

Initial Functionality Documentation

Goals

The goals statements will be the basis for every feature of our software product. Everything that is added to this project must be in support of one or more goals.

- Perform Administrative tasks in VM from a GUI.
- Little or No CP/CMS command knowledge necessary to operate.
- Documentation built into interface. (use of tools tips, and in-line help)
- Security is high priority. No plain text network traffic over public LAN!!!
- multi-user and privilege support.
- Portable code. Run from many OS's, maybe even web interface. (Java)

Functionality

Functionality is a high level overview of everything we want our product to do. This section should be as specific as possible as to what features we want to have without getting into implementation detail or specific methods of “doing”. The specific methods of “doing” will be addressed in the specification documentation. The implementation will be discussed in the detailed design documentation. All three documents (Functionality Doc, Spec Doc, and Detailed Design) form the Design Documentation for the project.

- Device management (DASD, OSA, CONS, RDR, PRT, CPU)
 - DASD (P1)
 - Formatting (CPFMTXA) all types.
 - Support a batch format Queue. (Time saver)
 - Way to attach/detach.
 - View dasd owners, labels, vaddr, active references (Q DASD)
 - Device & controller I/O utilization.
 - What mdisks are on DASD, who owns them?
 - Whats free? Whats not?
 - Query Device type, and size in cyls. HOW???
 - Can we query a DASD controller?
 - OSA (P1)
 - Attach & Detatch.
 - Whats free? Whats not?
 - Check I/O throughput.
 - CONS (P2)
 - (What does Q CONS output mean??)
 - See user on console, Ip address associated with.
 - RDR (P3)
 - Query for satus info.
 - Contents of size of data in reader????
 - purgre readers.
 - PRT (P3)
 - (What does Q PRT output mean??)

- CPU
 - Query CPU ID's. (P3)
 - Dedicate/Undedicate to guests. (P1)
 - Check single cpu I/O usage. (P1)
- Other devices??? (P1)
- Guest management
 - VM Directory Profile creatormanager.
 - Create/delete profiles.
 - Creation of VM Guests
 - Default set of allocations of MDISKS.. or maybe profiles .
 - Validate cylinder ranged.
 - Add MDISKS.
 - Dedicate devices.
 - Setting Machine Type (combo box fr supported types)
 - Included profiles
 - CPU Statements
 - Set Storage and MAXSTOR.
 - Set Privilege classes
 - Deletion of VM Guests
 - Deallocation of DASD and other devices.
 - Modification of VM Guests
 - Allow modification of devices, passwords, storage, permissions, etc.
 - Manage CPU and I/O usage of individual users.
- Channel Path management (Status only?)
- Live System Status.
 - Free storage, swap, guests running
 - CPU Utilization
 - Support for actions taken in case of predefined “alert” conditions. (P2) via shell command execution.
- Global System Management
 - View and add Page and Spool as needed.
 - Support RACF.. build in support.
 - Eventually add generic ESM support? Future release.
 - Allow VM scheduler queues to be managed.
 - Possibly allow management of permission classes.
 - Define custom privilege classes???
- LAN Management
 - LAN (permanent or temporary) addition, deletion, restricted or unrestricted.
 - If restricted, adding guests to that LAN.

The Graphical User Interface

- Need to have multiple menu/button text profiles so an advanced user is not confused by the “dumbed down” descriptions and the newer user is not confused by cryptic VM names.

- Gui needs multiple view options (folder, tree...) IF we decided to do a “tree” style representation of the data.

Lan 'viewing' w/a 'map' – Ability to dragNdrop connect guests to a LAN.

Features for the Future (and related topics)

Priority #'s

- 0 – DO THIS NOW!!!

- 1 – Needs to be in first version.

- 2 – Would be nice... finish 1's first.

- 3 – May not be all that important.

- Script for auto download and punch of Linux image files for install purposes.

- Multi-user support.