for the PC/LAN environment
Could change current scheduling practices (i.e., promotes device independence).
Will require someone to determine the scheduling requirements, establish schedules, periodically reassess the schedule as new things come and resolving disputes over scheduling.

Domain Principle 7

Performance monitoring will be done to measure efficiency and effectiveness, and provide the data that will be used for capacity planning and forecasting.

Rationale:

- Metrics can be used to improve efficiency
- Metrics can be used to identify opportunities to reduce cost
- Allows agency to forecast needs (expansion, upgrade hardware/software, etc)
- Allows proactive problem identification and mitigation thereby providing better service to customers

Implications:

- Ensures appropriate use of tools/technology
- Validates efficiency and projections (forecasts, capacity planning)
- Helps identify best practices
- Provides more complete information for audit reporting
- Provides statistical information to assess the impact of various technologies on processing
- Need to monitor the 'entire process' (components serially, and also process in conjunction with other processes)
- Performance monitoring adds processing overhead
- Need to define performance metrics for internal and outsourced activities
- Identify the organizations responsible for this function

Domain Principle 8

Help Desk software will be the repository of RRB problems and support the technicians and Help Desk staff in resolving the problem in a timely manner.

Rationale:

- Establishes a knowledge base to enable quicker and more consistent resolution of problems
- Establishes reassurance to the users and avoids multiple follow up calls
- Enables HELP DESK to identify trends or larger issues
- Assists in training and skills development for Help Desk staff
- Enables proactive identification and mitigation of problems (e.g. virus)
- Assists the agency in determining user training requirements from problem history

Implications:

- Supports the development of standard problem description and resolution effort
- Need to establish Service Level Agreements
- Help Desk software needs to be accessible to Help Desk personnel and users
- Need to support non-RRB personnel
- HELP DESK technicians and users both need appropriate training in software
- HELP DESK needs appropriate staffing levels and skills
- Need to identify subject matter experts across various technical areas and applications and establish a system for accountability and recognition
- Need Software available 24x7
- Need to create and utilize 'history capability'
- Need to coordinate with Asset Mgmt. Information repository
- May need to change how the helpdesk is handled at the RRB.

Domain Principle 9

During design consider the most appropriate (quality and cost) storage media to satisfy the business need.

Rationale:

Enables stored information to be available to the appropriate parties at the appropriate time
Enables better forecasting of storage needs
Reduces overall storage needs and 'mini-master' extracts

Implications:

Users, Platform, Application and Application Development staff must jointly consider media needs early in the SDLC process.
May require IT to support additional media technology.
Develop business need decision criteria
Need to maintain awareness of storage technology trends, new technology
May require more time in design as to business needs
Requires that business needs include our business partners
"Best" storage media may not be the most accessible
Legislation/regulation (e.g. Records management regulation) or external forces may affect decision.

Domain Principle 10

Automate software distribution for faster deployment to end points and consistent configuration across the user base.

Rationale:

More efficient use of user and IT staff time
Enables more consistent customer service
Reduces reliance on users
Improves software version control across customer base
Help Desk will be more likely to quickly pinpoint and resolve problems

Implications:

Need to acquire tools that enable automated distribution
Will have to take into consideration different versions of software in use at the RRB (coordinate/asset management)
Take distribution into consideration during planning and design
Must be disciplined about maintaining standard configuration
'Encourages' everyone to upgrade to latest version: eases maintenance
Requires more consistent and explicit communication
Operating system upgrades require compatibility testing of all existing applications (requires identification of all applications)
Allows us to do preventative maintenance (e.g. virus definition upgrade, etc.) on a regular basis by removing user dependence

Domain Principle 11

Capture and use information on fault and event management to continually improve RRB operations.

Rationale:

- Avoid reinventing solutions by benefiting from lessons learned
- Allows us to anticipate problems
- Allows for quicker response and resolution (contact right person)
- Reduces cost over long term
- Enables us to be proactive
- Enables longer term planning, problem prevention and mitigation (cause and effect)
- Helps identify event/problem indicators

Implications:

- Need tools to capture data, a repository to store data, and access by appropriate parties
- Assign appropriate personnel to administer, and manage the process
- Establish teams to respond to various types of events
- Needs to coordinate with Help Desk
- Need to identify conditions that are fault-events (thresholds, performance monitoring data, etc.)
- Helps justify investment or purchase that would permanently fix the problem

Pattern 1

Configuration Management

Purpose

Standardize procedure for installation and deployment of software and hardware (desk-top, printer, equipment, etc.).

Applicability

This pattern will be used in the installation, upgrade and maintenance of all hardware and software.

Installation is defined as a new introduction of software or hardware.

Upgrade is defined as a modification to existing software or hardware.

Maintenance is defined as a correction or repair to existing software or hardware.

Assumptions

We will attempt to standardize desktop configurations (types of hardware and software) based on organizational unit and job duties.

Modifications are controlled, documented, and located in a secure central repository available to the appropriate personnel.

Configurations are kept current and procedures are dated and show the identity of the person making the change.

When changes are made to a procedure or program they must be published and the appropriate personnel notified- examples include -

Mainframe - a method of communications (notification procedures) must be made from installer to units impacted

Server - a method of communications (notification procedures) must be made from installer to units impacted

Desktop - a method of communications (notification procedures) must be made from the unit responsible for development to the unit responsible for installation.

The program developer will follow the procedures in the systems development life cycle (documentation will include but not be limited to - location of programs, user manual, caveats, installation tips, etc.)

All issues associated with interaction and compatibility of hardware and software will be addressed prior to installation.)

All program developers regardless of organization will follow this pattern.

Structure Overview

Create and populate configuration repository
Request
Approved
Acquire Product
Install Base Product
Test Base Product
Review repository for profiles and procedures

Apply Authorized Modifications (according to procedures)
Test Authorized Modifications
Deploy Product
Document Configuration
Request Change

Detailed Pattern Description

None.

Benefits

Makes installation easier, more consistent and faster
Provides input to procurement process
Simplifies training of supplemental staff (i.e., Contractual services, details, volunteers, etc.)
Identifies all the configurations used
Provides a framework for documentation consistency
Facilitates accountability

Consequences

Repository may not include all exceptions
Creation of the repository will entail additional cost
May require additional resources (staff, hardware, software).

Variations

- Pilot product evaluation
  - Use standard pattern up to “deployment”
  - Document results in the form a report to appropriate personnel overseeing the pilot implementation.
- Prohibit unauthorized software
  - Authorized software will be listed in our asset management database
  - Consider the appropriate configuration in the repository
- “Plug ins” (real player, omniform filler, adobe reader, etc.)
  - Determine how to handle and where to store for future authorized use

Related Patterns

None

Known Uses

None

Pattern 2

Operations Management Documentation

Purpose

Standardize procedures for applications, installation, maintenance and deployment of software and hardware (desktop, printer, equipment, etc.), document those procedures and make them accessible to the appropriate personnel.

Applicability

This pattern will be used in the installation, upgrade and maintenance of all software and hardware.
Installation is defined as a new introduction of software or hardware.
Upgrade is defined as a modification to existing software or hardware.
Maintenance is defined as a correction or repair to existing software or hardware.

Assumptions

Desktop configurations (types of software and hardware) should be standardized based on organizational unit and job duties as prescribed by the agency standard.

Modifications are controlled, documented, and located in a secure, central repository available to the appropriate personnel.

Configurations are kept current and procedures are dated, showing the identity of the person making the change.

When changes are made to a procedure or program they must be published and the appropriate personnel notified- examples include -

Mainframe - a method of communications (notification procedures) must be made from installer to units impacted

Server - a method of communications (notification procedures) must be made from installer to units impacted

Desktop - a method of communications (notification procedures) must be made from the unit responsible for development to the unit responsible for installation.

The program developer will follow the procedures in the systems development life cycle (documentation will include but not be limited to - location of programs, user manual, caveats, installation tips, etc.)

All issues associated with interaction and compatibility of software and hardware will be addressed prior to installation.

All program developers regardless of organization will follow this pattern.

Detailed Pattern Description

How Document Repository Will Be Used

- Create and populate documentation repository
- Request for service made (e.g., change to current application, new hardware, added functionality)
- Request is approved by the appropriate party
- Service request is acted upon (e.g., acquire product, system modified, SDLC, etc.)
- For commercial products, install base-line product
- For commercial products, test base-line product
- Review repository for profiles and procedures
- Apply Authorized Modifications (according to procedures)
- Test Authorized Modifications

- Deploy Product
- Document changes

Documentation Categories Needed in the Repository
Detailed categories for the repository must be defined
These should include but are not limited to:

- requester ID
- authorized by
- date assigned
- person or persons assigned
- name of the product or service
- version
- changed completed date
- description of change
- vendor/developer
- appropriate links to asset management
- dependencies

Definition of Configurations
Must develop definitions of configurations based on organizational unit and job duties as prescribed by the agency standard.

Benefits

Makes installation easier, more consistent and faster
Provides input to procurement process
Simplifies training of supplemental staff (i.e., contractual services, details, volunteers, etc.)
Identifies all the configurations used
Provides a framework for documentation consistency
Facilitates accountability
Prohibits unauthorized software (e. g., authorized software will be listed in our asset management database, consider the appropriate configuration in the repository, etc.)

Consequences

Documentation requirements must be defined
Repository may not include all exceptions
Creation of the repository will entail additional cost
May require additional resources (staff, hardware, software).

Variations

The following situations are allowable deviations to the standard pattern:

- Product evaluation
  - use standard pattern up to “deployment”
  - document results in the form a report to appropriate personnel overseeing the pilot implementation.
- Downloading “plug ins” (Real Player, OmniForm filler, Adobe Reader, etc.) to the desktop
  - determine how to handle and where to store for future authorized use

Related Patterns

None.

Known Uses

None.

Domain Participants

Domain Team Leader: Joe Hammon (Alternate: John Cunniff)

Line of Business Representatives: Denise LeSeur-Waechter, Chuck Mierzwa

Domain Participants: Patricia Lee, Edward Pyrek, Robert Johnson, Paul Klocek, Louis Spadavecchia, Colin Bruce

APG Representative: Sally Mui

Appendix 1: Domain Glossary

Term	Definition
Configuration Management	Configuration management is the detailed recording and updating of information that describes the relative arrangement of hardware components and software parameters used for installation and maintenance reference.
Fault event management	The identification, tracking, notification and automated control of hardware malfunctions, software errors, and resource conflicts that can occur in a computing environment.
Right sponsorship	Right Sponsorship - An individual or group that can secure the buy- in of key people who articulate the vision of how the change supports the mission and goals. It is someone who has the passion to bring the idea to reality.

Appendix 2: Conceptual to Domain Principle Matrix

	Relationship Between RRB's Domain Principles And Conceptual Architecture Principles																								
Domain Principle	***Conceptual Architecture Principles***																								
	CA 1	CA 2	CA 3	CA 4	CA 5	CA 6	CA 7	CA 8	CA 9	CA 10	CA 11	CA 12	CA 13	CA 14	CA 15	CA 16	CA 17	CA 18	CA 19	CA 20	CA 21	CA 22	CA 23	CA 24	CA 25
D-1	X	X	X		X	X	X		X		X	X		X				X			X				
D-2	X	X	X				X								X	X									
D-3	X	X	X		X										X			X							
D-4	X	X	X	X	X		X																		
D-5	X	X	X		X										X							X			
D-6	X	X	X	X	X													X		X		X	X		
D-7	X	X	X				X						X		X							X			
D-8	X	X	X				X																		
D-9	X	X	X		X								X					X		X					
D-10	X	X	X	X	X		X											X							
D-11	X	X	X	X	X	X	X					X			X			X							

Conceptual Architecture Guiding Principles:

1. Use guidelines consistent with the Federal Enterprise Architecture. 2. Support a single Enterprise Wide Technical Architecture (EWTA). 3. IT projects are to be consistent with the Enterprise Architecture. 4. IT projects are to be consistent with the Enterprise Architecture. 5. Reduce integration complexity. 6. Technical architecture must be extensible and scalable. 7. Manage information and data as enterprise-wide assets. 8. Validate information as close to its source as possible. 9. Enhance the ability to capitalize on and exploit business information. 10. Support multiple data types. 11. Make an informed buy versus lease versus build decision before proceeding with any new development project. 12. Require shorter development cycle times. 13. Keep current with emerging technologies and their applicability to enterprise architecture. 14. Maximize infrastructure asset reuse. 15. Sustain reliable connectivity. 16. IT systems will be implemented in adherence with the agency's security, confidentiality and privacy policies. 17. The agency will use a consistent set of security interfaces and procedures. 18. Reduce total cost of operation (TCO). 19. Extend E-Mail to Become a Corporate Information Exchange Vehicle. 20. Adopt Open Systems Standards. 21. Reduce duplicate information systems. 22. Reduce duplicate information systems. 23. Maximize and exploit Internet and Intranet technologies and approaches. 24. Integrate Enterprise Architecture into the investment management process. 25. Customer perception is a measure of the quality of the automation processes.