Multi-Platform-Inclusive IT Optimization Assessment Methodology

Monte Bauman
IBM Columbus
mbauman@us.ibm.com

Friday, March 16, 2012: 09:30AM - 10:30AM, OMNI Hazelnut
Session Number 10491

Topics

Topics

Abstract

Topics

Abstract

- It is great having choices. Choices are the harvest of competition, driving innovation and price/performance (and we all want that!).
- But making choices in the vast - complex - fast moving technology and business space that is "IT" is difficult and inefficient and all too often ineffective. The results of ill-made choices can have long lasting dire effects including failed projects, blown budgets, delayed timelines, and worse (i.e. careers veering off course).
- In this presentation, our speaker will suggest an approach for making IT decisions that is architectural-based, requirements-oriented, and platform-inclusive. Our speaker's suggested approach has three parts:
- (1) A suggested taxonomy for depicting (i.e. drawing) an application architecture including its code (and data components), the containers (where the code runs), the connectors (communications between containers), and platforms (hardware and hypervisors) ... cognizant of application tiers AND application development lifecycle stages
- (2) A "local-factors"-based requirements analysis of leading architectural options (derived in step (1)) resulting in a requirements scorecard providing a relativistic rating of the architected options. Requirements are solicited from the varied and multiple viewpoints of a project (architects, developers, engineers, operations, etc. ... these "local people" provide the facts and information that are the "local factors").
- (3) A "TCO"-based analysis, creating a technically-sound and requirements-equitable cost of ownership scorecard for the top-rated architecture options (derived from step (2)).
- The result of the approach is the derivation of decision-making artifacts including understandable architectural diagrams, a requirements scorecard, and a cost scorecard, wherein the information required to assess the "best fit" (the best "choice") is brought together in an "inclusive" manner (inclusive of all viable architectural elements, AND inclusive of all local-factor-based requirements).

Topics

End of Section

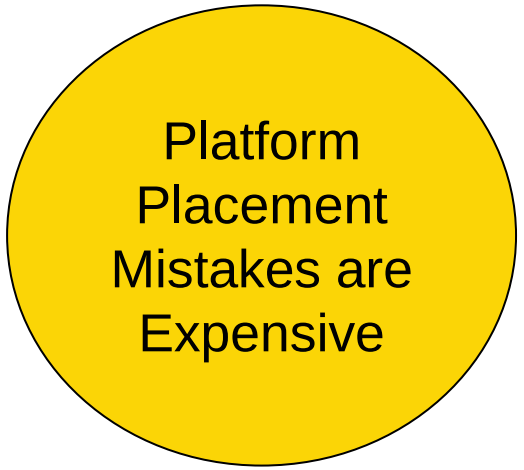
Topics

Multi-Platform-Inclusive IT Optimization Assessment Methodology Overview

Topics

Platform Placement
IT Optimization Challenges

- (Platform Placement) Choices are great to have…
- Robust server marketplace
  - x86
  - System z
  - POWER
  - SPARC
  - etc.
- Emerging “Cloud” marketplace
  - Private (see above)
  - Public
- Middleware everywhere (portable containers)
- Choices are (u-pick) … hard fun time-consuming painful!

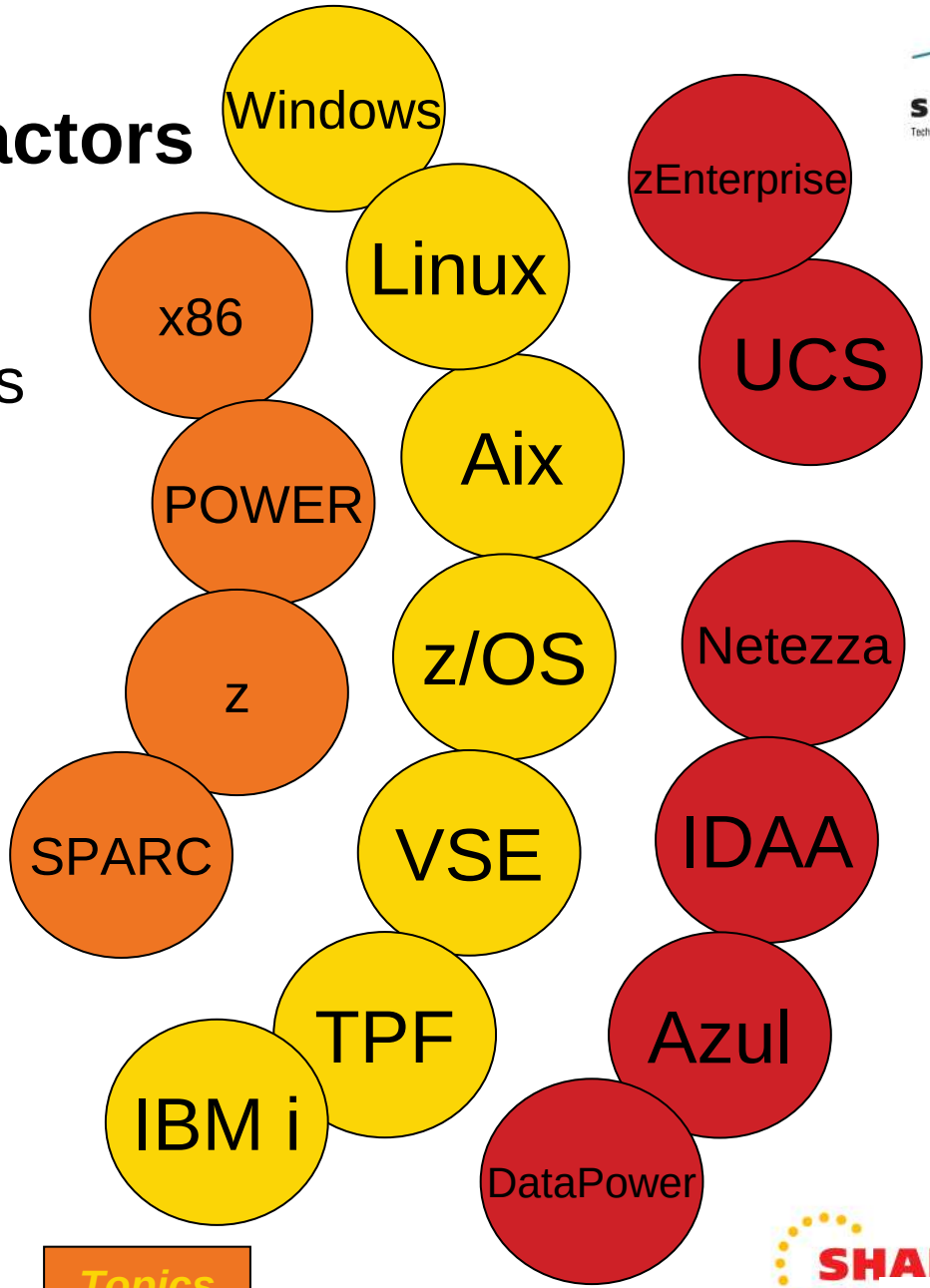


Topics

New Server Form Factors

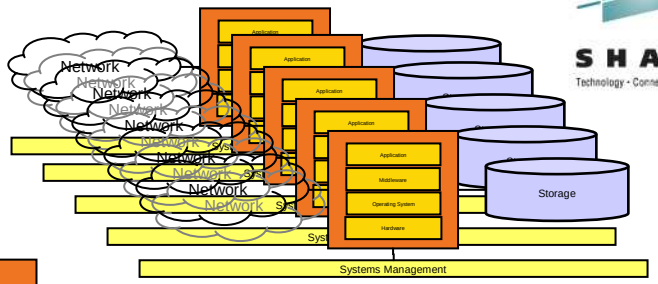
- Pre-Integrated Servers
- Heterogeneous Servers
- Enclosed Clusters
- Single System Images
- Robust Hypervisors
- Appliances

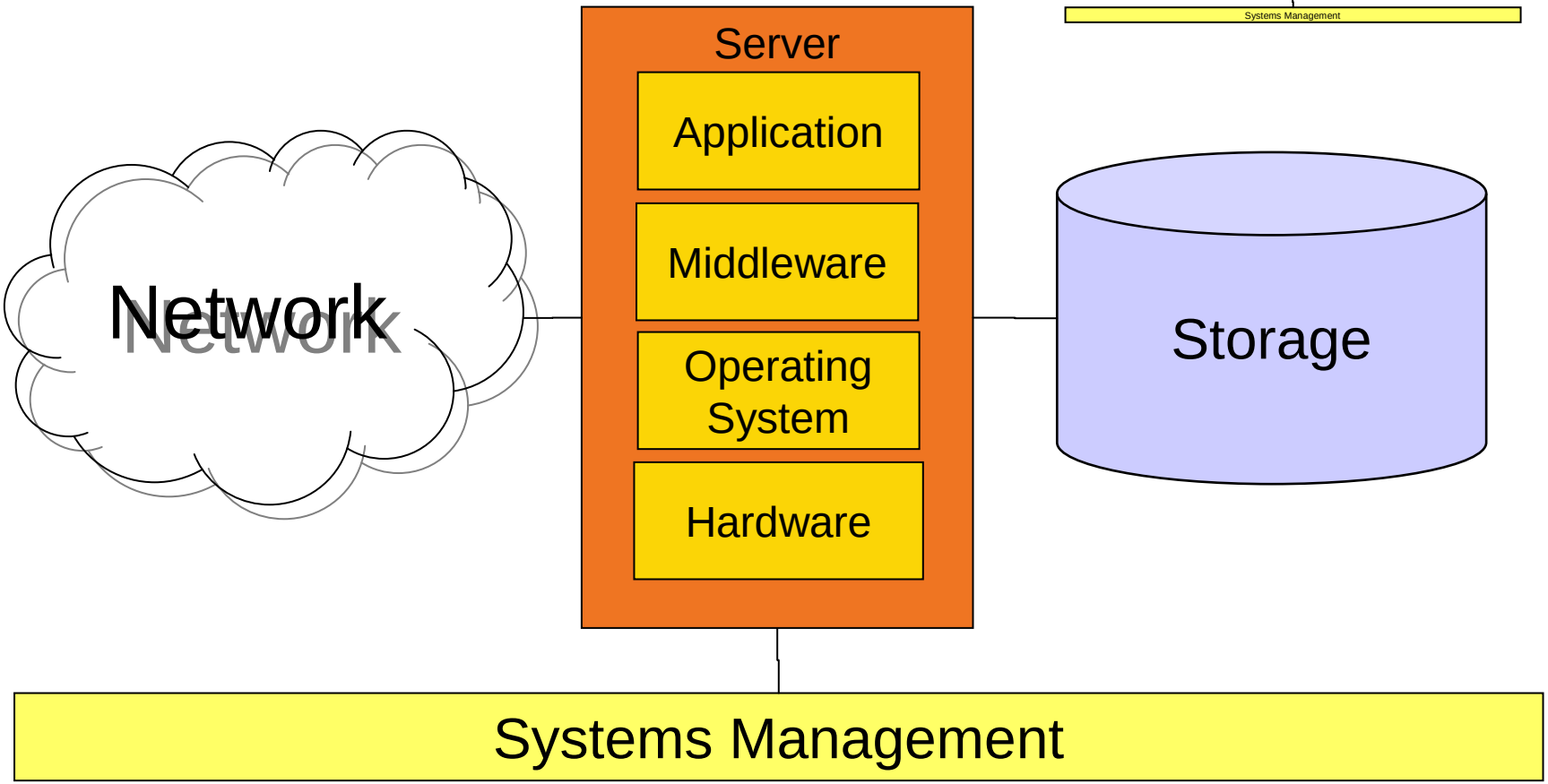
- Examples:
  - zEnterprise
  - Cisco UCS
  - More coming



Topics

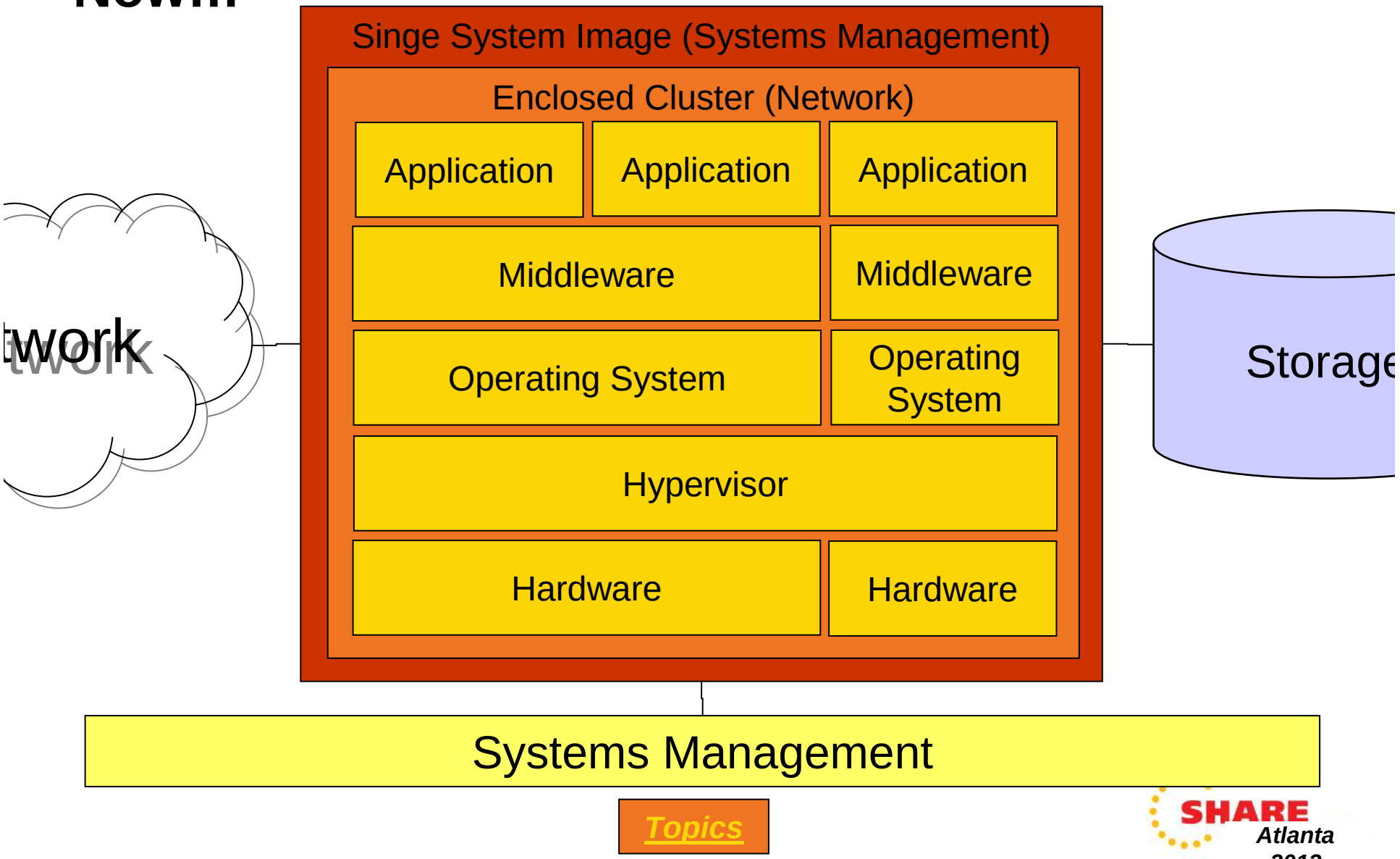
Building a System Then…





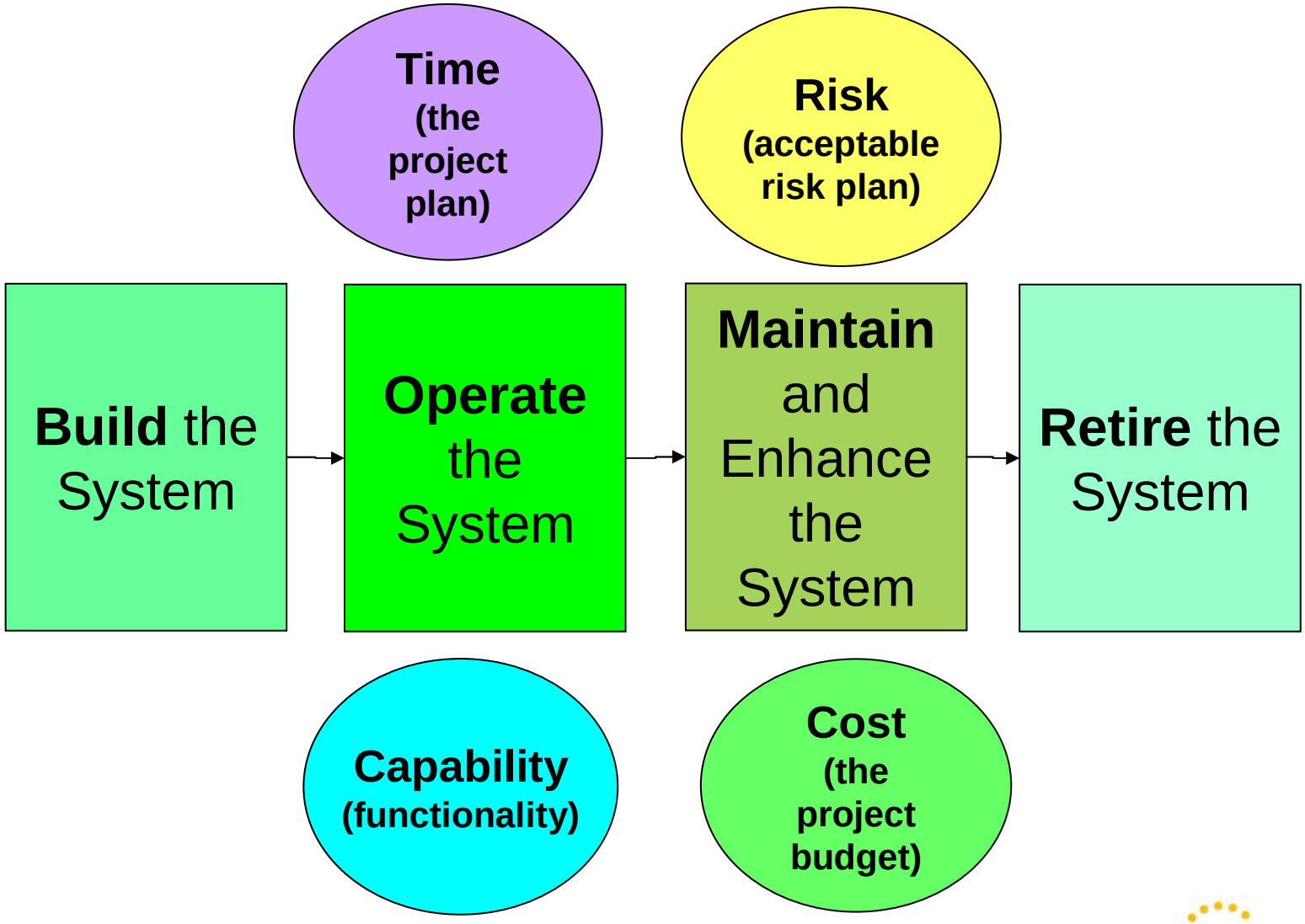
Topics

Building a System Now…



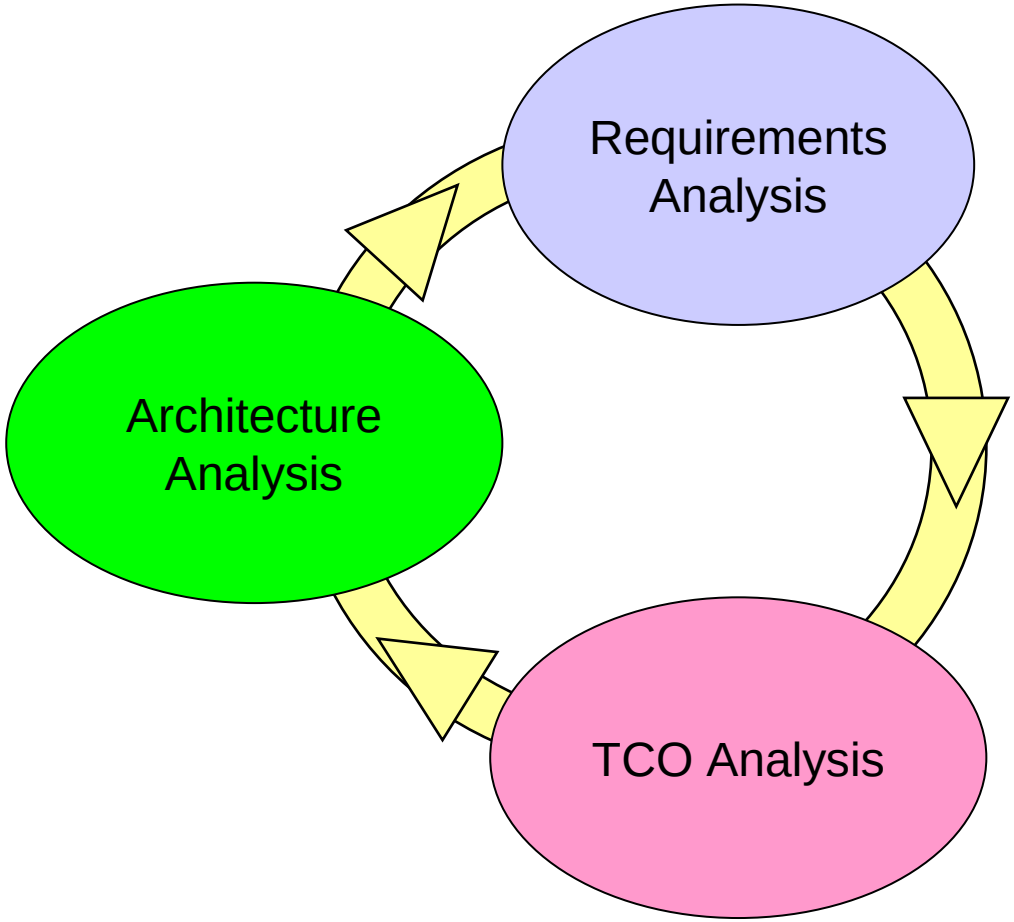
Topics

Evaluating a System



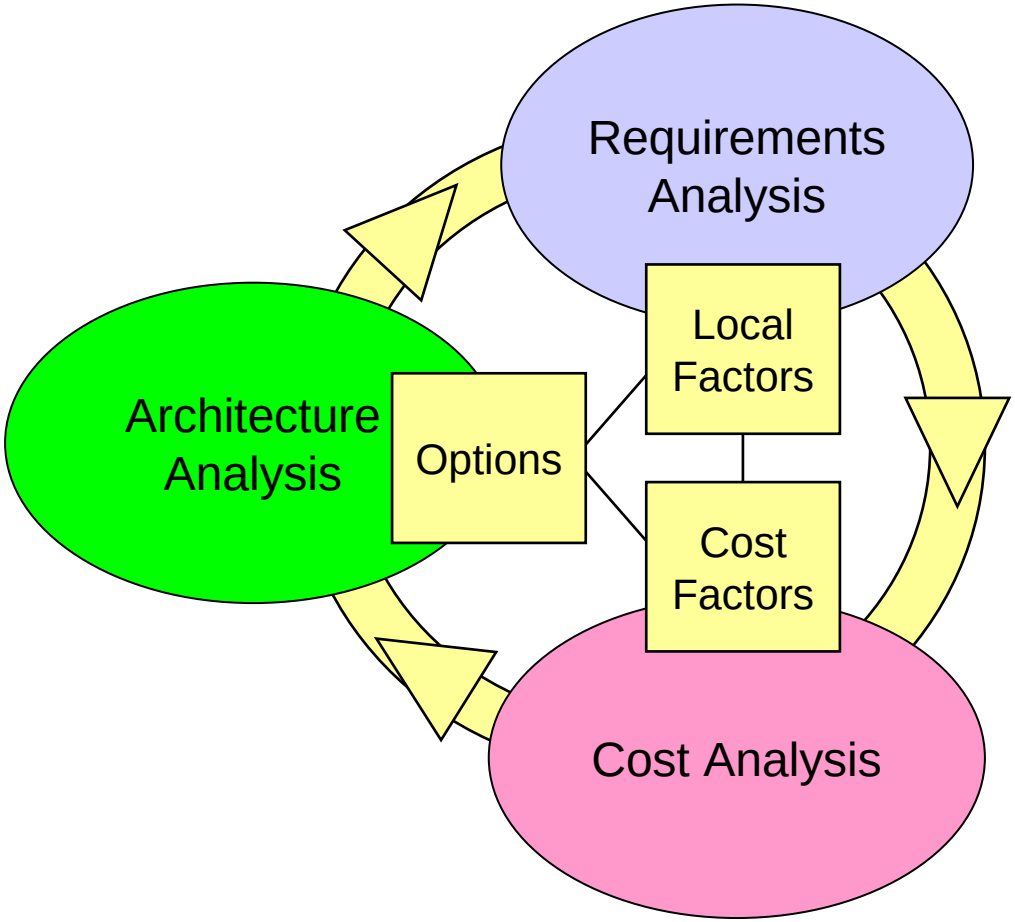
Topics

IT Optimization Assessment Methodology Overview



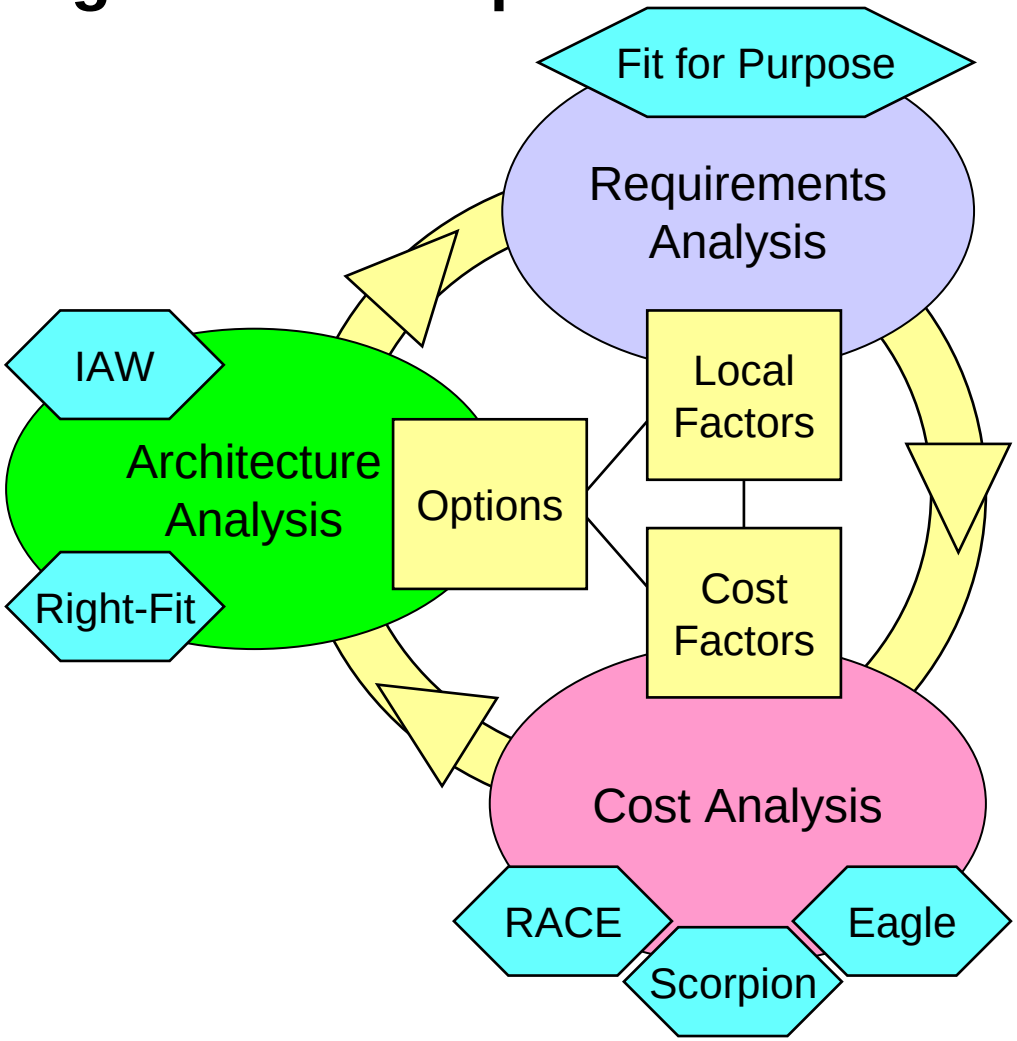
Topics

IT Optimization Assessment Methodology Factors



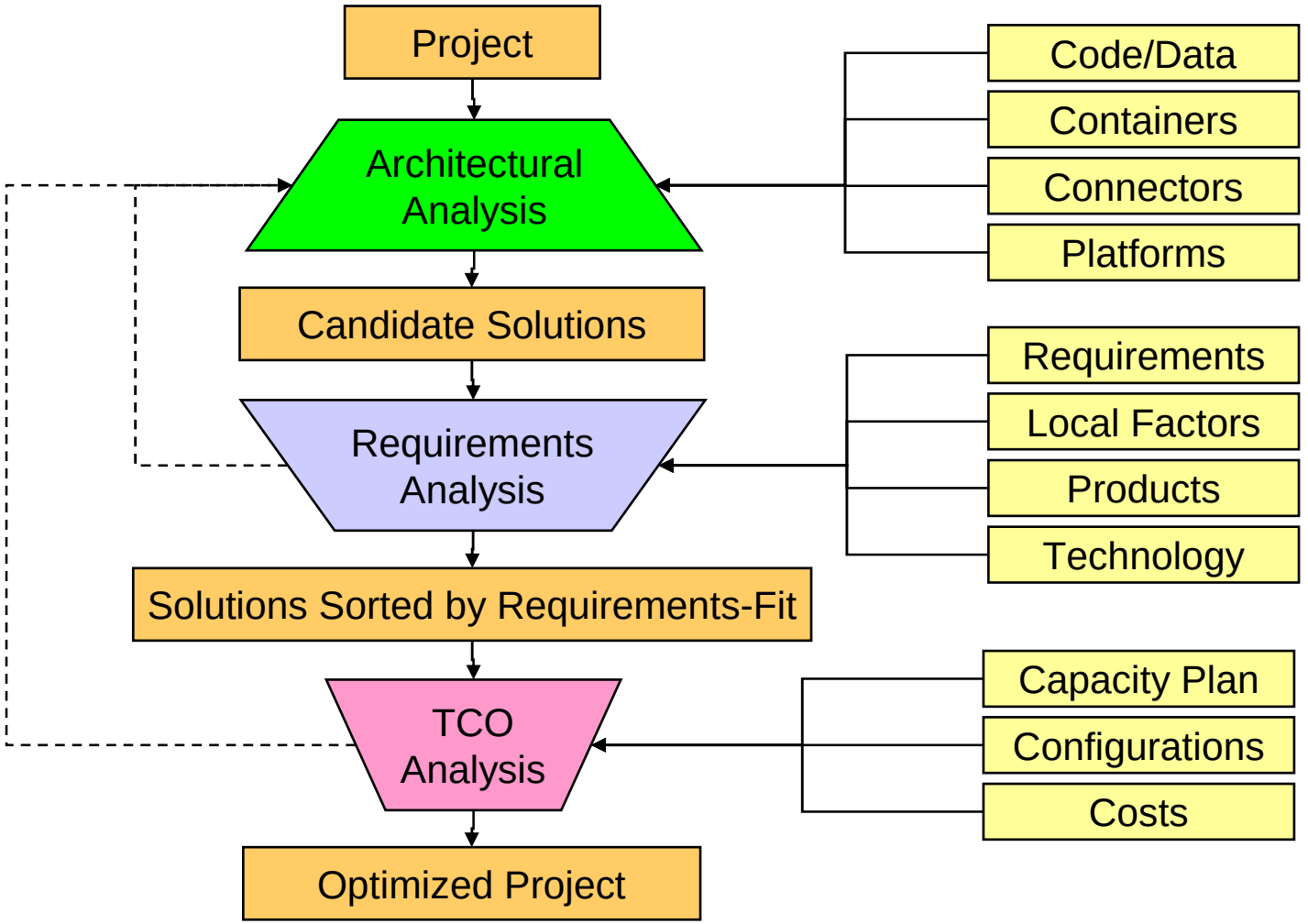
Topics

IT Optimization Assessment Methodology
IBM Offerings / Workshops



Topics

IT Optimization Assessment Methodology Example



Topics

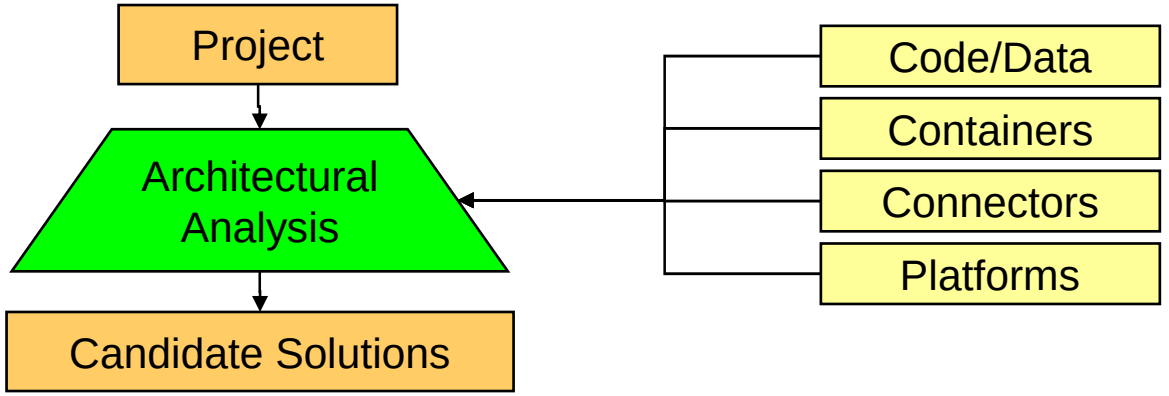
End of Section

Topics

Architectural Analysis

Topics

Architectural Analysis

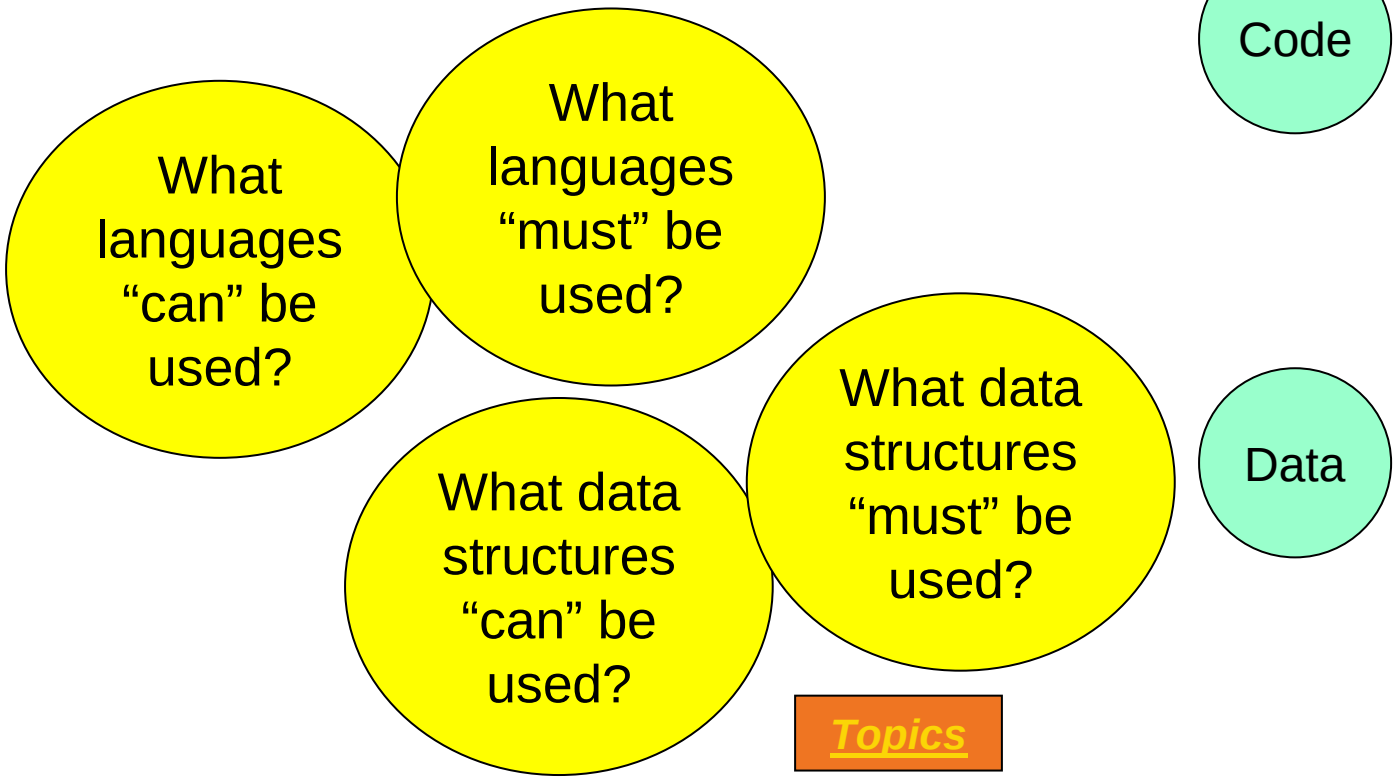


It's About the Options!
Being "Platform Inclusive"

Topics

IT Solution Composition Components

- **Components**
  - Code
  - Data
  - Rules/Scripts/Config Files/etc.



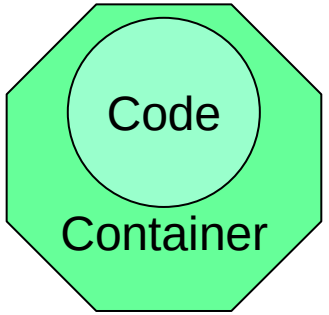
IT Solution Composition Containers

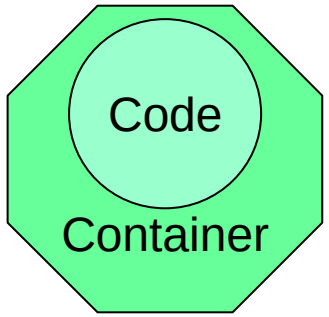
- **Componer**
  - Code
  - Data
  - Rules/S

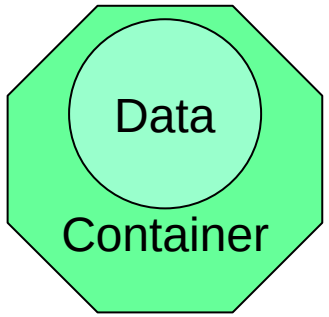
What containers "can" be used?

What containers "must" be used?

- **Containers**
  - Operating Systems
  - Middleware
  - Processes/Address Spaces
  - The "API" Provider
  - etc.







Topics

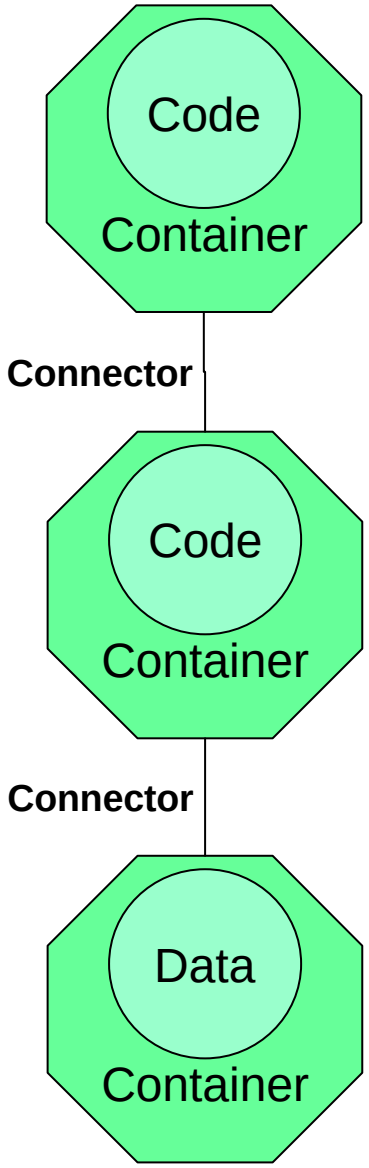
IT Solution Composition Connections

- **Component**
  - Code
  - Data
  - Rules/S
- **Containers**
  - Operating Sy
  - Middleware
  - Processes/Address Spaces
  - The “API” Provider
  - etc.

What connectors “can” be used?

What connectors “must” be used?

- **Connections**
  - TCP/IP RPC
  - TCP/IP DRDA
  - TCP/IP MQ
  - TCP/IP IEDN RPC
  - etc.



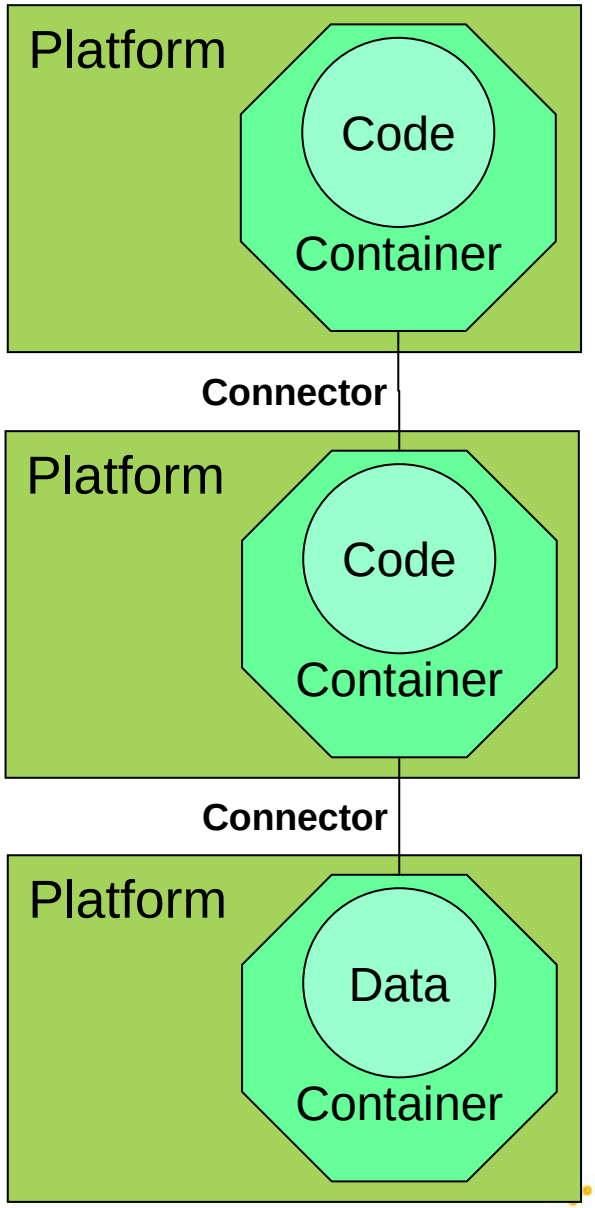
Topics

IT Services Composition Platform

- **Components**
  - Code
  - Dat
  - Files/etc.
- **Co**
  - etc.
- **Connections**
  - TCP/IP RPC
  - TCP/IP DRDA
  - TCP/IP MQ
  - etc.
- **Platform**
  - Discrete Server
    - and Operating System
  - Virtual Machine
    - and Operating System
    - and Hosting Server / Hypervisor

What platforms "can" be used?

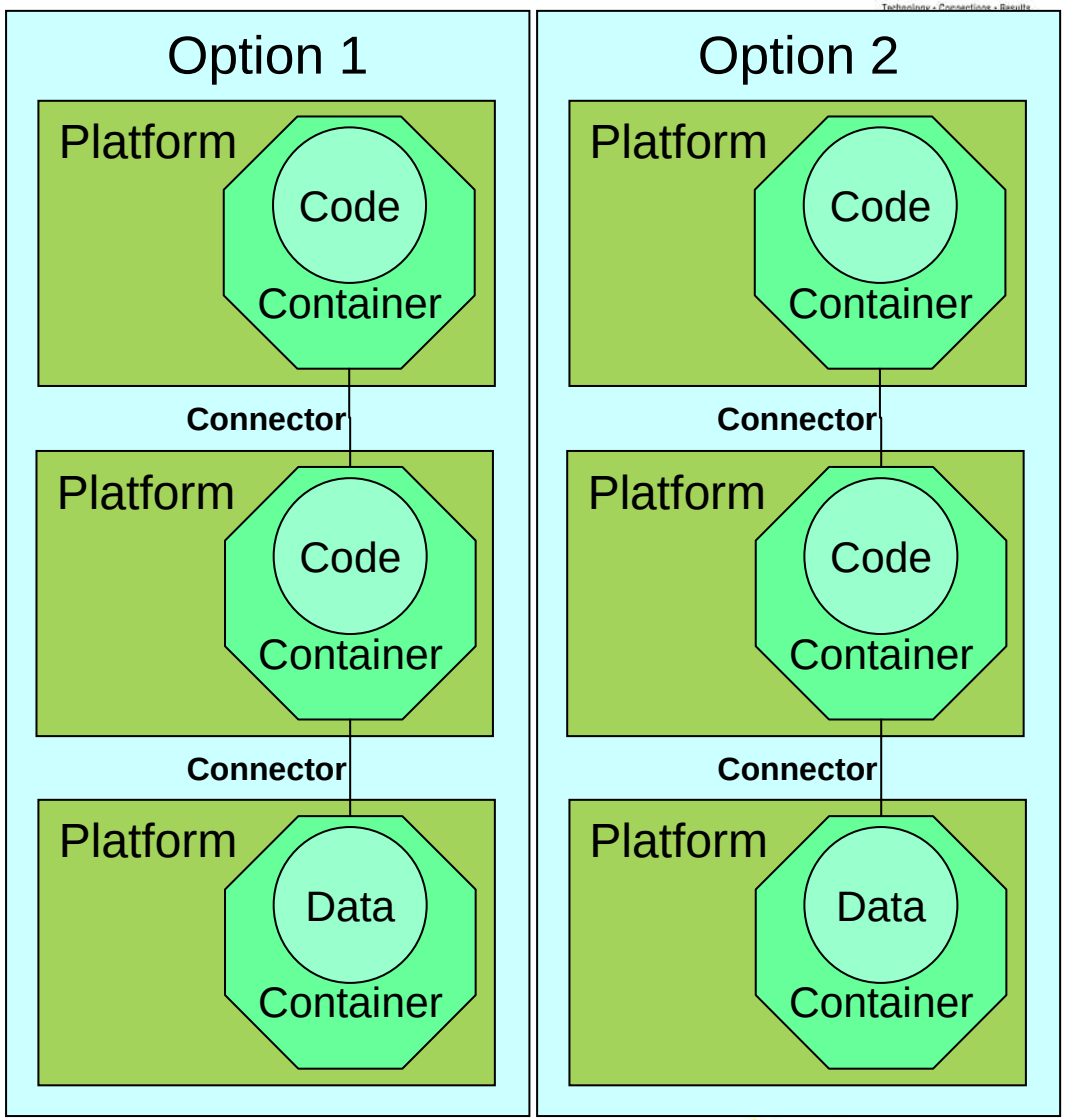
What platforms "must" be used?



Topics

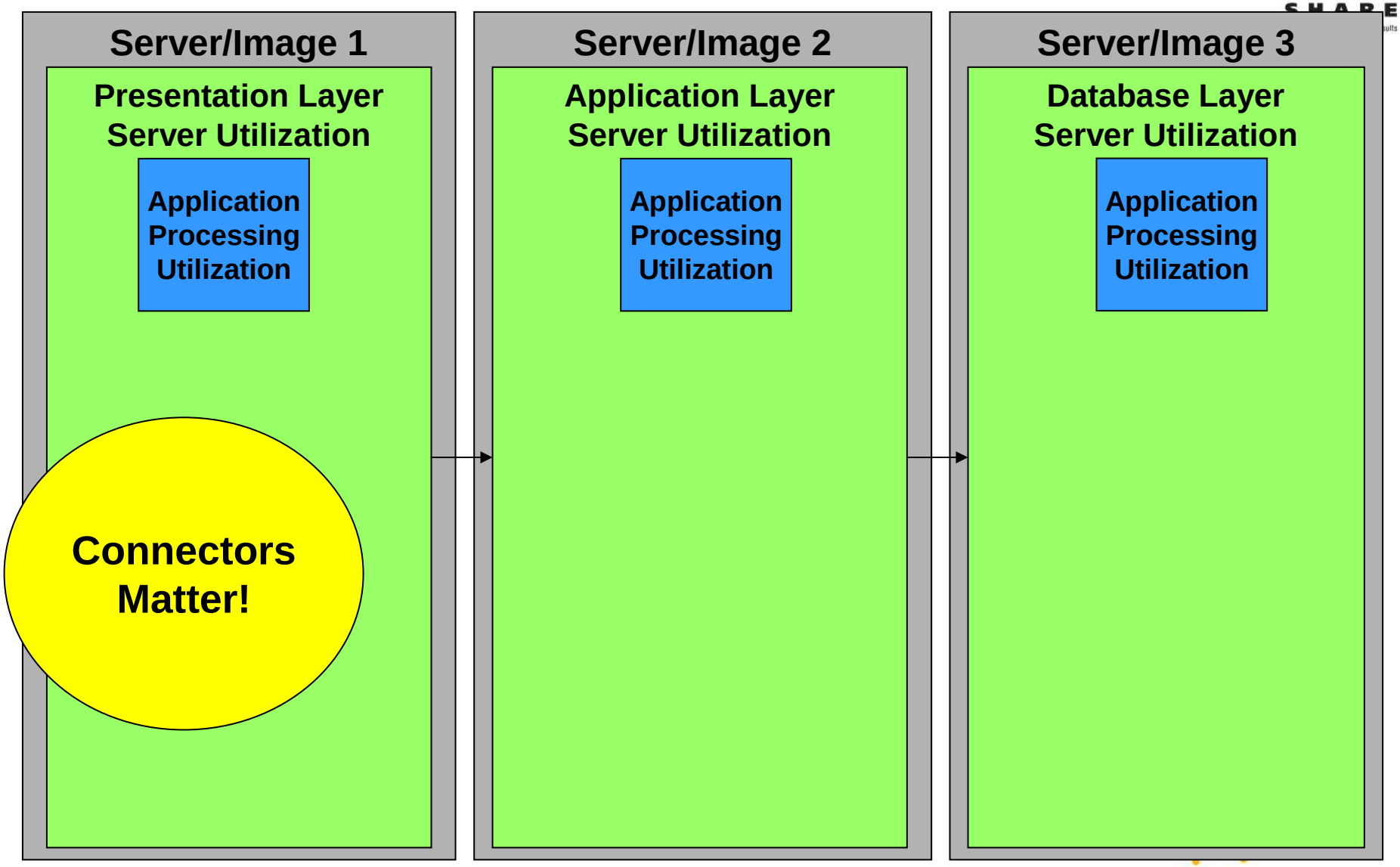
Architectural Options

- **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
    - and Operating System
  - Virtual Machine
    - and Operating System
    - and Hosting Server / Hypervisor



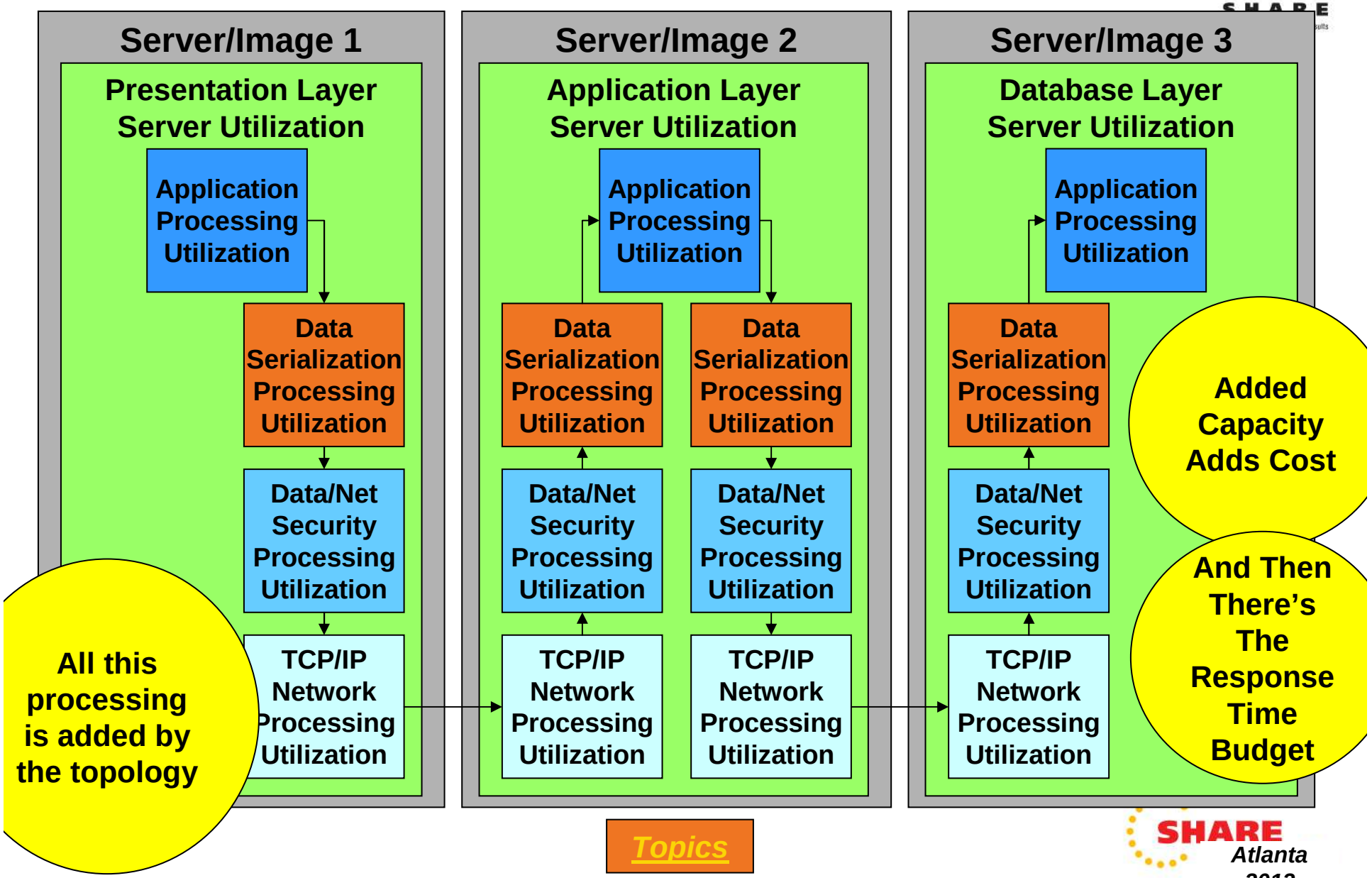
Topics

Topology Variable Overhead

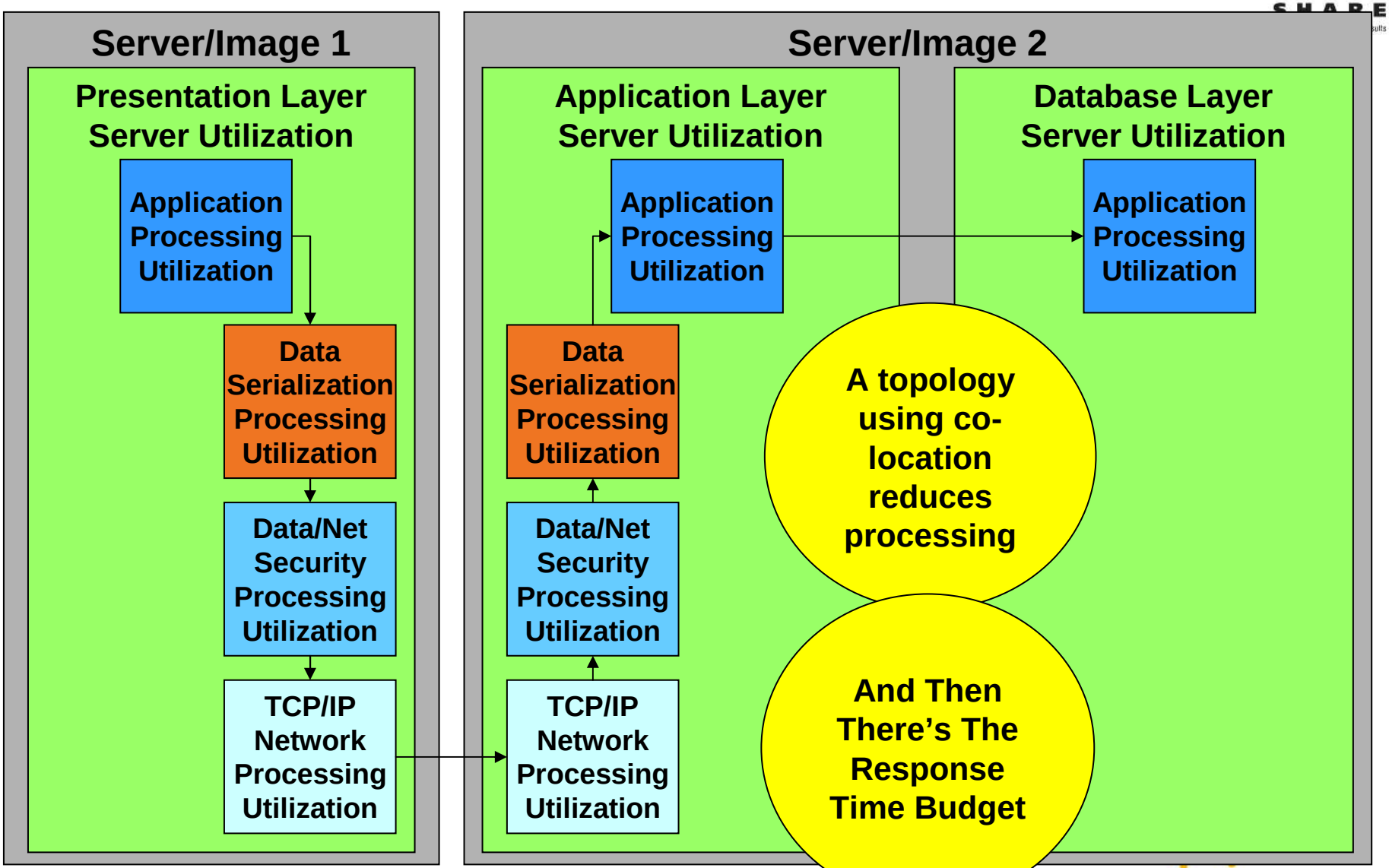


Topics

Topology Variable Overhead - Components



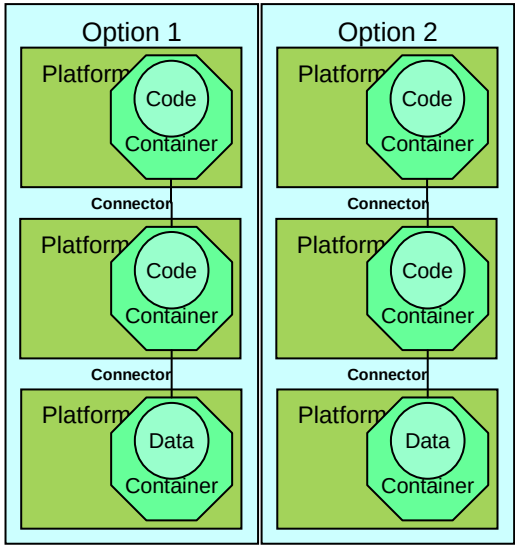
Topology Variable Overhead – Tier Consolidation



Topics

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” (leading / bleeding / trailing)
  - Boiling the ocean is not very productive

- ***Document the Architectural Options (Architectural Artifacts)***

IAW Right-Fit

Topics

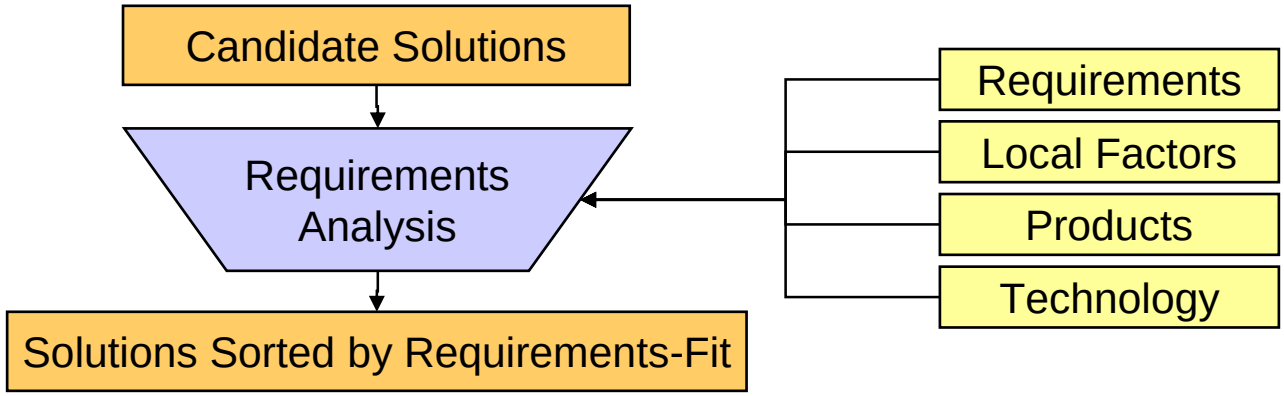
End of Section

Topics

Requirements Analysis

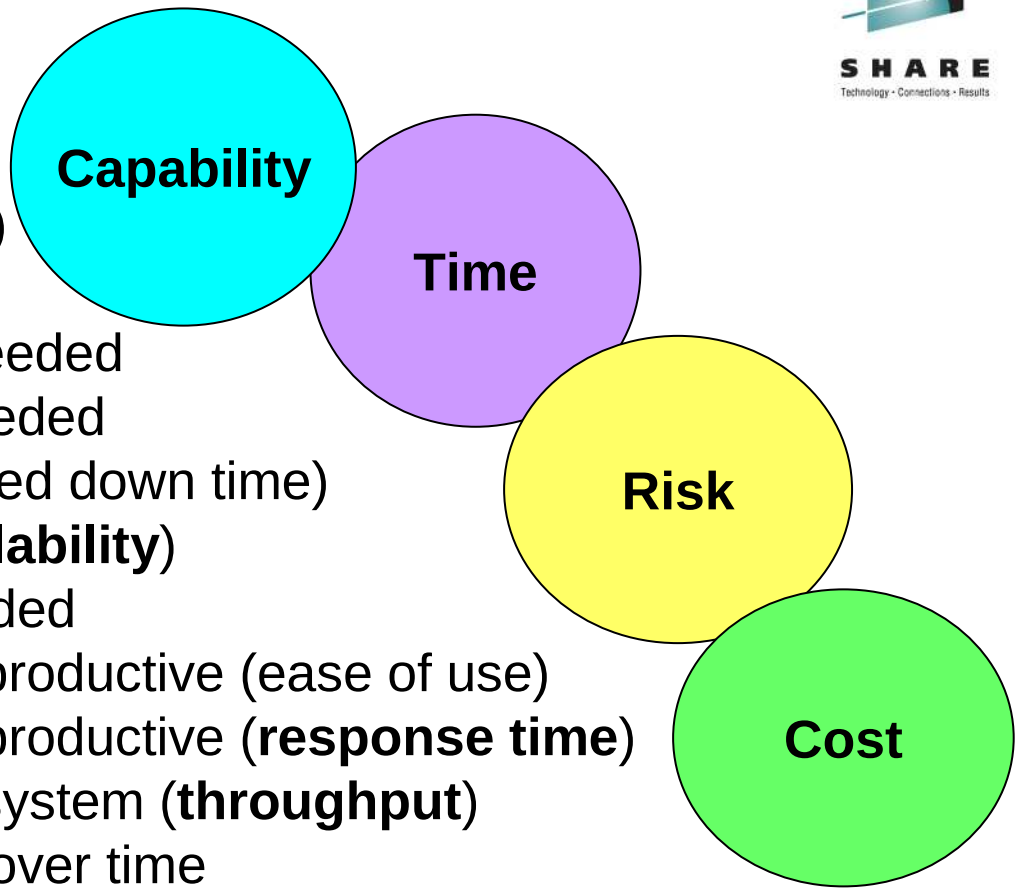
Topics

Requirements Analysis



Topics

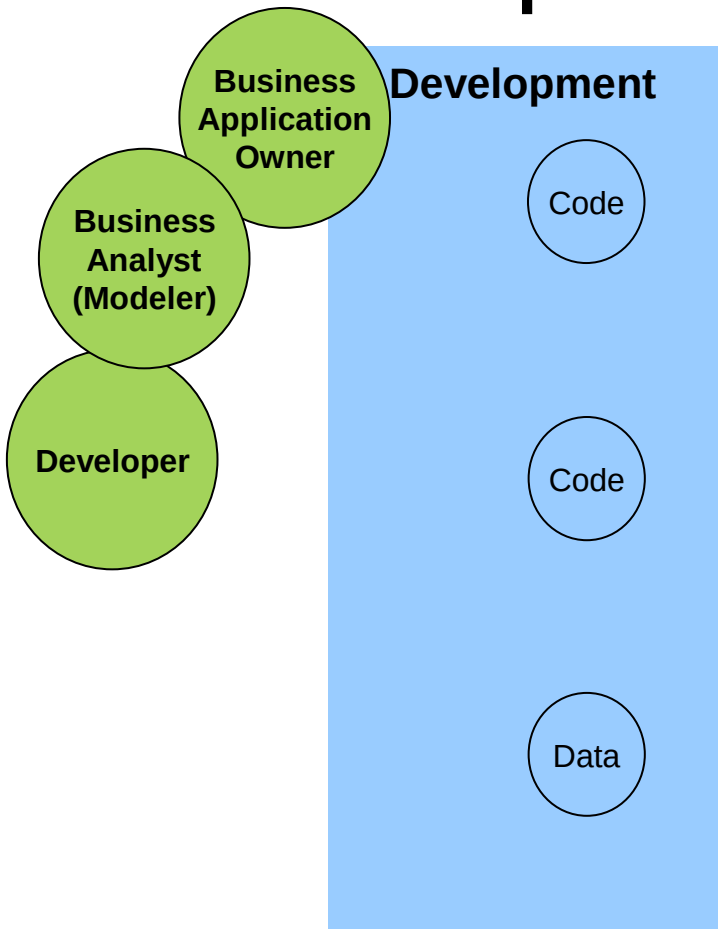
Requirements (examples)



- Perform a function (**capability**)
- Deliver the function on **time**
- Perform that function where needed
- Perform that function when needed
  - planned up time (and planned down time)
  - unplanned down time (**availability**)
- Perform that function how needed
  - in a manner the user finds productive (ease of use)
  - in a manner the user finds productive (**response time**)
  - For all the users using the system (**throughput**)
- Make changes to the function over time
- Protect the function from illicit or illegal access or use (**security**)
- Provide ROI (deliver on **budget**)

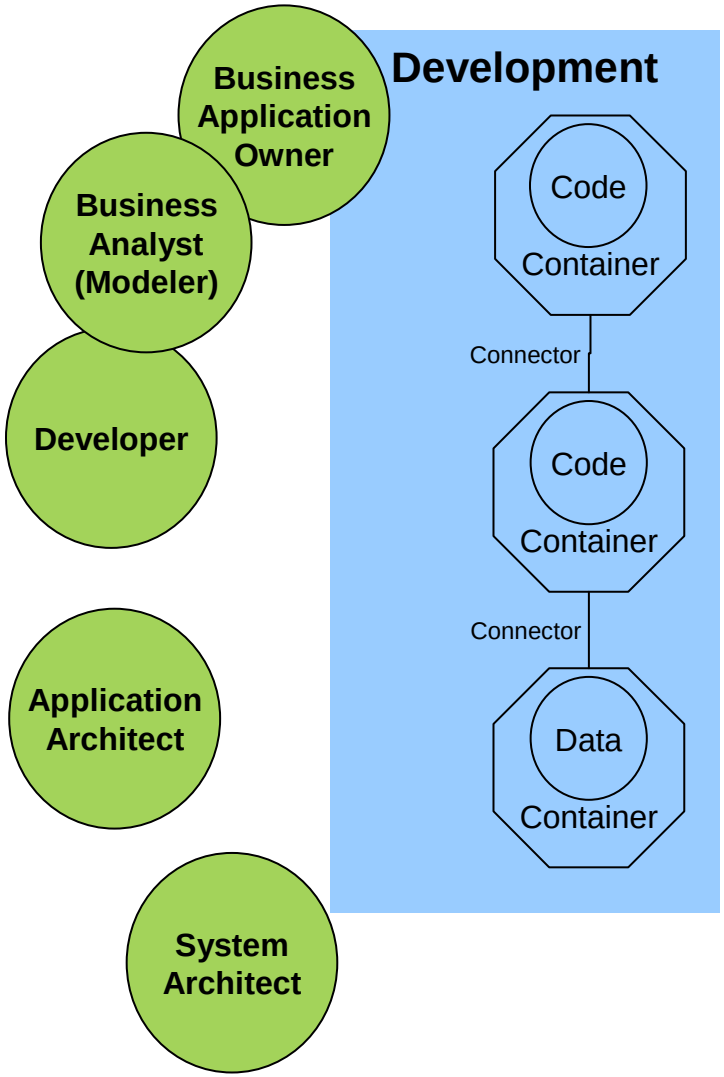
Topics

Solution Viewpoints
1. Build Components



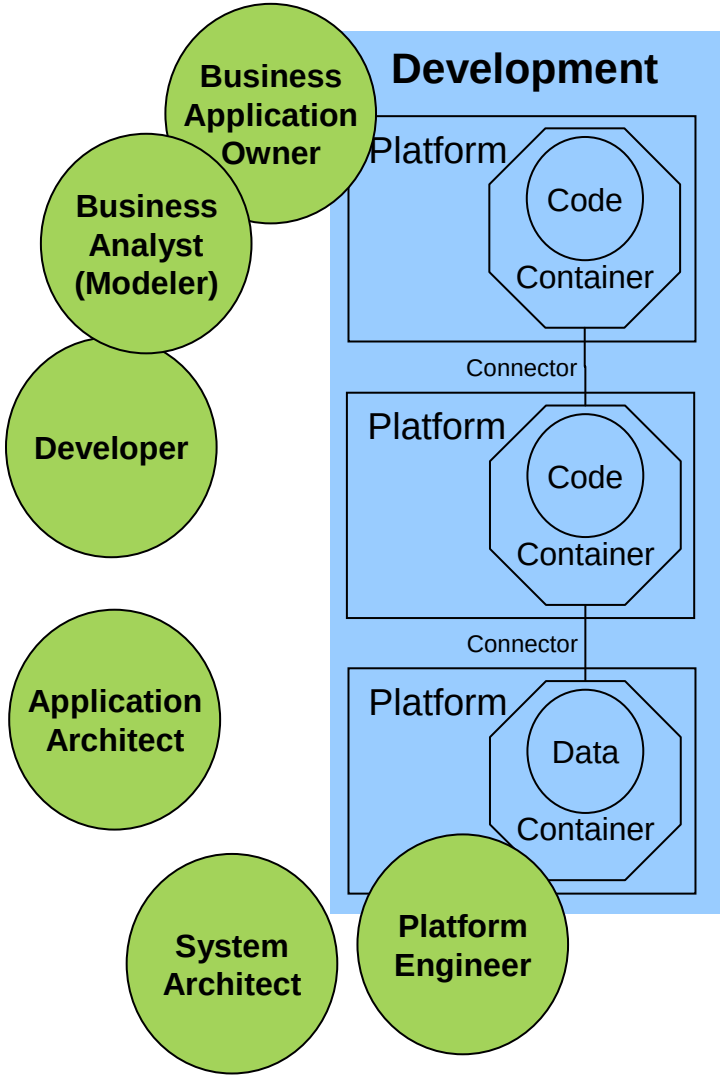
Topics

Solution Viewpoints
2. Build Containers and Connectors



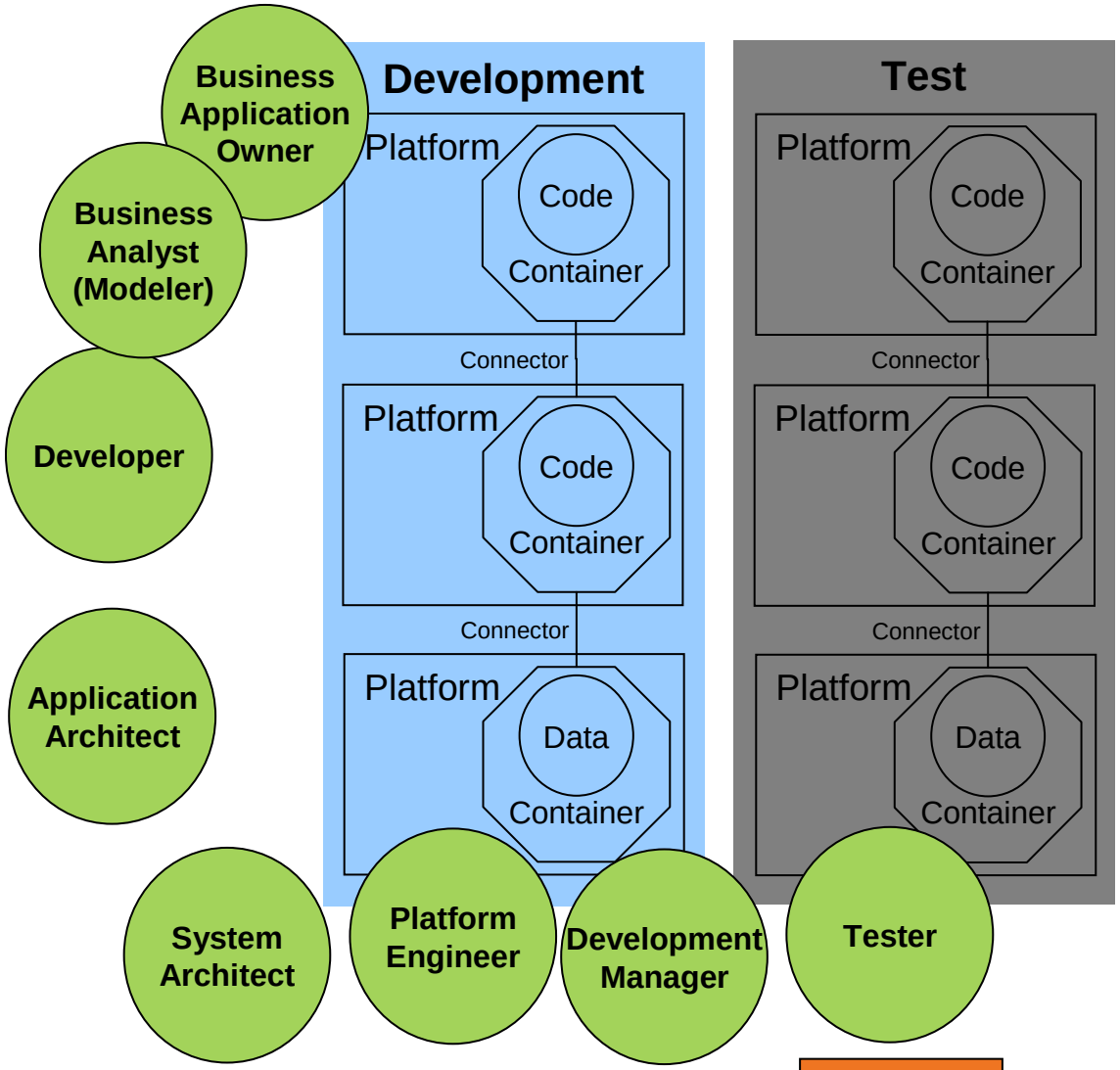
Topics

Solution Viewpoints
3. Build Platforms



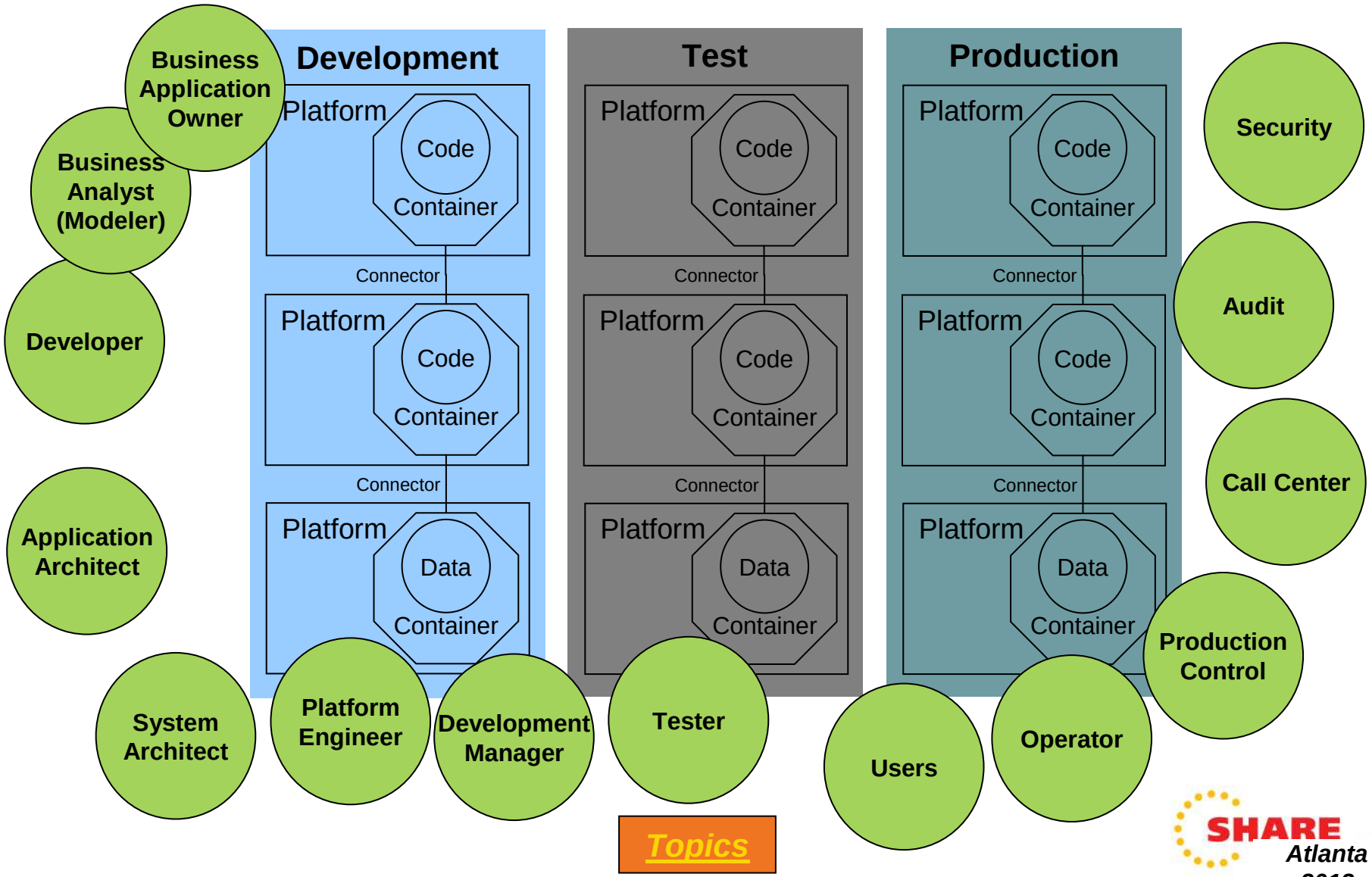
Topics

Solution Viewpoints
4. Build Test Environment



Topics

Solution Viewpoints
5. Build Production Environment



Solution 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

Topics

So What is Optimal?

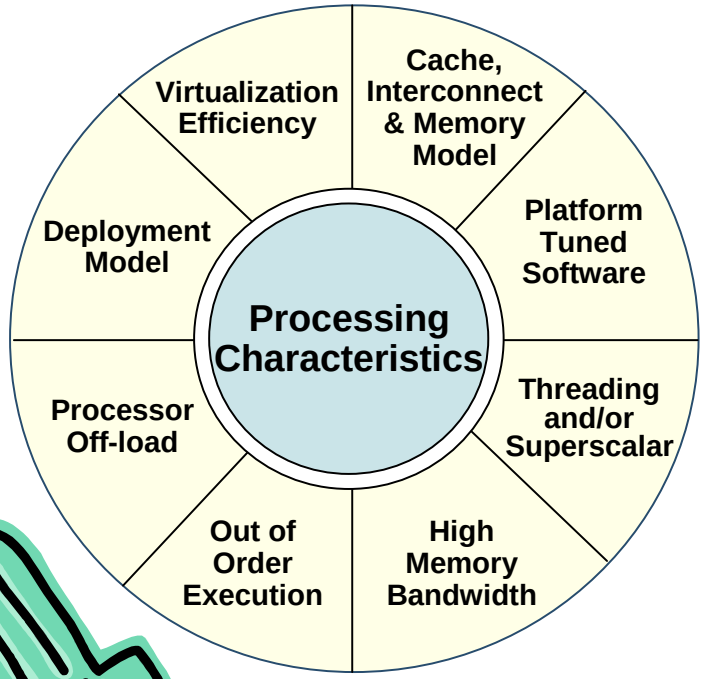
- Balance ALL viewpoints:
  - Requirements-based decision-making
  - Consensus building “argument”
  - Transparent requirements-based tradeoffs
- And decide:
  - How to build **components**
  - What **containers** to use
  - What **connectors** to use
  - **What platform to use**
    - For each container
    - For each lifecycle stage

Topics

Product/Technology Matching Requirements

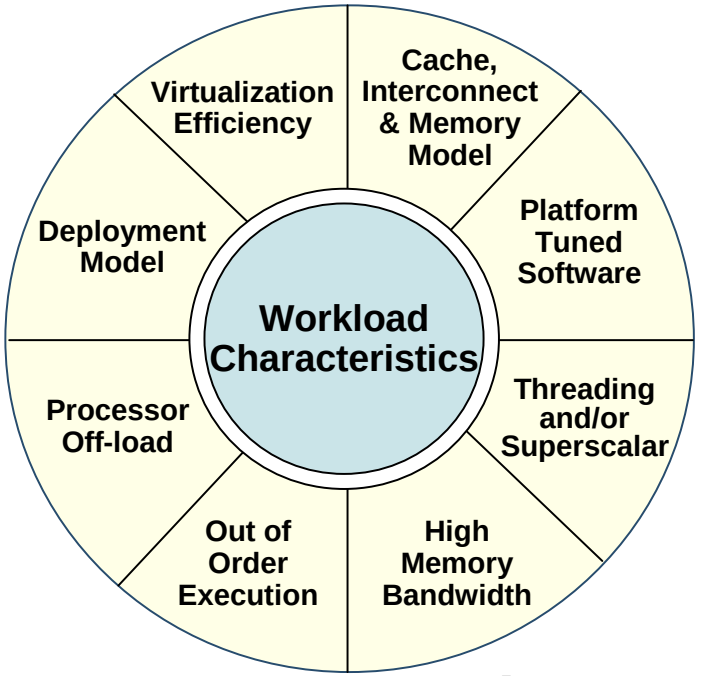
Product

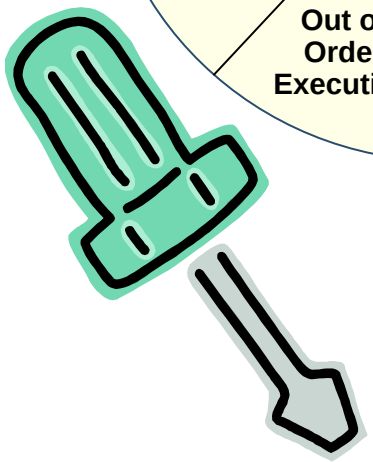
processing characteristics



Workload

processing characteristics





Match the “job” with the “tool”

Topics

Requirements Analysis

- IBM Design Centers
  - Proven Track Record

- Structured Requirements Analysis
  - aka Structured Argument

			Option 1 (x86)		Option 2 (z)	
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
5	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
				123		157

1. Define Requirements
   - coming from all viewpoints
2. Prioritize Requirements
3. Assess Each Solution Option's Ability to Meet Requirements

- ***Create a Requirements Analysis Scorecard***

Fit for Purpose

Topics

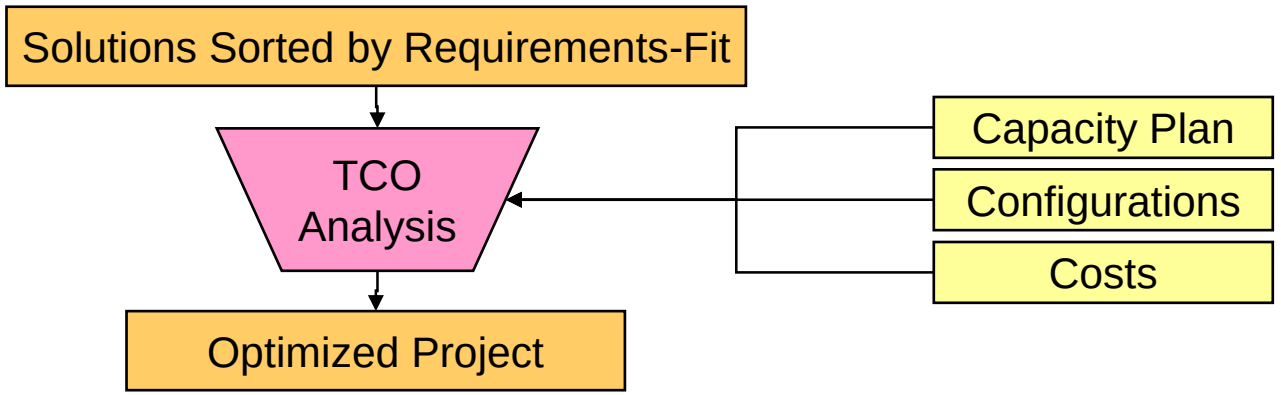
End of Section

Topics

Cost Analysis

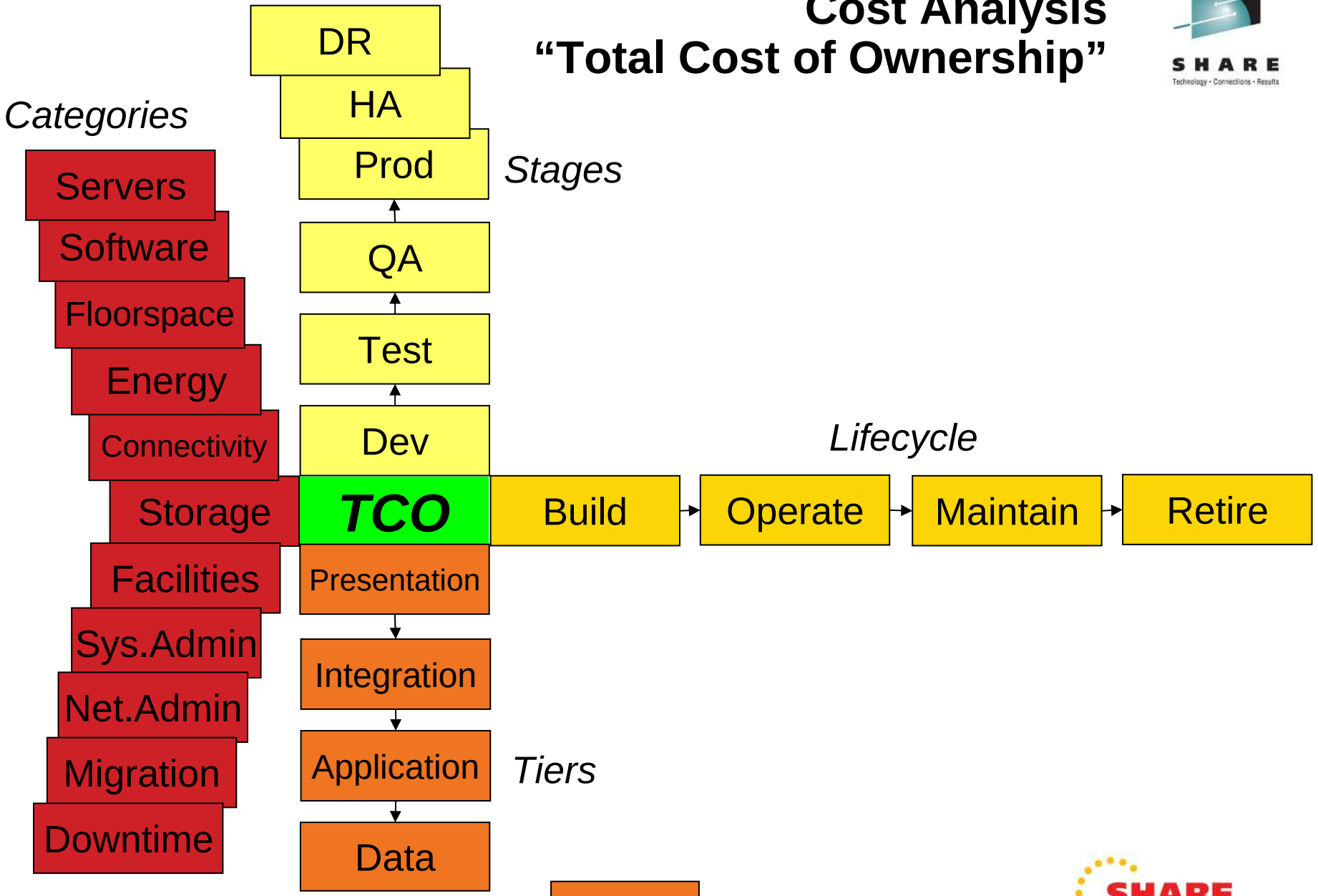
Topics

Cost Analysis



Topics

Cost Analysis
"Total Cost of Ownership"



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

Topics

Cost Analysis - Prerequisites

- 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

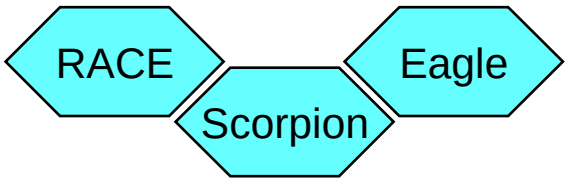
Topics

Cost Analysis – Tools

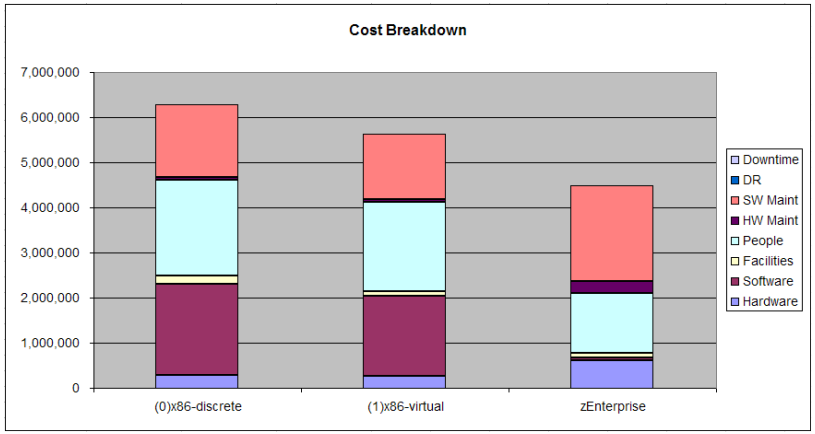
- IBM TCO Modeling Tools
  - RACEv for distributed server (including Linux) costing
  - RACEzOS for z/OS hosted workloads

- Use RACEv and RACEzOS together to model “hybrid solutions”
  - aka Multi-Tier Multi-Platform configurations
    - e.g. zEnterprise Heterogeneous Computing

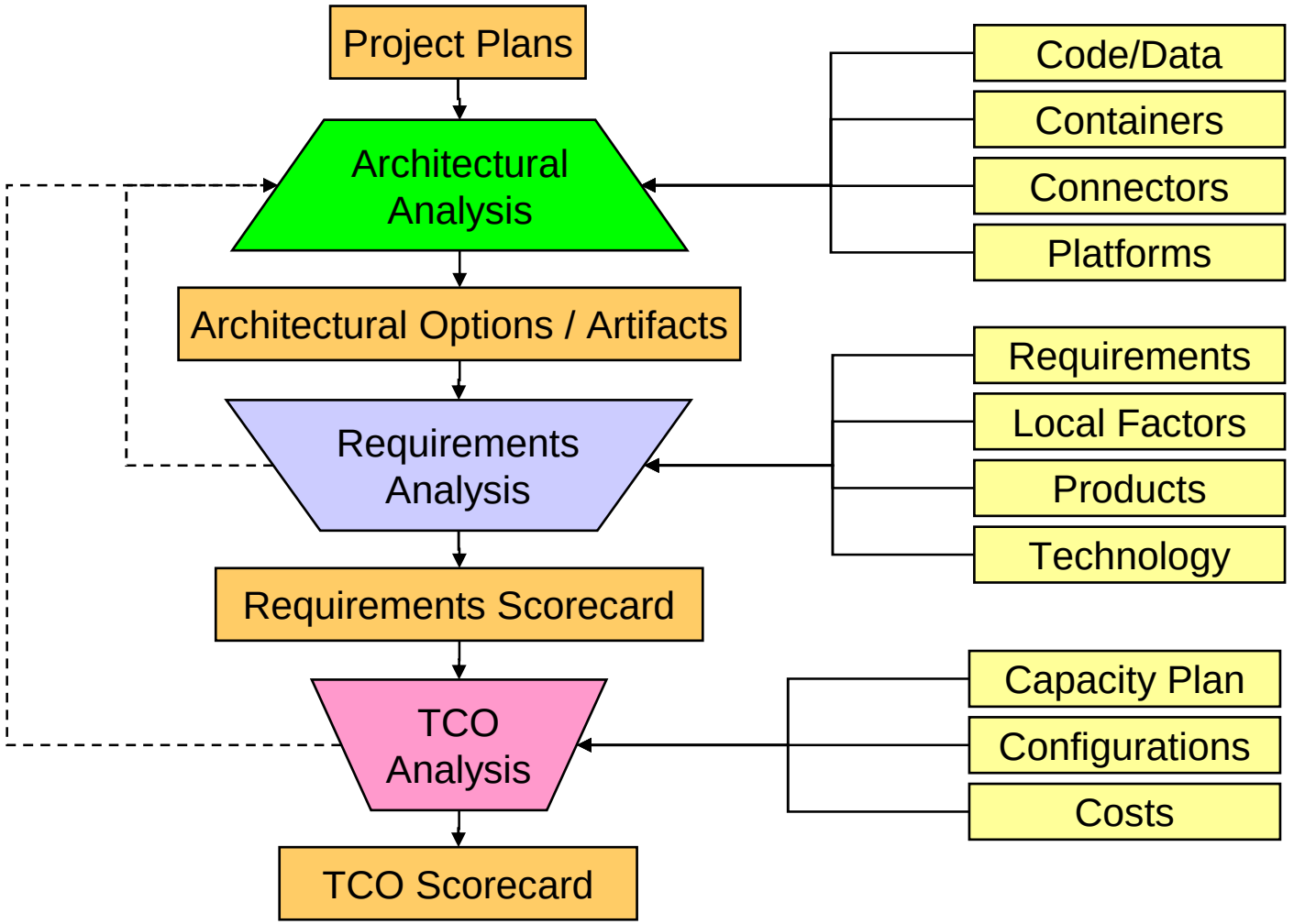
- ***Create a TCO Scorecard***



Topics



IT Optimization Assessment Methodology Putting it All Together



Topics

End of Section

Topics

Case Study

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)

Topics

Business 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)

Topics

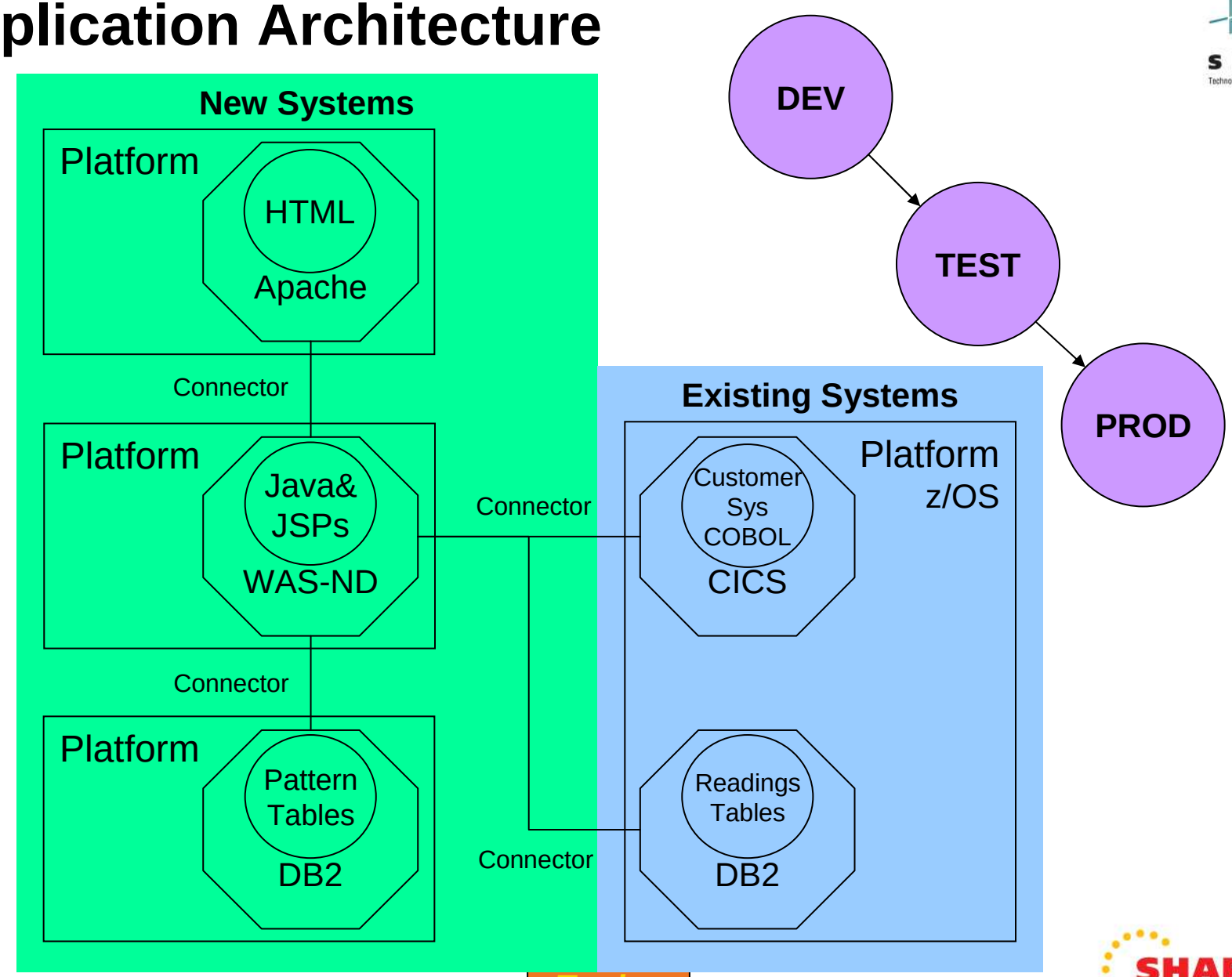
End of Section

Topics

Case Study – Architectural Analysis

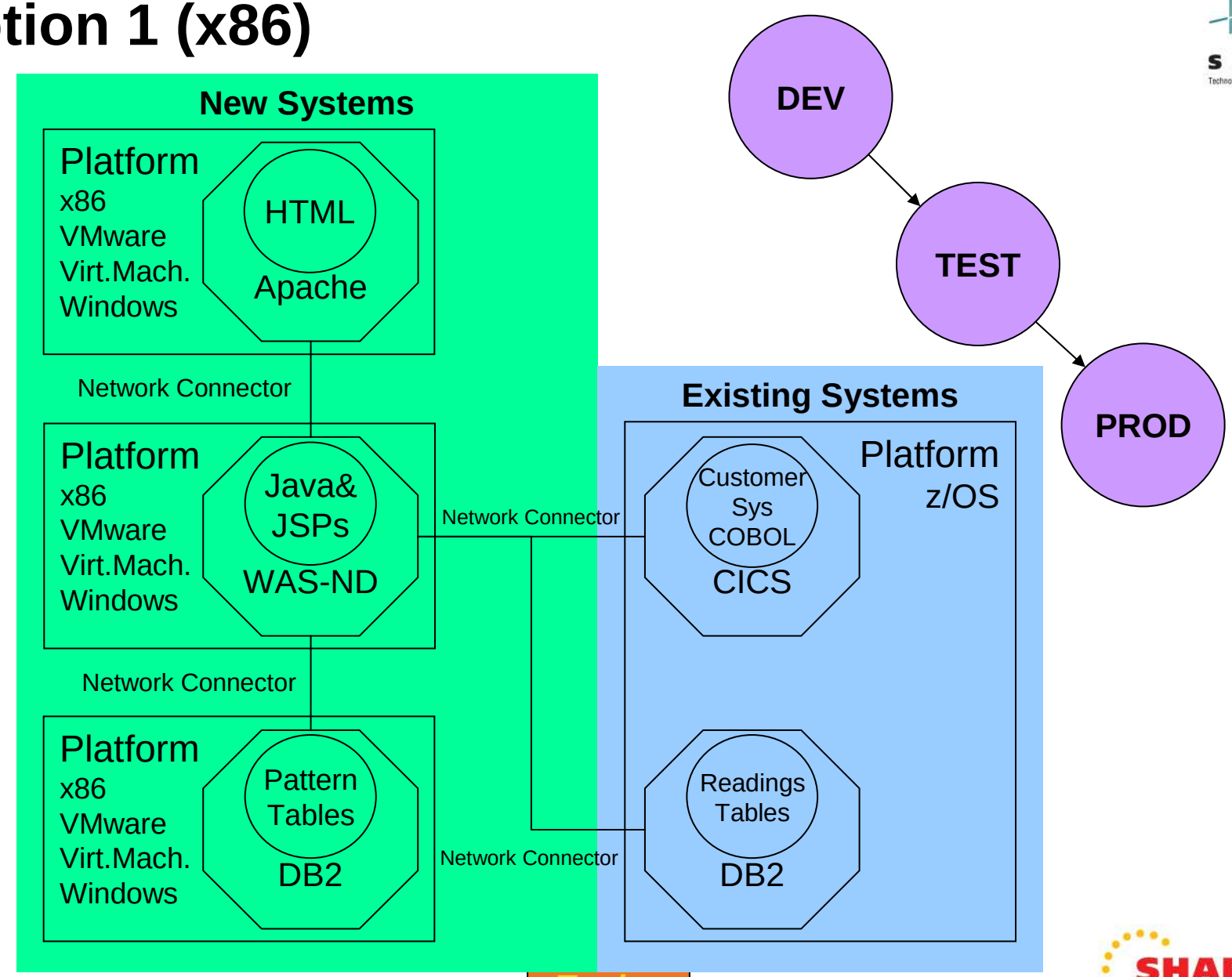
Topics

Application Architecture



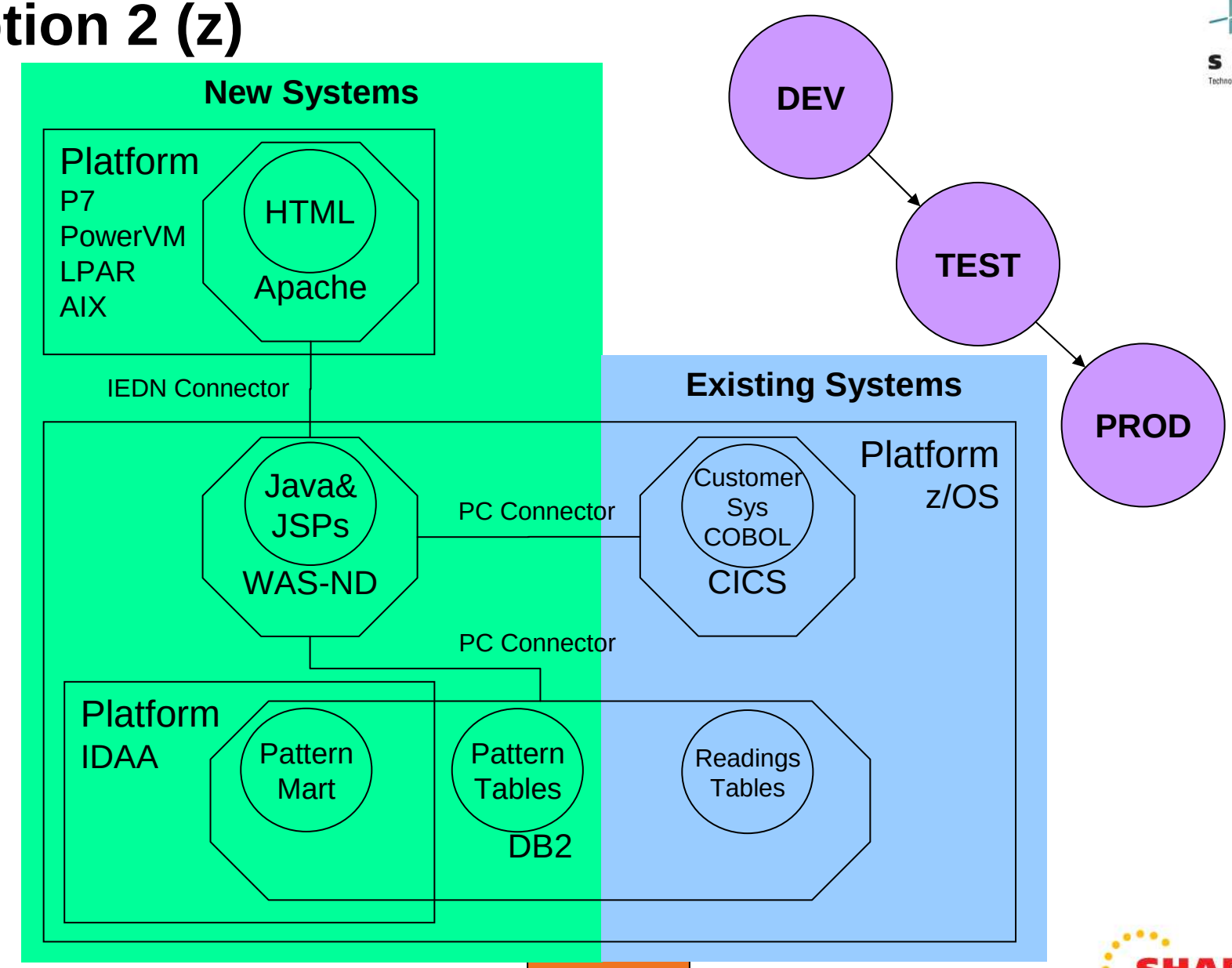
Topics

Option 1 (x86)



Topics

Option 2 (z)



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

Topics

End of Section

Topics

Case Study – Requirements Analysis

Topics

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

Topics

Requirements List

Requirements List
Data privacy
Scalability
Integration with existing customer management system
Development lifecycle support
Availability
Performance
Manageability
Integration with existing smart meter readings database

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

Rank	Requirements List	Weight
1	Performance	6
2	Scalability	5
3	Integration with existing customer management system	4
4	Integration with existing smart meter readings database	4
5	Availability	4
6	Data privacy	3
7	Manageability	3
8	Development lifecycle support	3

Score the Options

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

Qualifier	Label	Score
Demonstrated Can Exceed	D.Exceed	7
Likely to Exceed	L.Exceed	6
Demonstrated Can Meet	D.Meet	5
Likely to Meet	L.Meet	4
Marginal	Marginal	2
Does Not Meet	No.Meet	0

			Option 1 (x86)		Option 2 (z)	
Rank	Requirements List	Weight	Score		Score	
1	Performance	6	L.Meet		L.Exceed	
2	Scalability	5	L.Meet		L.Exceed	
3	Integration with existing customer management system	4	L.Meet		D.Meet	
4	Integration with existing smart meter readings database	4	L.Meet		D.Meet	
5	Availability	4	Marginal		D.Meet	
6	Data privacy	3	L.Meet		D.Meet	
7	Manageability	3	L.Meet		D.Meet	
8	Development lifecycle support	3	D.Meet		L.Meet	

Topics

Rate the Options

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

			Option 1 (x86)		Option 2 (z)	
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
5	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
				123		157

This is a "Requirements Scorecard"

Topics

End of Section

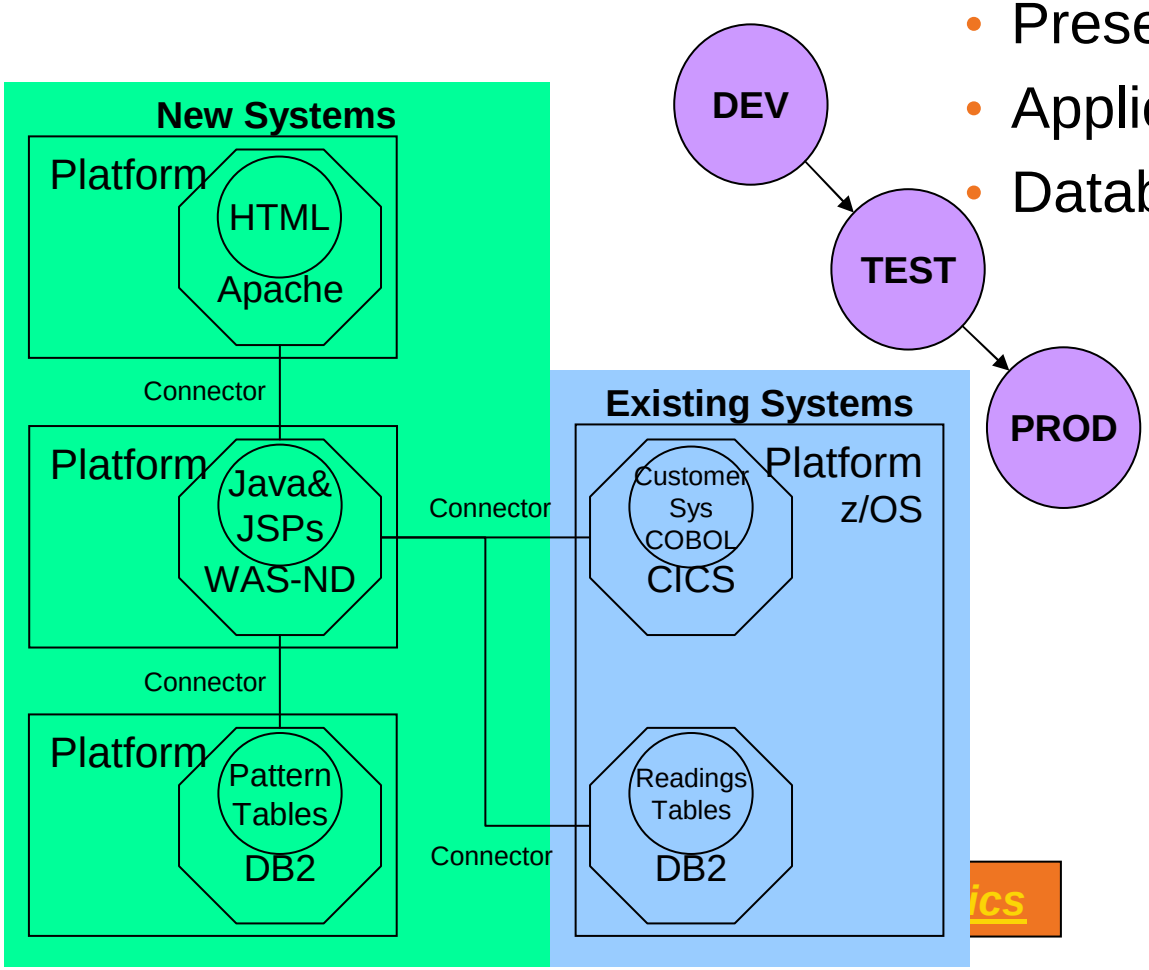
Topics

Case Study – Cost Analysis

Topics

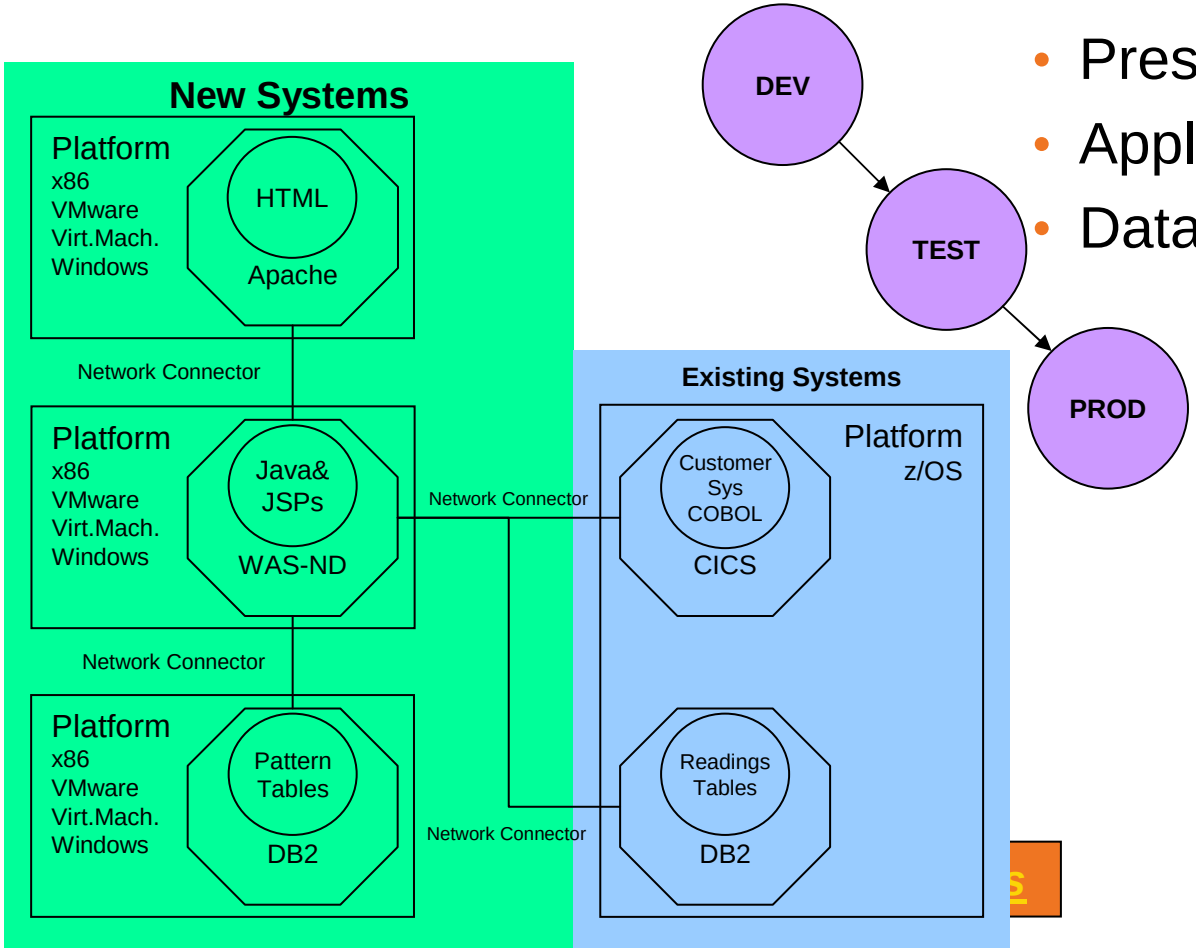
Step 1 – RACEv Subject Servers

- The set of discrete x86 servers
  - Presentation – Prod/Test/Dev
  - Application – Prod/Test/Dev
  - Database – Prod/Test/Dev



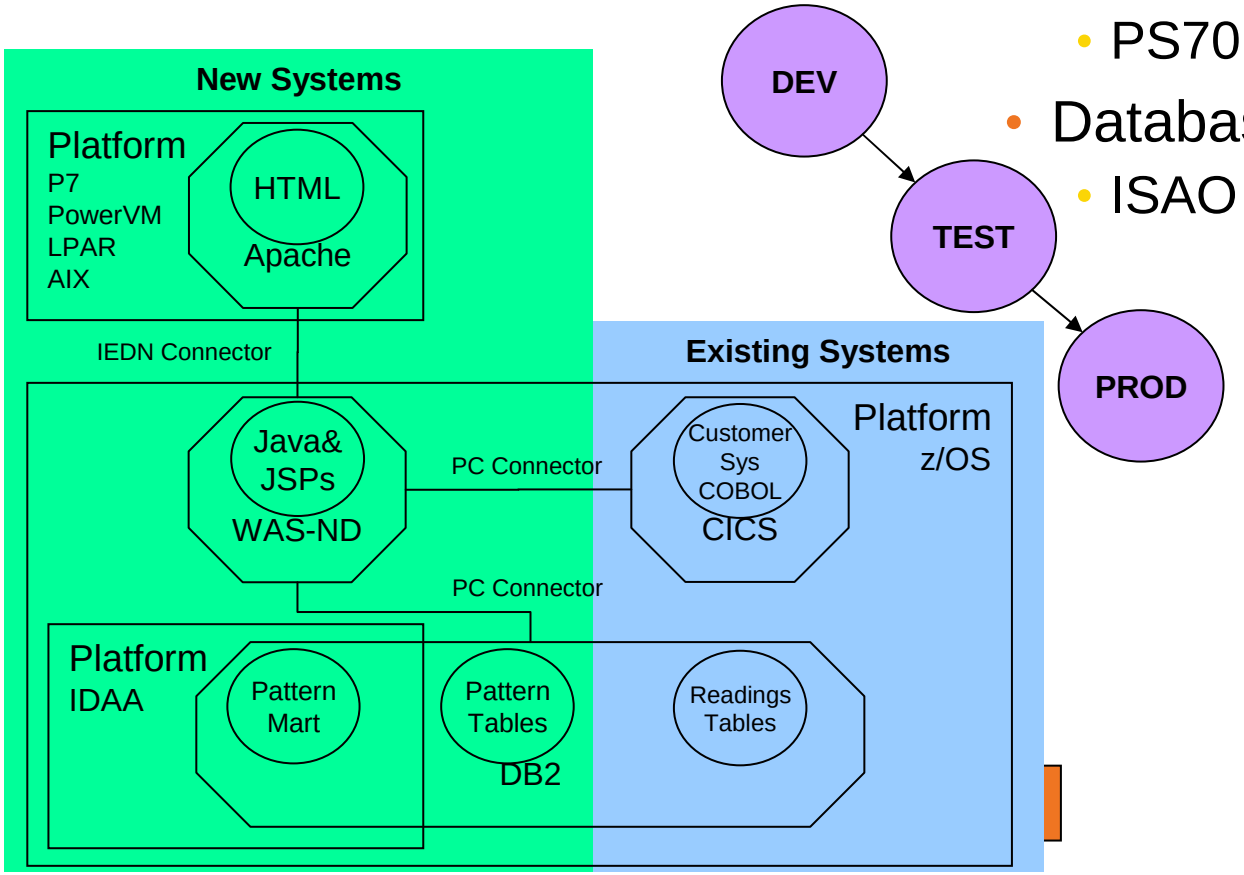
Step 2 – RACEv x86 Server Target

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



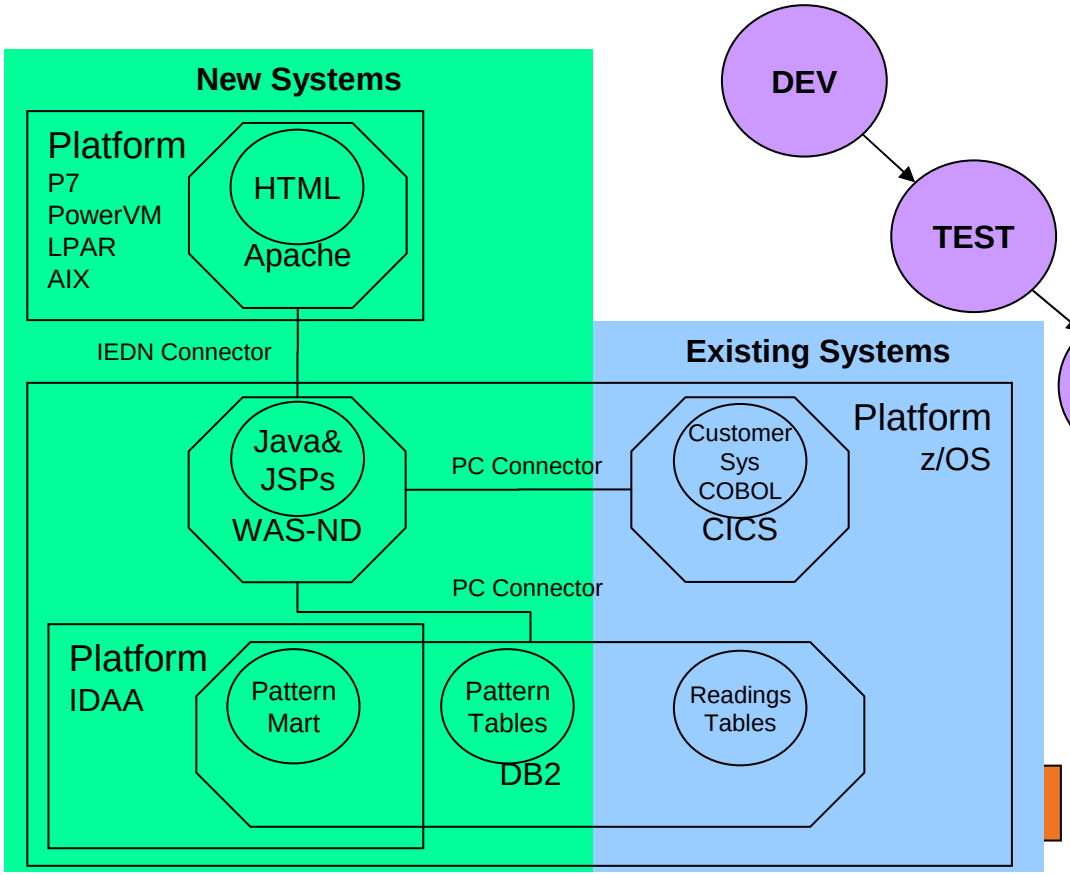
Step 3 – RACEv zEnterprise Target Distributed Elements

- 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

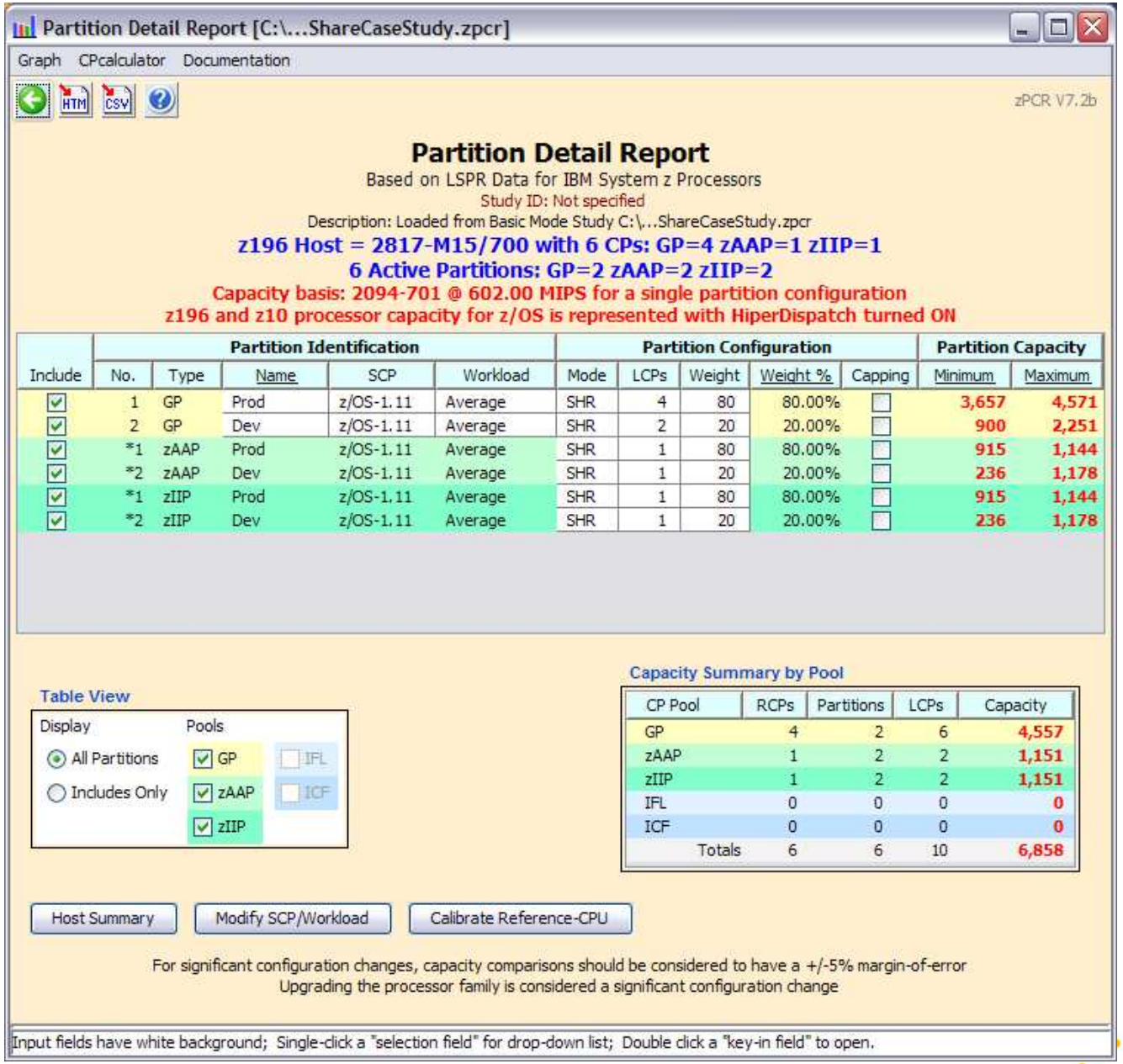


Step 4 – RACEzOS zEnterprise Elements

- 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



z196



Partition Detail Report [C:\...ShareCaseStudy.zpcr]

Graph CPcalculator Documentation

zPCR V7.2b

Partition Detail Report

Based on LSPR Data for IBM System z Processors

Study ID: Not specified

Description: Loaded from Basic Mode Study C:\...ShareCaseStudy.zpcr

z196 Host = 2817-M15/700 with 6 CPs: GP=4 zAAP=1 zIIP=1

6 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 turned ON

	Partition Identification					Partition Configuration					Partition Capacity	
Include	No.	Type	Name	SCP	Workload	Mode	LCPs	Weight	Weight %	Capping	Minimum	Maximum
☑	1	GP	Prod	z/OS-1.11	Average	SHR	4	80	80.00%	☐	3,657	4,571
☑	2	GP	Dev	z/OS-1.11	Average	SHR	2	20	20.00%	☐	900	2,251
☑	*1	zAAP	Prod	z/OS-1.11	Average	SHR	1	80	80.00%	☐	915	1,144
☑	*2	zAAP	Dev	z/OS-1.11	Average	SHR	1	20	20.00%	☐	236	1,178
☑	*1	zIIP	Prod	z/OS-1.11	Average	SHR	1	80	80.00%	☐	915	1,144
☑	*2	zIIP	Dev	z/OS-1.11	Average	SHR	1	20	20.00%	☐	236	1,178

Table View

Display: ◉ All Partitions ○ Includes Only

Pools: ☑ GP ☐ IFL ☑ zAAP ☐ ICF ☑ zIIP

Capacity Summary by Pool

CP Pool	RCPs	Partitions	LCPs	Capacity
GP	4	2	6	4,557
zAAP	1	2	2	1,151
zIIP	1	2	2	1,151
IFL	0	0	0	0
ICF	0	0	0	0
Totals	6	6	10	6,858

Host Summary | Modify SCP/Workload | Calibrate Reference-CPU

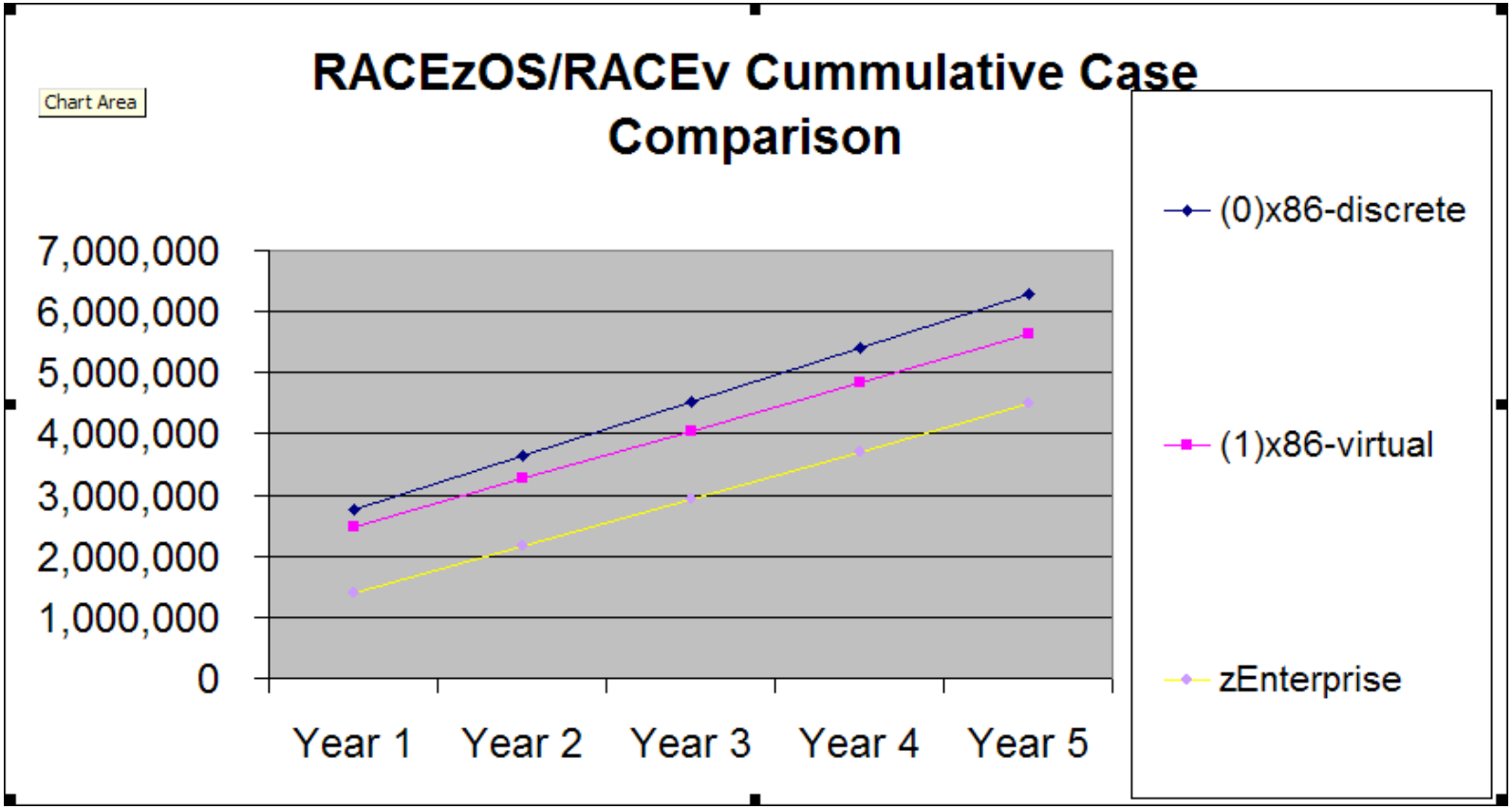
For significant configuration changes, capacity comparisons should be considered to have a +/-5% margin-of-error

Upgrading the processor family is considered a significant configuration change

Input fields have white background; Single-click a "selection field" for drop-down list; Double click a "key-in field" to open.

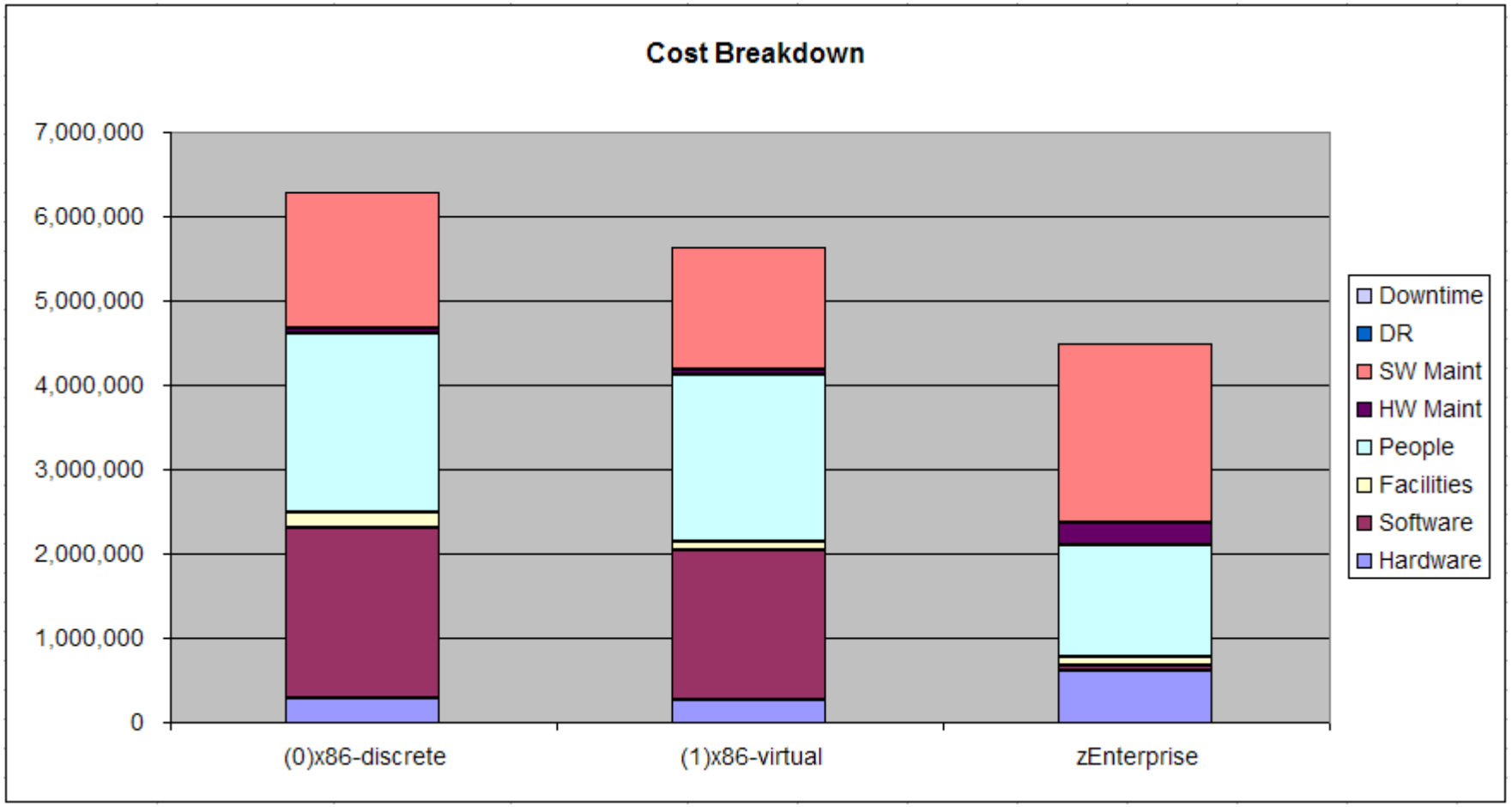
Topics

Step 6 – Complete the Models
Examine the Results



Topics

Step 7 – Examine More Results

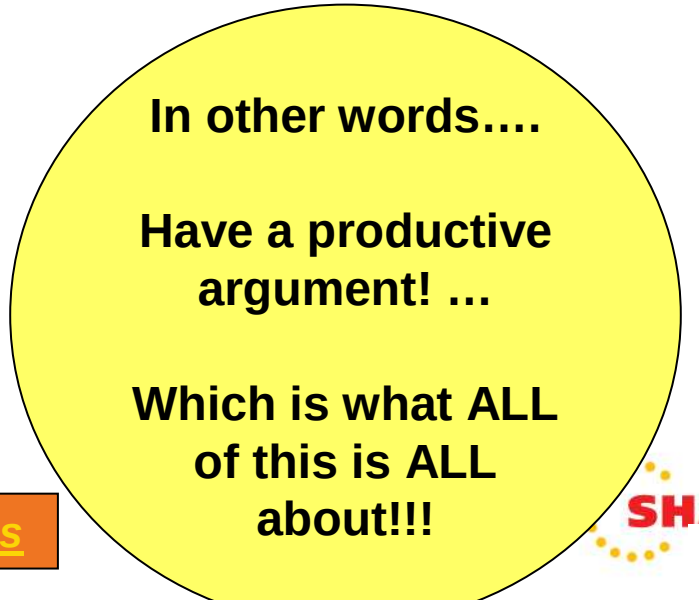


Topics

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.

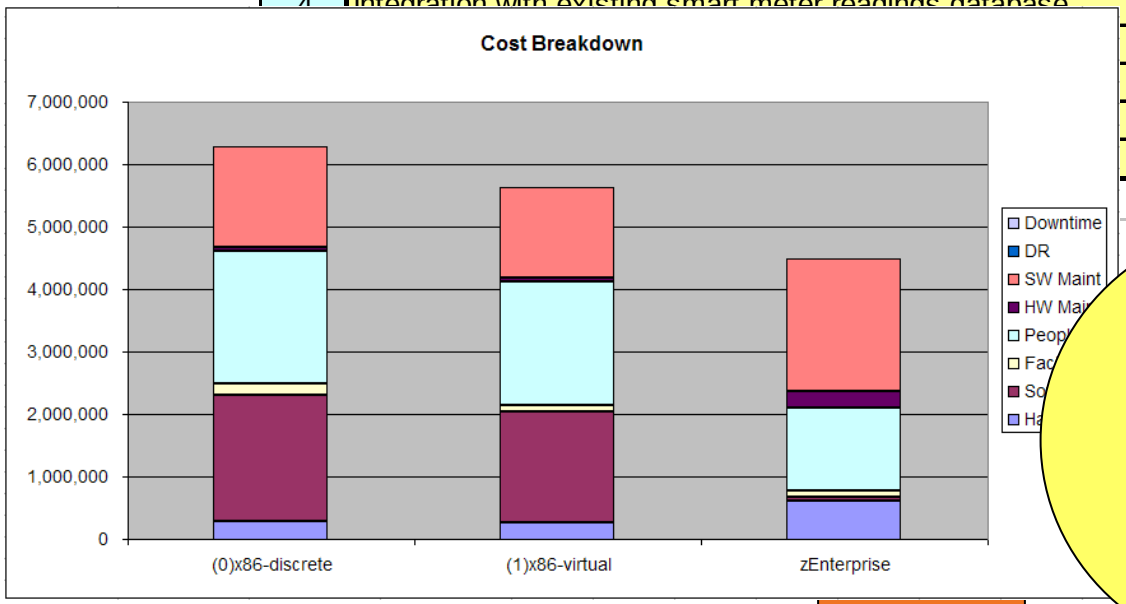


Topics

And Finally

- Merge the Requirements Analysis…
- And the TCO Analysis

			Option 1 (x86)		Option 2 (z)	
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	Marginal	8	D.Meet	20
		3	L.Meet	12	D.Meet	15
		3	L.Meet	12	D.Meet	15
		3	D.Meet	15	L.Meet	12
				123		157



And make an "optimal" decision!

Topics

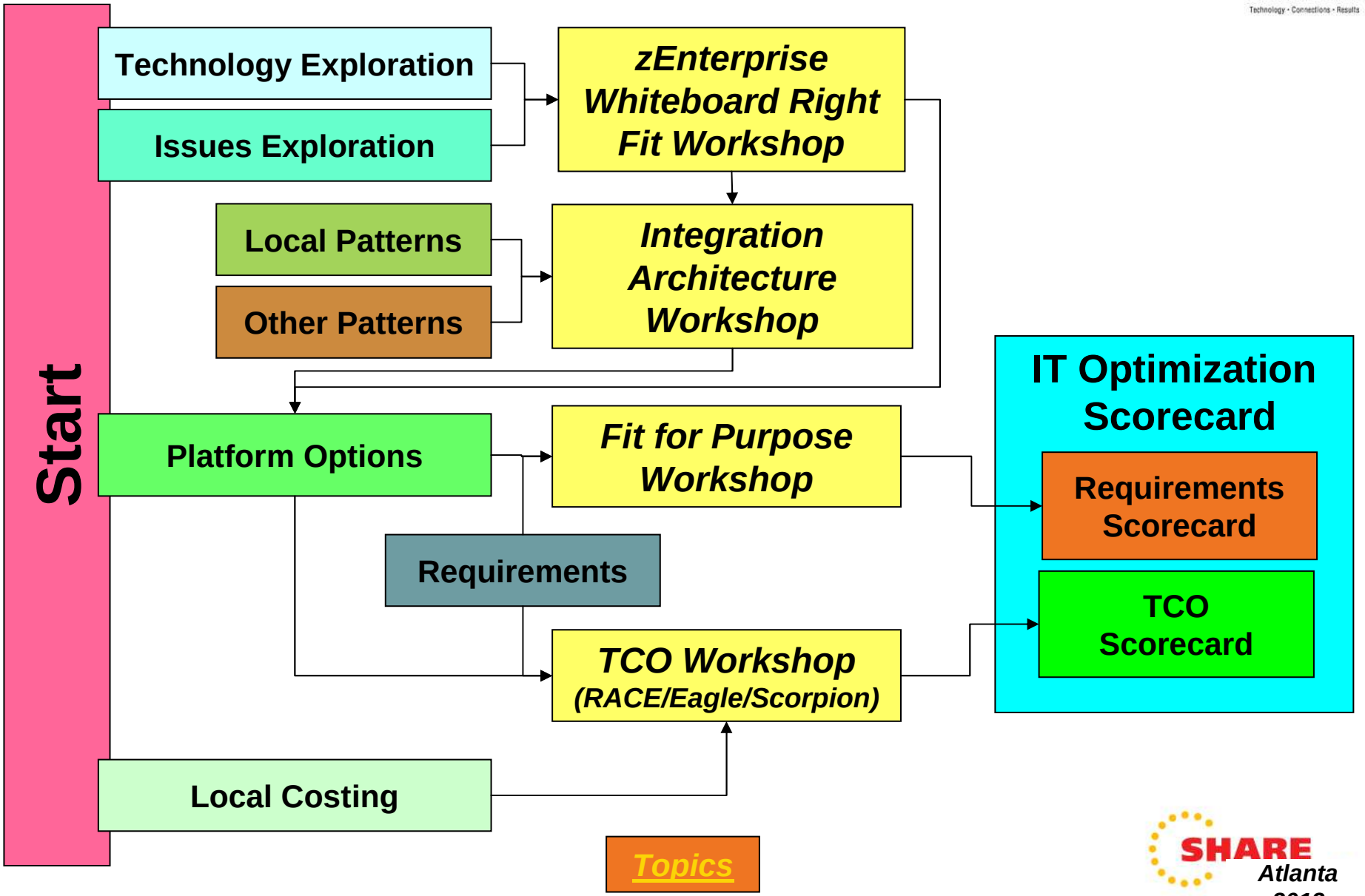
End of Section

Topics

Conclusion

Topics

IBM IT Optimization Analysis



End of Section

Topics

Questions?

Comments?

Critiques?

Requests??!?!!

Monte Bauman

IBM Columbus

mbauman@us.ibm.com

Thank You

Topics