Migration Techniques & Strategies

Dilemma of Change

- Hardware Cost
- Software Cost
- Re-training of:
  - Systems personnel
  - Operations personnel
  - Users
- Business Impact Cost
- Cycle time to system implementation

Options for Platform Change

- Stay put ...Do nothing
- Purchase new hardware
  - Purchase new software ...Packages
  - Write new software ...Development
  - Migrate existing software

Stay Put

- Upside
  - Requires nothing to be done
  - Definitely ON Budget and On Time
- Downside
  - Maintenance becomes more expensive
  - Hardware & Operating system support no longer available
  - Minimal, if any, Operating System enhancements
  - Inability to interface effectively to new technology
  - Putting off the inevitable

Purchase new software Packages

- Requires extensive research in determining if package satisfies business requirements
- How much ($$$ and TIME) needed for any customization
- Licensing cost
- IS and User Training
- New packages for all current systems
  - High Risk of being over budget and late

Write new software Development

- Need to find qualified technical people
- Need to develop detail functional and programming specifications
- Size of project team may be unmanageable
- Probably most expensive and time consuming
- High Risk of being over budget and late

Migration of existing Software

- Locate and use automated tools
- least difficult
  - No new functions
  - No changes on logic
  - No enhancements
  - Few, if any changes to user procedures
  - Few, if any changes to other application interfaces
  - Business applications are positioned to interface with new technology

Migration of existing Software (continued)

- Best choice for on budget and on time
  - Usually lower software costs
  - Minimal retraining of Users
    - IS personnel have current knowledge of system and programs
    - Operations knows how system runs
    - Users know how system functions

Migration Process

- Based on selecting the option of migrating existing software to a new platform, how do I go about building a project plan that will support this process ?

The Migration Process Continued

- **Project Plan**
  - The Analysis phase
  - The Pilot phase
  - The Conversion phase
  - The Implementation phase

Analysis Phase

- Develop a complete Inventory, by type of the items to be converted
- Match source to object
- Identify when programs run
  - Daily; Weekly; Monthly; Quarterly; Yearly; Upon Request
  - Determine which programs you can discard and remove from the inventory list

Analysis Phase **(continued)**

- **Project Inventory Report**

Program#	Language	#lines	Freq	Type	comments
SAL10	Cobol	1000	Daily	Online	
SAL15	Cobol	2000	Mth	Batch	
SAL30	Cobol	2350	Req	Batch	**clear Y/E files**
SAL35	Cobol	800	Req	Batch	
S1000	JCL	350	Daily	Batch	
S1010	JCL	500	Mth	Batch	
datcon	Assembler	45	Daily	Daily	

Analysis Phase (continued)

- Investigate and select conversion tools
  - COBOL to: Acucorp COBOL, Micro Focus COBOL, OS400 COBOL,etc
  - Forms conversion
  - Job Control Language conversion
    - File overrides
    - Control cards
    - Sorts

Analysis Phase **(continued)**

- Investigate and, if necessary, select testing tool
- Establish training requirements
  - Both for hardware and software
- Establish conversion standards
  - Naming conventions
    - programs
    - files
  - Parameter usage

Analysis Phase (continued)

- Determine handling of data migration
  - How are we going to move the data?
    - FTP - Do we have large enough bandwidth?
    - Tape - Do we have compatibility?
  - On unloads, develop record counts and if necessary, depending on data sensitivity, hash totals
  - Record pre and post file counts and hash totals
  - Create programs to be able to repeat unload and load processes

Analysis Phase **(continued)**

- Character set concerns
  - ASCII to ASCII
    - Straight over. No translation of characters necessary, sequencing remains the same
  - ASCII to EBCDIC
    - Requires translation of character set
    - Packed fields requires special translation table
    - Sequencing different between ASCII and EBCDIC

Analysis Phase **(continued)**

- Establish procedure for tracking and reporting of program change control
- Identify programs to be converted for pilot phase

Pilot Phase

- Convert forms, programs, & JCL that make up the pilot
- Test converted programs
  - Verify screens look the same
  - Verify program functionality is the same
  - Verify output data and reports are correct
- Develop conversion guide document
  - Includes step by step process of getting the programs converted and tested

Pilot Phase

(continued)

- Develop detailed project work plan
  - Include estimates & schedules of each task
- Determine how the inventory is to be packaged for conversion
  - If possible, group the programs for conversion
    - By sub-system
    - By definable groups within sub-system

Conversion phase

- Organize project team
  - Must have someone on team with knowledge of current applications
  - Staff must be very knowledgeable of source platform
  - Staff must have some working knowledge of target platform
  - Project team must be kept manageable in size (Swat team)
  - Team works together (everybody wins)

Conversion phase (continued)

- Project team (continued)
  - Must believe project due date is achievable
  - Project team must include member of user and management community
  - Management team member must have decision authorization
  - Must have regular scheduled status meetings
- Training for the team

Conversion phase (continued)

- Provide all necessary collateral of group to be converted to team members
  - Current source modules
    - programs, Forms, JCL
  - System and/or program flowcharts
  - Run documentation
  - Scripts for testing
  - Test data
  - Test results

Conversion phase (continued)

- Freeze programs in group
  - Initiate change control
- Process
  - Convert the forms, programs, & JCL
  - Incorporate any manual changes that are required to get to clean compile
    - Changes should be made to pre converted code thus allowing re-conversion of module if any tool enhancements or tool fixes are needed, or re-conversion of programs due to changes.

Conversion phase (continued)

- Test converted programs
- Correct any system differences, preferably to pre-converted code
  - Format of system date & time, etc.
- Update ‘Issues log’ with any new issues that come up.
- Repeat conversion process until all are complete
- Write weekly or bi-monthly‘Status Report’

Conversion phase (continued)

- **Migration Issue Log**

Log#	Date Raised	Issue Description	Raised By	Need By	Assgn	St
xxx	xx/xx/x	xxxxxxxxxxxx	xxx	xx/xx/x	xxx	xx

Conversion phase (continued)

- **Project Status Report**

Migration Status Report
As of XX/XX/XX

Tasks accomplished this period

Tasks currently in process

Additional notes

Implementation phase

- Make sure there are no open items in the ‘issue log’
- Process ‘change control’ items
  - Re-convert and test programs that have changed. *(May need to schedule this step more than once through life of project)*
  - Freeze entire inventory
  - Changes that are made from here on out need to be made to both sets of code

Implementation phase **(continued)**

- Determine implementation strategy
  - Phase in the applications gradually
    - Additional temporary interfaces may need to be written and moved into production
    - Some data files on the target platform may have to be moved back to the source platform
  - Big bang approach
    - Keep all programs on the shelf until system testing and parallel are completed, Cut over all at once

Implementation phase **(Continued)**

- Create the production environment
- Update User Profiles
- Develop time table for unloading and loading of data
- Establish the data on new platform
- Assign Users to do some inquiries after data migrated to insure things look OK
- Pre implementation review

Implementation phase **(Continued)**

- Execute System test
  - Enter same transactions (Source & Target platform)
  - Verify online transactions
  - Execute batch run
  - Verify batch reports
  - Correct any converted programs that generated discrepancies & re-execute this step

Implementation phase **(continued)**

- Execute Parallel
  - Execute data cut-over
  - Enter online transactions
  - Verify online transactions
  - Execute batch run
  - Verify batch reports
  - Correct any converted programs that generated discrepancies & re-execute this step

Implementation phase **(continued)**

- Post project review
- Project sign-off