Right-Fitting Applications into Multi-Tier Hybrid Server Environments

Montgomery Bauman
mbauman@us.ibm.com

Thursday, March 3, 2011: 8:00 AM-9:00 AM
Room 201A (Anaheim Convention Center)
Session Number 8620
MVS Core Technologies Project
**Session Abstract**

- **Right-Fitting Applications into Multi-Tier Hybrid Server Environments**
- Thursday, March 3, 2011: 8:00 AM-9:00 AM
- Room 201A (Anaheim Convention Center)
- Speaker: Montgomery Bauman (IBM Corporation)
- A lot is being said about "Workload Optimization" these days. The notion at hand is ages old. Placing the right workloads onto the right servers (or virtual servers) so as to optimize IT value (i.e. reducing cost, maximizing time to value, and minimizing risk).
- The new IBM zEnterprise server offers not just a new bigger better faster mainframe (the z196), but a new approach to server design and systems construction. Whether you view zEnterprise as a "system of systems", or as a "hybrid server", the zEnterprise approach is a novel and potentially powerful new tool to be used to advance the cause of "Workload Optimization" and to derive the benefits therefrom.
- In this session, we will seek to understand zEnterprise as a platform for "Workload Optimization". We will also explore a methodology and toolset that facilitates the comparative analysis of zEnterprise vs traditional servers. The comparative analysis covers both technical and financial attributes of traditional multi-tier heterogeneous server configurations compared with hybrid-server configurations on zEnterprise servers.
- Past Share talks have been done for "Right-Fitting Applications into Consolidated Environments (RACE)". This talk expands the mission and scope of the RACE tools to explore not just virtualization cost & value analysis, but to also explore right fit analysis leading to overall "workload optimized" solutions.... solutions that cost less, improve time to value, and reduce risk.
Topics

1. Workload Optimization - Defined
2. Fit for Purpose - Requirements Fitness
3. Total Cost of Ownership - Financial Fitness

• Appendix
Workload Optimization - Defined
Workload Optimization - Defined

- Run the right **code**…
  - (and serve the right data)
  - (aka applications)
  - (aka workloads)

- On the right **platform(s)**…

- To satisfy (or exceed) **requirements**…

- And minimize **costs**!!!
Requirements

- Perform a function
- Deliver the function on time
- Deliver the function within budget
- Perform that function where needed
- Perform that function when needed
  - planned up time (and planned down time)
  - unplanned down time
- Perform that function how needed
  - in a manner the user finds productive (ease of use)
  - in a manner the user finds productive (response time)
- Make changes to the function over time
- Protect the function from illicit or illegal access or use
Costs

- **Building Costs**
  - The cost of building (or buying) & implementing components
  - The cost of building (or buying) & implementing containers
  - The cost of building (or buying) & implementing connectors
  - The cost of building (or buying) & implementing platforms

- **Operating Costs**
  - The cost of operating components, containers, connectors, and platforms
  - Including managing, monitoring, energizing, cleaning, and replenishing consumables

- **Maintaining Costs**
  - The cost of changing components, containers, connectors, and platforms
  - Including building changes, testing changes, and implementing changes
IT Services Composition

- **Components**
  - Code
  - Data
  - Rules/Scripts/Config Files/etc.
- **Containers**
  - Operating Systems
  - Middleware
  - Processes/Address Spaces
  - The “API” Provider
  - etc.
- **Connections**
  - TCP/IP RPC
  - TCP/IP DRDA
  - TCP/IP MQ
  - etc.
- **Platform**
  - Discrete Server (& Operating System)
  - Virtual Machine (& Operating System)
    - And Hypervisor
    - And Hosting server
## IT Service Delivery Lifecycle:
### (and Component Development Lifecycle)

<table>
<thead>
<tr>
<th>Steps</th>
<th>Business Model</th>
<th>Architect</th>
<th>Code and Data</th>
<th>Engineer: Container Connector Platform</th>
<th>Test Plans</th>
<th>Operations Support Procedures</th>
<th>Operations Call Center</th>
</tr>
</thead>
<tbody>
<tr>
<td>Build</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Test</td>
<td></td>
<td></td>
<td>5</td>
<td>5</td>
<td></td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>Run</td>
<td>6</td>
<td></td>
<td>6</td>
<td>6</td>
<td></td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Maintain</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
</tr>
</tbody>
</table>
1. Build Components

- Business Application Owner
- Business Analyst (Modeler)
- Developer

Development
- Code
- Code
- Data

Topics
2. Build Containers and Connectors

- Business Application Owner
- Business Analyst (Modeler)
- Developer
- Application Architect
- System Architect

Diagram:
- Development
  - Code Container
  - Connector
  - Code Container
  - Connector
  - Data Container
3. Build Platforms

- Business Application Owner
- Business Analyst (Modeler)
- Developer
- Application Architect
- System Architect
- Platform Engineer

Development Platform
- Code Container
- Connector

Platform
- Code Container
- Connector

Platform
- Data Container
- Connector
4. Build Test Environment
5. Build Production Environment

- Development
  - Platform
  - Code Container
  - Connector
  - Platform
  - Data Container
  - Connector
  - Development Manager

- Test
  - Platform
  - Code Container
  - Connector
  - Platform
  - Data Container
  - Connector
  - Tester

- Production
  - Platform
  - Code Container
  - Connector
  - Platform
  - Data Container
  - Connector
  - Operator
  - Users
  - Production Control

Topics:
- Business Application Owner
- Business Analyst (Modeler)
- Developer
- Application Architect
- System Architect
- Platform Engineer
- Development Manager
- Tester
- Users
- Operator
- Security
- Audit
- Call Center
- Business Application Owner
- Business Application Owner
- Business Application Owner
- Business Application Owner
- Business Application Owner
Viewpoints

- Users - The Business
  - Business Process Owner / Application Owner / End User
  - Finance
- Architects
  - Patterns / Standards / Connectors
- Developers
  - Components / Languages / Data / Data Structures
- Engineers - Platform and Infrastructure Engineers
  - Container Providers
    - Middleware and Operating Systems
  - Platform Providers
    - (Virtual) Servers and (Virtual) Storage
  - Connector Providers
    - (Virtual) Networks and Middleware
- Testers
  - Functional Test
  - Usability Test
  - Acceptance Test
  - Load Test
  - Quality Assurance Test
- Operators - Operations
  - Operators / Help Desk / Automation
  - Production Control
  - Problem / Performance Management
  - Change / Configuration Management
- Auditors
  - Logs
  - Security
  - Data and privacy protection

Each “viewpoint” defines “OPTIMAL” in their own terms
So What is Optimal?

• Balance of all viewpoints
  • Requirements-based decision-making
  • Consensus building
  • Balance of all cost points
  • Transparent requirements-based tradeoffs

• And decide:
  • How to build **components**
  • What **containers** to use
  • What **connectors** to use
  • **What platform to use**
    • For each container
    • For each lifecycle stage
An Approach

- Technology Exploration / Architecture Assessment
  - Solution Design Options
  - Options Generation & Understanding
  - e.g. zEnterprise Whiteboard Right-Fit Workshop
  - e.g. IBM Integration Architecture Workshop

- Fit for Purpose Assessment
  - Requirements-based Options Analysis
  - Creation of a Requirements-Fitness Scorecard
  - e.g. IBM Fit for Purpose Workshop

- Total Cost of Ownership Assessment
  - TCO-based Options Analysis
  - Creation of a Financial-Fitness Scorecard
  - e.g. IBM RACE Workshop

Topics
Architecture Analysis

• What’s been done in the past?
  • Local architectural patterns
  • Industry architectural patterns

• Which edge do you seek?
  • Leading edge? (innovator / adopt and adapt new patterns)
  • Bleeding edge? (inventor / make your own patterns)
  • Trialing edge? (adopter / re-use tried and true patterns)
Architecture Analysis - Options

- Define the option:
  - Components
  - Containers
  - Connectors
  - Platforms
  - Lifecycle Support (dev/test/qa/prod environments)

- Limit the options analysis to those that are truly viable
  - Based upon your “patterns”
  - Based upon your “edge”
  - Boiling the ocean is not very productive
Requirements Analysis: The Big Three

- **Time** (the project plan)
- **Risk** (risk plan)
- **Cost** (the project budget)
Requirements Analysis: Factors and Considerations

- System z
- System x
- Power
- Time Horizon
- ISV Support
- Non-functional Requirements
- Geographic Considerations
- Environmental Constraints
- Strategic Direction
- TCO Model
- Skills
- Politics (Local Factors)
- Architecture
- Technology Adoption
- Deployment Model

Requirements

Options

IBM Fit for Purpose Workshop

Requirements Scorecard
Financial Analysis

- Starts with requirements equity
- Depends upon technical equity
- And delivers a TCO scorecard
IBM and IT Optimization Assistance

- Local Patterns
- Other Patterns
- Options
- IBM Integration Architecture Workshop
- IBM Fit for Purpose Workshop
- IBM RACE Workshop (TCO Models)
- Requirements
- TCO Scorecard
- Requirements Scorecard
- IT Optimization Scorecard

Topics
Fit for Purpose Case Study – Requirements Fitness
The New Business Service
(New Smart Meter Customer Application)

- Industry: Electric Utility
  - Smart Grid adopter (Smart Meter provider)
  - Collecting customer meter data on 15 min intervals
- New customer service
  - Home energy usage alerts
  - When energy usage “over budget”, owner gets notified
  - When energy usage “off pattern”, owner gets notified
  - Security use case: “Away From Home”
    - If “away”, and if energy usage pattern is amiss
    - Then owner (and optionally additional parties) get notified
- Web 2.0 UI (lite-browser)
Requirements

- Function: Home and Away-from-home energy-usage alerts
- Time Line: Production offering coincident with stockholders meeting
- Available to all home owners equipped with smart meters
  - From any browser and/or from downloaded mobile app
  - Large (growing) number of users – solution scalability required
- 7x24x365 Uptime
  - High availability runtime with DR (RTO=30min RPO=5min)
- Homeowner userid / pin (customer profile) access control
  - Assured data privacy high priority
- Integration with existing customer management system
  - z/OS – Sysplex – CICSplex – CICS TS - Web Services (or EXCI)
- Integration with existing smart meter readings database
  - z/OS – Sysplex – DB2 for z/OS – Remote SQL (jdbc or similar)
Application Architecture

**New Systems**
- Platform
  - HTML
  - Apache

**Existing Systems**
- Platform
  - Java& JSPs
  - WAS-ND
- Platform
  - Pattern Tables
  - DB2
- Platform
  - Customer Sys
  - COBOL
  - CICS
- Platform
  - Readings Tables
  - DB2

**Topics**
- New Systems
- Existing Systems
- HTML
- Apache
- Java& JSPs
- WAS-ND
- Pattern Tables
- DB2
- Customer Sys
- COBOL
- CICS
- Readings Tables
- DB2

**Platforms**
- DEV
- TEST
- PROD
### Option 1 (x86)

#### New Systems
- **Platform**
  - x86
  - VMware
  - Virt.Mach.
  - Windows

  - HTML
    - Apache

  - Network Connector

- **Platform**
  - x86
  - VMware
  - Virt.Mach.
  - Windows

  - Java & JSPs
    - WAS-ND

  - Network Connector

- **Platform**
  - x86
  - VMware
  - Virt.Mach.
  - Windows

  - Pattern Tables
    - DB2

  - Network Connector

#### Existing Systems
- **Platform**
  - z/OS

- **Customers**
  - Sys
  - COBOL
  - CICS

- **Readings**
  - Tables

- **DB2**
Option 2 (z)

New Systems

- Platform
  - P7
  - PowerVM
  - LPAR
  - AIX

- HTML
- Apache

IEDN Connector

Java& JSPs
- WAS-ND

Existing Systems

Platform
- z/OS

Customer Sys
- COBOL

CICS

PC Connector

Pattern Mart

Pattern Tables
- DB2

Readings Tables

DEV

TEST

PROD

Topics
Options 3 4 5 6 7 8 and …

• There are certainly many more options
  • z/VM zLinux
  • POWER PowerVM AIX
  • zBX POWER AIX
  • zEnterprise z/VM zLinux
  • etc.

• In a “real” Fit for Purpose effort, all viable options would be examined … but care need be taken not to “boil the ocean”

• But for this talk (for this case study) we will keep things simple
  • And just look at Options 1 and 2
Requirements Analysis – Step 1

- Focus on requirements
- Respect for viewpoints
- Seek consensus
  - When consensus not reached
    - Understanding differences and distances
    - Sensitivity analysis (do differences matter?)

- Step 1 - List the requirements
## Requirements List

<table>
<thead>
<tr>
<th>Requirements List</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data privacy</td>
</tr>
<tr>
<td>Scalability</td>
</tr>
<tr>
<td>Integration with existing customer management system</td>
</tr>
<tr>
<td>Development lifecycle support</td>
</tr>
<tr>
<td>Availability</td>
</tr>
<tr>
<td>Performance</td>
</tr>
<tr>
<td>Manageability</td>
</tr>
<tr>
<td>Integration with existing smart meter readings database</td>
</tr>
</tbody>
</table>
Sort and Weight the Requirements

- Step 2
  - Prioritize (i.e. sort) the requirements in the list
  - Assign a weight to each requirement
    - To indicate relative importance of each requirement

<table>
<thead>
<tr>
<th>Rank</th>
<th>Requirements List</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Performance</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>Scalability</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>Integration with existing customer management system</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>Integration with existing smart meter readings database</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>Availability</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>Data privacy</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>Manageability</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>Development lifecycle support</td>
<td>3</td>
</tr>
</tbody>
</table>
Score the Options

- Step 3 – for each requirement, score the options
  - Assess each option's ability to meet the requirement

### Qualifier
- Demonstrated Can Exceed
  - Score: 7
- Likely to Exceed
  - Score: 6
- Demonstrated Can Meet
  - Score: 5
- Likely to Meet
  - Score: 4
- Marginal
  - Score: 2
- Does Not Meet
  - Score: 0

### Requirements List

<table>
<thead>
<tr>
<th>Rank</th>
<th>Requirements List</th>
<th>Weight</th>
<th>Score</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Performance</td>
<td>6</td>
<td>L.Meet</td>
<td>L.Exceed</td>
</tr>
<tr>
<td>2</td>
<td>Scalability</td>
<td>5</td>
<td>L.Meet</td>
<td>L.Exceed</td>
</tr>
<tr>
<td>3</td>
<td>Integration with existing customer management system</td>
<td>4</td>
<td>L.Meet</td>
<td>D.Meet</td>
</tr>
<tr>
<td>4</td>
<td>Integration with existing smart meter readings database</td>
<td>4</td>
<td>L.Meet</td>
<td>D.Meet</td>
</tr>
<tr>
<td>5</td>
<td>Availability</td>
<td>4</td>
<td>Marginal</td>
<td>D.Meet</td>
</tr>
<tr>
<td>6</td>
<td>Data privacy</td>
<td>3</td>
<td>L.Meet</td>
<td>D.Meet</td>
</tr>
<tr>
<td>7</td>
<td>Manageability</td>
<td>3</td>
<td>L.Meet</td>
<td>D.Meet</td>
</tr>
<tr>
<td>8</td>
<td>Development lifecycle support</td>
<td>3</td>
<td>D.Meet</td>
<td>L.Meet</td>
</tr>
</tbody>
</table>
Rate the Options

- Step 4
  - Multiply the requirement’s weight by the option’s score
  - Add up the results

<table>
<thead>
<tr>
<th>Rank</th>
<th>Requirements List</th>
<th>Weight</th>
<th>Score</th>
<th>Result</th>
<th>Option 1 (x86)</th>
<th>Option 2 (z)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Performance</td>
<td>6</td>
<td>L.Meet</td>
<td>24</td>
<td>L.Exceed</td>
<td>30</td>
</tr>
<tr>
<td>2</td>
<td>Scalability</td>
<td>5</td>
<td>L.Meet</td>
<td>20</td>
<td>L.Exceed</td>
<td>25</td>
</tr>
<tr>
<td>3</td>
<td>Integration with existing customer management system</td>
<td>4</td>
<td>L.Meet</td>
<td>16</td>
<td>D.Meet</td>
<td>20</td>
</tr>
<tr>
<td>4</td>
<td>Integration with existing smart meter readings database</td>
<td>4</td>
<td>L.Meet</td>
<td>16</td>
<td>D.Meet</td>
<td>20</td>
</tr>
<tr>
<td>5</td>
<td>Availability</td>
<td>4</td>
<td>Marginal</td>
<td>8</td>
<td>D.Meet</td>
<td>20</td>
</tr>
<tr>
<td>6</td>
<td>Data privacy</td>
<td>3</td>
<td>L.Meet</td>
<td>12</td>
<td>D.Meet</td>
<td>15</td>
</tr>
<tr>
<td>7</td>
<td>Manageability</td>
<td>3</td>
<td>L.Meet</td>
<td>12</td>
<td>D.Meet</td>
<td>15</td>
</tr>
<tr>
<td>8</td>
<td>Development lifecycle support</td>
<td>3</td>
<td>D.Meet</td>
<td>15</td>
<td>L.Meet</td>
<td>12</td>
</tr>
</tbody>
</table>

This is a “Requirements Scorecard”
Total Cost of Ownership Case Study
– Financial Fitness
Financial Fitness Analysis / TCO

• Prereq 1 – Technical Equity
  • Each option being analyzed must be sized and configured to meet the same set of requirements

• Prereq 2 – Solution Lifecycle Equity
  • Each option being analyzed must similarly meet the solution’s lifecycle stages and timetable

• Prereq 3 – Financial Equity
  • Each option being analyzed must receive similar financial analysis treatment
Technical Equity

• Equitable containers

• Equitable connectors
  • Capacity
  • Availability

• Equitable platforms
  • Capacity
  • Availability

• Equitable component-lifecycle support
Solution Lifecycle Equity

- Solution investment period
- Production date
- Component development staging
Financial Equity

- Treatment of assets at end of life
- Depreciation treatment
- Trade-In treatment
- Tech refresh treatment
RACEv and RACEzOS

- TCO Cost Modeling Tools

- Spreadsheets
  - RACEv for distributed server (including zLinux) costing
  - RACEzOS for non-zLinux costing (usually z/OS)

- Use RACEv and RACEzOS together to model “hybrid solutions”
  - aka Multi-Tier Multi-Platform configurations
Step 1 – RACEv Subject Servers

- The set of discrete x86 servers
  - Presentation – Prod/Test/Dev
  - Application – Prod/Test/Dev
  - Database – Prod/Test/Dev
Step 2 – RACEv x86 Server Target

- The set of x86 VMware virtual server hosting blade servers
  - Presentation – Prod/Test/Dev
  - Application – Prod/Test/Dev
  - Database – Prod/Test/Dev

New Systems

- Platform: x86 VMware Virt.Mach. Windows
  - HTML: Apache
  - Java&JSPs: WAS-ND
  - Pattern Tables: DB2

Existing Systems

- Platform: z/OS
  - Customer Sys: COBOL CICS
  - Readings Tables: DB2

Network Connectors:
- DEV
- TEST
- PROD
Step 3 – RACEv zEnterprise Target Distributed Elements

- The set of distributed elements in the zEnterprise solution
  - Presentation – Prod/Dev/Test
    - PS701s in the zBX
  - Database – Prod/Dev/Test
    - ISAO blades in the zBX

New Systems
- Platform
  - P7
  - PowerVM
  - LPAR
  - AIX
- HTML
- Apache
- IEDN Connector
- Java&JSPs
- WAS-ND

Existing Systems
- Platform
  - z/OS
- PC Connector
- Customer Sys
  - COBOL
  - CICS
- Pattern Tables
- Pattern Mart
- DB2
- Readings Tables
- PC Connector
- PROD
- TEST
- DEV
Step 4 – RACEz(OS) zEnterprise Elements

- The z/OS Elements of the solution
  - Application Servers
  - WAS for z/OS
  - PROD LPAR
  - TEST/DEV LPAR
- WAS/DB2 co-location
- zAAP on zIIP engines
**Partition Detail Report**

Based on LSPR Data for IBM System z Processors

**Description:** Loaded from Basic Mode Study C:\...ShareCaseStudy.zpr

**Host = 2817-M15/700 with 6 CPs: GP=4 zAAP=1 zIIP=1**

**Active Partitions:** GP=2 zAAP=2 zIIP=2

Capacity basis: 2094-701 @ 602.00 MIPS for a single partition configuration

z196 and z10 processor capacity for z/OS is represented with HiperDispatch turne

<table>
<thead>
<tr>
<th>Include</th>
<th>No.</th>
<th>Type</th>
<th>Name</th>
<th>SCP</th>
<th>Workload</th>
<th>Partition Identification</th>
<th>Partition Configuration</th>
<th>Partition Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>GP</td>
<td>Prod</td>
<td>z/OS-1.11</td>
<td>Average</td>
<td>Mode</td>
<td>LCPs</td>
<td>Weight</td>
</tr>
<tr>
<td>✔</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SHR</td>
<td>4</td>
<td>80</td>
</tr>
<tr>
<td>✔</td>
<td>2</td>
<td>GP</td>
<td>Dev</td>
<td>z/OS-1.11</td>
<td>Average</td>
<td>SHR</td>
<td>2</td>
<td>20</td>
</tr>
<tr>
<td>✔</td>
<td>2</td>
<td>zAAP</td>
<td>Prod</td>
<td>z/OS-1.11</td>
<td>Average</td>
<td>SHR</td>
<td>1</td>
<td>80</td>
</tr>
<tr>
<td>✔</td>
<td>2</td>
<td>zAAP</td>
<td>Dev</td>
<td>z/OS-1.11</td>
<td>Average</td>
<td>SHR</td>
<td>1</td>
<td>80</td>
</tr>
<tr>
<td>✔</td>
<td>2</td>
<td>zIIP</td>
<td>Prod</td>
<td>z/OS-1.11</td>
<td>Average</td>
<td>SHR</td>
<td>1</td>
<td>20</td>
</tr>
<tr>
<td>✔</td>
<td>2</td>
<td>zIIP</td>
<td>Dev</td>
<td>z/OS-1.11</td>
<td>Average</td>
<td>SHR</td>
<td>1</td>
<td>20</td>
</tr>
</tbody>
</table>

**Capacity Summary by Pool**

<table>
<thead>
<tr>
<th>Pool</th>
<th>RCPs</th>
<th>Partitions</th>
<th>LCPs</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>GP</td>
<td>4</td>
<td>2</td>
<td>6</td>
<td>4,557</td>
</tr>
<tr>
<td>zAAP</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1,151</td>
</tr>
<tr>
<td>zIIP</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1,151</td>
</tr>
<tr>
<td>IFC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Totals</td>
<td>6</td>
<td>6</td>
<td>10</td>
<td>6,858</td>
</tr>
</tbody>
</table>

For significant configuration changes, capacity comparisons should be considered to have a +/-5% margin-of-error. Upgrading the processor family is considered a significant configuration change.
Step 6 – Complete the Models
Examine the Results

RACEzOS/RACEv Cummulative Case Comparison

- (0)x86-discrete
- (1)x86-virtual
- zEnterprise
Step 7 – Examine More Results
Step 8 - Iterate

- Refine Inputs
- Add Additional Cases and Solution Configurations
- Sensitivity Analysis

- Assumptions Analysis
  - The cost & value of zEnterprise Unified Resource Manager
  - The value of co-location
  - The cross-server sizing
  - Discounting
  - Admin ratios
  - etc.
  - etc.
  - etc.

In other words….

Have a productive argument! …

Which is what ALL of this is ALL about!!!
And Finally

- Merge the Requirements Analysis...
- And the TCO Analysis

### Rank Requirements List Weight Score Result Score Result
1. Performance 6 L.Meet 24 L.Exceed 30
2. Scalability 5 L.Meet 20 L.Exceed 25
3. Integration with existing customer management system 4 L.Meet 16 D.Meet 20
4. Integration with existing smart meter readings database 4 L.Meet 16 D.Meet 20
4. Availability 4 Marginal 8 D.Meet 20
6. Data privacy 3 L.Meet 12 D.Meet 15
7. Manageability 3 L.Meet 12 D.Meet 15
8. Development lifecycle support 3 D.Meet 15 L.Meet 12

Option 1 (x86) Option 2 (z)

<table>
<thead>
<tr>
<th>Rank</th>
<th>Requirements List</th>
<th>Weight</th>
<th>Score</th>
<th>Result</th>
<th>Score</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Performance</td>
<td>6</td>
<td>L.Meet</td>
<td>24</td>
<td>L.Exceed</td>
<td>30</td>
</tr>
<tr>
<td>2</td>
<td>Scalability</td>
<td>5</td>
<td>L.Meet</td>
<td>20</td>
<td>L.Exceed</td>
<td>25</td>
</tr>
<tr>
<td>3</td>
<td>Integration with existing customer management system</td>
<td>4</td>
<td>L.Meet</td>
<td>16</td>
<td>D.Meet</td>
<td>20</td>
</tr>
<tr>
<td>4</td>
<td>Integration with existing smart meter readings database</td>
<td>4</td>
<td>L.Meet</td>
<td>16</td>
<td>D.Meet</td>
<td>20</td>
</tr>
<tr>
<td>4</td>
<td>Availability</td>
<td>4</td>
<td>Marginal</td>
<td>8</td>
<td>D.Meet</td>
<td>20</td>
</tr>
<tr>
<td>3</td>
<td>Integration with existing customer management system</td>
<td>4</td>
<td>L.Meet</td>
<td>12</td>
<td>D.Meet</td>
<td>15</td>
</tr>
<tr>
<td>3</td>
<td>Integration with existing smart meter readings database</td>
<td>4</td>
<td>L.Meet</td>
<td>12</td>
<td>D.Meet</td>
<td>15</td>
</tr>
<tr>
<td>3</td>
<td>Data privacy</td>
<td>3</td>
<td>D.Meet</td>
<td>15</td>
<td>L.Meet</td>
<td>12</td>
</tr>
</tbody>
</table>

And make an “optimal” decision!
End of Section
Two-Column Slide (Type Size=28)

- Topic A (Type Size=24)
  - Subtopic 1 (Type Size=22)
  - Subtopic 2 (Type Size=22)
  - Subtopic 3 (Type Size=22)
  - Subtopic 4 (Type Size=22)
- Topic B (Type Size=24)

- Topic C (Type Size=24)
  - Subtopic 1 (Type Size=22)
  - Subtopic 2 (Type Size=22)
  - Subtopic 3 (Type Size=22)
    - Sub-subtopic 1 (Type Size=20)
    - Sub-subtopic 2 (Type Size=20)
- Topic D (Type Size=20)
Slide with Table

Topics
End of Section
End of Presentation