Three+ Years with z/VM and Linux on System z

Richard Ralston
Humana Inc.
Where We Started – 2009

- Original justification
  - Reduced overall cost (considering hardware used for MQ only)
  - Increased performance
  - Increased utilization of available resources
    - 15 AIX/Win servers, avg. 10–15% CPU each
    - .8 of one IFL engine, 28 images @ 2–3% utilization each
  - Reduced memory
  - Reduced DASD
  - Increased failover and redundancy
  - Significantly reduced footprint (space, cooling, electric)

- There was a list of 10–12 other possible infrastructure candidates including
  - Communication Server
  - Oracle
  - HOD (Host on Demand)
  - DB2 Connect
Justification – Platform Viability

- Reduce overall cost (hardware, software, maintenance, footprint)
- Increase performance
- Increase utilization of available resources
- Increase failover and redundancy capabilities
- Increase scalability
- Quick provisioning
  - Less than 1 minute to bring up a new image
  - Add memory, dasd, cpu on the fly
  - “Rent” (OOCoD) another IFL engine for peak times
Z Platform Upgrades

<table>
<thead>
<tr>
<th>Model</th>
<th>Year</th>
<th>CPU (Ghz)</th>
<th>MQ IFL Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z9</td>
<td>2008</td>
<td>1.7</td>
<td>1.5</td>
</tr>
<tr>
<td>Z10</td>
<td>2009 (Jan)</td>
<td>4.4</td>
<td>.8</td>
</tr>
<tr>
<td>Z196</td>
<td>2010 (Nov)</td>
<td>5.2</td>
<td>.5 (Anticipated)</td>
</tr>
</tbody>
</table>

200% decrease in engine resources.
No engine or software cost for this increase.
Initial effort/POC

- 6 zVM LPARs
  - 2 prod, 2 test, 2 sandbox
  - 2 CECs with 2 IFLs each
  - zVM 5.3, SLES 10 SP2
  - Tivoli Omegamon
    - zVM for zLinux XE
    - ITCAMS for MQ
- No current zVM or Linux experience in house
- No useful monitoring, Capacity Planning etc.
- No Security – wasn’t critical during POC
- Used existing staff, adding split responsibilities
Determined to move to production environment

Additional needs caused a re-evaluation of the platform
  ◦ Cost of adding 3rd party security software licenses caused a revisit of using zVM and Linux on z
  ◦ Automation
  ◦ Initial cost saving not as robust as originally projected

Resolved to management satisfaction
Go Ahead

- Licensing issues resolved
- Decided to move MQ
  - 1 to 1 move
  - No consolidation of small MQ servers
- POC environment became production
  - Used original DASD setup (FICON)
  - Original LPAR topology
- Design point: all guests must be able to run on one CEC
- After MQ we moved Communications Servers
- New requests for new functions/applications
Moving Forward – Staffing

- Acquired a dedicated zVM FTE (repurposing)
- Linux support comes from AIX team
  - 1 FTE spends most of his time on Linux
- 1 FTE working performance & Omegamon
  - Jack of all trades, zVM, Linux, Omegamon & tuning and capacity planning, works where needed
- zOS capacity planner also doing zVM Linux capacity planning and trouble shooting
- All are learning as we go
Where We Are Today

- 2 z196 CECs with 2 IFLs each
  - 6 LPARS – 2 prod, 2 sandbox, 2 upgrading to 6.2 & SSI
  - Upgrading to 2 EC12s with 2 IFLS each in late February

Current Linux guests
- 36 MQ – 22 prod, 14 test
- 22 WSRR – 18 prod, 4 test
- 6 WODM – 4 prod, 2 test
- 9 Communications Server – 6 prod, 3 test
- 2 Network Monitors – 1 prod, 1 test
- 6 Tivoli products
- 1 ILMT
- 1 Security Blanket
- 2 Oracle (playground)

- There are about 40 zVM support virtual machines per zVM LPAR
Additional Linux Guests for 2013

- Waiting for the EC12s
  - 2 WODM (Prod)
  - 2 Tivoli Access Manager
  - 2 HOD
- Held back for legal reasons (resolved)
  - 10 MQ
  - 4 Communications Server
- Big Unknown
Issues and What We’ve Learned

- zVM and Linux on z has been a bumpy ride
- Lack of in-house knowledge
- Poor planning
- Lack of support
  - Lack of zVM/Linux network – few friends to call for help
  - Geography
  - Unable to join zVM listserv at Marist
  - Limited opportunity to send people to SHARE & other conferences
- Management issues
We’ve had trouble getting good information about how much memory a guest should have.

What ratio of virtual to real should be used for memory:
- Currently using 1.5 virtual to real
- Sum of guest RAM & V-disk = virtual

Have many guests with too much memory:
- All MQ guests have 1024 meg Ram & 512 meg V-disk
- Busiest MQ guest is averaging 35% busy (1 logical IFL)
- It stands to reason all other MQ guests could use less memory

How do you determine how much memory a Linux guest uses/needs?
Design point: all guests must be able to run on one CEC

- Requires the LPARs to be at least twice as big as needed in order to have the memory to run everything

Keeping track of guest memory size and mapping to LPARs is currently a manually intensive effort

I have memory map spreadsheets

- Change/add a guest requires updating the spreadsheet(s)
- **Is there a better way?**

Memory has been tight on my machines

Because 90% of my Linux guests are communications infrastructure, taking them down to add memory to an LPAR has been difficult

- SSI will help with this?
Performance Monitoring

- The installed performance monitor was difficult to use, not intuitive
- In general the metrics and data were there, but...
- 1 person was almost dedicated to making the monitor work
- Very manually intensive to keep running
- Difficult to tailor displays to meet desire/needs
- One person could wipe out another person’s displays/graphs
  - Multiple id’s with admin authority
- Out of sync data, zVM data 5 minute intervals, Linux data 1 minute intervals
One Friday afternoon we had a serious problem, MQ and Comm Server traffic at a crawl, all IFLs 100% busy, every Linux guest 100% busy and bogged down

- We couldn’t find anything useful in the performance monitor
- We couldn’t log onto a guest to use native tools
- It took 3 hours to find the problem, it was by accident

A couple of people finally learned enough to use the monitor making future problems less painful, if they were available

With our last CA negotiation we acquired a new zVM/Linux monitor

- So far we are much happier, even though, it too has quirks and issues

With knowledge and experience we are getting better at solving performance issues
Guest Creation/Provisioning

- Creating a guest is still a manual process, no automated provisioning
- Takes 1–3 days depending upon the workloads of the people involved
- Provisioning considerations
  - Which LPAR(s) should host the guest?
  - Is there enough memory?
  - Is there enough disk?
  - Guest priority/share?
  - VLAN IP addresses?
- It appears that application software installation can be a CPU intensive operation
  - Software installs and upgrades during prime shift have caused high CPU utilization and MQ performance problems
  - Guest priority and share has helped with this
Capacity Planning

We’ve been trending IFL busy by LPAR from RMF data in the MICS PDB

We finally got Monwrite data FTP’d to z/OS daily for MXG

Too many MXG files and too little time to spend diving into it effectively
  ◦ Lack of SAS expertise with everyone working on zVM & Linux except 1 person

A handful of queries to MXG–L helped find guest utilization and memory allocation

Started writing SAS code to scan Dirmaint disk maps to figure out disk space usage
  ◦ Is there a better way?

In our last CA negotiation MICS for zVM/Linux was acquired
  ◦ Working on making monitor data available in MICS
Disk

- So far all zVM and Linux disk has been standard FICON and count key data format (z/OS)
  - Benefit: replicated to our hot site
- The storage group says this is wasting a lot of space in the DS8800s
  - Similar to zFS in z/OS
- Until recently, we had no spare channels to define as Fiber Channel to connect to our distributed Disk systems
  - We will play around with some Fiber Channel disk especially for the Oracle playground
- Long term, we will probably be a mixed environment
  - FICON for zVM and Linux executables, etc.
  - Fiber channel for large data storage (Oracle)
We reached a point where we needed to add a 3rd IFL to both CECs

The software upgrade cost from 4 to 6 IFLs for some software was a killer

IBM countered with a Sub-Capacity Licensing Agreement
  ◦ Software is charged based upon the least (smallest) number of the following:
    ▪ Number/sum of real IFLs running the software
    ▪ Number/sum of logical IFLs assigned to zVM LPARs running the software
    ▪ Number/sum of virtual IFLs assigned to Linux guests running the software
  ◦ This number/sum is called cores

We needed to re-architect our z/VM topology to take advantage of Sub-Capacity Licensing
Bad Linux Apps

- We acquired a business app that was originally developed for Windows and a 3rd party MQ equivalent.
- We insisted it run on Linux on z (where our MQ is at) and that it use MQ.
- The resultant app was a real pig.
  - Used 1 full IFL whether processing data or waiting for data.
  - Spin loops rather than stimers.
- This got us looking at and implementing guest share and priority options in an attempt to limit its impact.
- Almost forced us to the 3rd IFL per CEC.
- Moved the app to Windows.
- Eliminated the need for the 3rd IFL.
The Sub-Capacity License Agreement forced an LPAR re-architecture.

In reality the ‘test’ guests have the same operational characteristics as productions guests they just access test applications:
- Merge these guests into the same LPARs as the production guests.

Minimize the number physical and/or logical IFLs used by any given piece of software:
- MQ and Comm server will run in one pair of zVM LPARs.
- All other software will run in another pair of LPARs.
- If we have other future software with licensing cost issues we will create LPARs for it or place it in the MQ/Comm server LPARs if it will fit.

The 2 ‘test’ LPARs were eliminated, to be replaced with 2 new production LPARs.
zVM 6.2 and SSI

- About the same time the Sub Capacity License Agreement came into play we started working on zVM 6.2 and SSI
- The LPAR Re-Architecture has been delayed until zVM 6.2 and SSI is installed
- The 2 sandbox LPARs have 6.2 and SSI
- 2 new LPARs have 6.2 and SSI and are almost ready for production
- We discovered the Linux guests must be SLES 11 at SP2 for SSI to work (dynamic relocation)
- Rexx execs have been developed to make moving guests (relocation) from LPAR to LPAR much easier
Very Near Future

February 2012
- Upgrade both CECs to EC12s

March 2012
- Migrate all guests from current LPARs to the new 6.2 SSI LPARs
- Upgrade the old LPARs to 6.2 SSI
- Move the MQ and Comm Server guests to the original LPARs
- Start converting Linux guests to SLES 11 SP2
  - Currently 3–4 hours per guest
  - Is there an easy way to do this?
Intermediate Future

- Implement the Linux guests planned for 2013
- Continue playing with Oracle
- Connect some fiber channel disk (for Oracle)
- We acquired the CA zVM suite of products
  - Review and exploit the products that make sense
    - ACF2
    - DASD Backup
    - Improve provisioning
    - Automation
- Look at what else can move to zVM/Linux
Conclusions/Results

- zVM and Linux on z is here to stay
- We’ve grown beyond the original MQ migration with no additional cost except
  - The application software
  - Monitor replacement
  - The CA Suite
- It's been a wild ride
- The IFLs (2/CEC) are currently in the 50–60% busy range during prime shift
- We’ve still got a lot to do make the platform meet Humana business standards/requirements
- We are learning something new everyday
Questions?
Thanks!

- Don Dunaway, Humana – zVM & Linux ‘jack-of-all trades’ and monitors
- Wendell Miller, Humana – zVM
- Bill Head, Humana – Linux on z