

Operational Aspect





Module outline

▶ **Introduction and objectives**

What is Operational Modeling (OM) and why do we need it?

How does OM evolve and how is it represented?

Constructing an Operational Model

Summary and references



Learning objectives

At the end of this lecture, you should be able to:

- Understand and explain the:
 - Purpose of the operational aspect of architecture
 - Key concepts and artifacts of the operational aspect
 - Relationships between the functional aspect and the operational aspect
 - Techniques for creating Operational Models from organized sets of requirements
- Assist an experienced Architect in constructing and documenting the Operational Model
- Review an existing Operational Model document for completeness



Module outline

Introduction and objectives

► **What is Operational Modeling (OM) and why do we need it?**

How does OM evolve and how is it represented?

Constructing an Operational Model

Summary and references



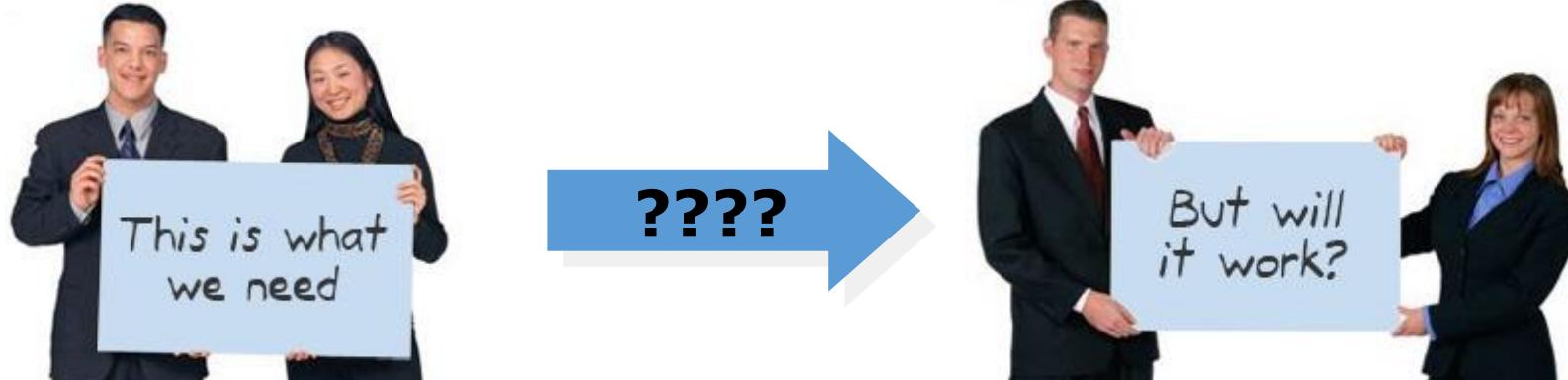
- The Functional aspect of Architecture <- **Previous Module**
 - Describe the structure of components (of all kinds) related to actual functionality,
 - ...their dynamic behavior (collaboration),
 - ...and their interfaces.
 - Embodied in the Component Model work product
- **The Operational aspect of Architecture <- This Module**
 - Describes what runs where (component placement).
 - ...and includes the network topology (hardware nodes, locations, etc.),
 - It ensures the achievement of the solution's service level characteristics (performance, availability),
 - ...and describes the management and operation of the IT system.
 - Embodied in the Operational Model work product



Functional systems must be deployed operationally - either using new or an existing infrastructure/cloud services

The design of a system's operational aspect may be challenging to develop or understand.

- Many competing concerns must be resolved:
 - NFR's: for example, performance, availability, manageability, security
 - Constraints: for example, affordability, standards compliance, existing Infrastructure
- Multiple activities must be coordinated:
 - **Detailed design:** capacity planning, software and data distribution, network, hardware, service management
 - **Commercial:** planning, pricing, procurement, service level agreement

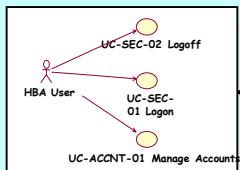


The design of a system's operational aspect may be challenging to develop or understand.

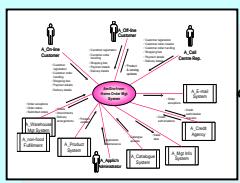


Operational Modeling helps ensure the system's non-functional (NFR) requirements are delivered, within all constraints

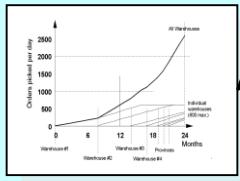
System Requirements



What do we have to do?
(Use Case Model)

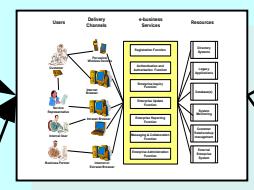


How do we fit in?
(System Context)

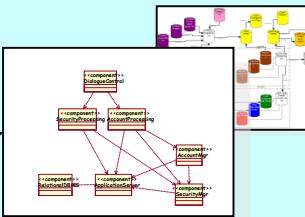


How good does it have to be?
(NFR Requirements)

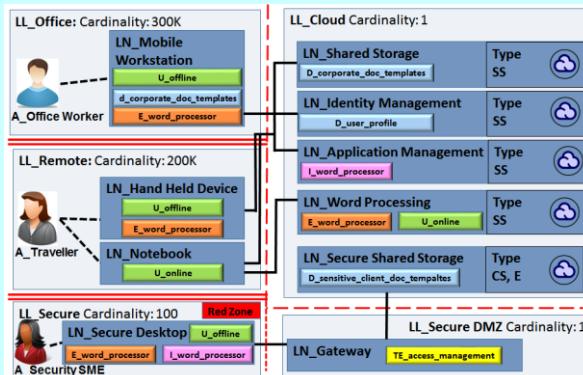
System Architecture



What is our approach?
(Architecture Overview)

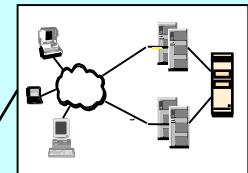


How is the application structured?
(Component Model & Data Model)



Where does everything go?
(Operational Model)

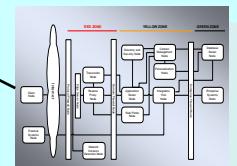
System Constraints



What is the current environment?
(Current IT)



What other constraints are there?
(Standards)



What experience do we have?
(Reference Models)

Designing for Performance and Capacity

Designing for Availability

Designing for Security

Designing for System management

Assessing viability



Time for a question



Select the correct statement from the following:

- A. The Operational Model focuses on the structure of the solution by identifying components and their collaborations.
- B. Service level requirements are delivered through the deployment of the components identified in the Operational Model.
- C. The Operational Model describes how components are supported to meet the service level requirements.
- D. The Operational Model is used as input when building the Location Model and the Node Model.



Module outline

Introduction and objectives

What is Operational Modeling (OM) and why do we need it?

► **How does OM evolve and how is it represented?**

Constructing an Operational Model

Summary and references

The Operational Modelling technique consists of multiple steps



- 1. Understand the non-functional requirements**
 - What NFRs need to be met ?
- 2. Conceptualize from an application perspective (ALOM¹):**
 - Identify application level nodes and their placement.
- 3. Conceptualize the Logical Operational Model (LOM):**
 - Determine where all of the logical application components are deployed.
 - Determine what logical technical components are required and where they will be deployed to support the logical application components.
- 4. Transform into the Physical Operational Model (POM)**
 - Determine real technologies to deliver all of these specifications.

1M transactions per hour

Finance

Warehouse Management

Customer Relationship

ESB

Load balancer

Database

SAP

Message Broker

F5 BIG-IP LTM-4200v

DB2

Application

Logical

Physical

¹*Application Logical Operational Model (ALOM) is not discussed in this lecture – see the Operational Modelling class for details on ALOM*

To achieve this goal, we need a structured, formal language to describe the elements featured on the Operational Model



- The Operational Model represents the system’s “infrastructure architecture” using a variety of model elements, including:
 - The geographic structure of the **locations** and their **borders**, over which the IT system will be deployed and operated
 - The placement of the system’s **nodes** into these locations
 - The deployment of the system’s components across these nodes, using **deployment units**
 - The **connections** between nodes
 - The organization of the system’s elements into **zones**
- In addition to this description of the deployed system, the OM also documents:
 - The overall **physical configuration** of the technologies and products necessary to deliver the functional and nonfunctional requirements of the IT system
 - Sizing and other **hardware specifications** for all the computers, storage devices, and network technologies

Operational Modeling (OM) notation used throughout this lecture and upcoming exercises.



Actors & Infrastructure

| Type | Logical Example | Physical Example |
|--------------|-----------------------|--|
| Location | LL_Office | PL_IBM Bolumbia Head Office |
| Node | LN_Mobile Workstation | PN_Apple Laptop |
| Human Actor | A_Office Worker |   |
| System Actor | | A_Respondent PC |

Connections

No permitted connection

External intermittent connection

Internal high speed connection

External high speed connection

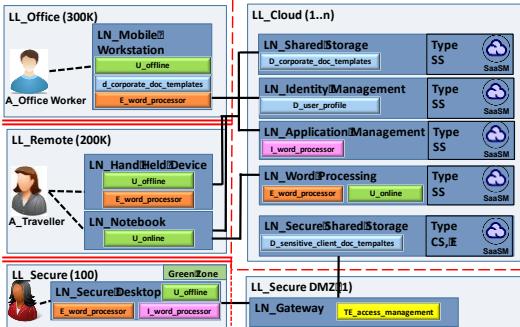
Deployment Units

Capital letter = Master copy
and lower case = copy

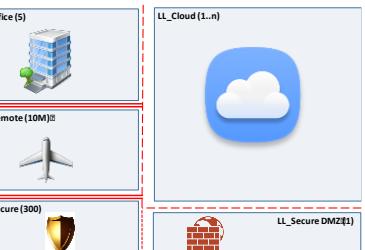
| Type | Short ID | Long ID |
|--------------|----------|---------|
| Presentation | U<X> | U_Name |
| Execution | E<X> | E_Name |
| Installation | I<X> | I_Name |
| Data | D/d<X> | D_Name |



Operational Model inter-dependencies with other artefacts

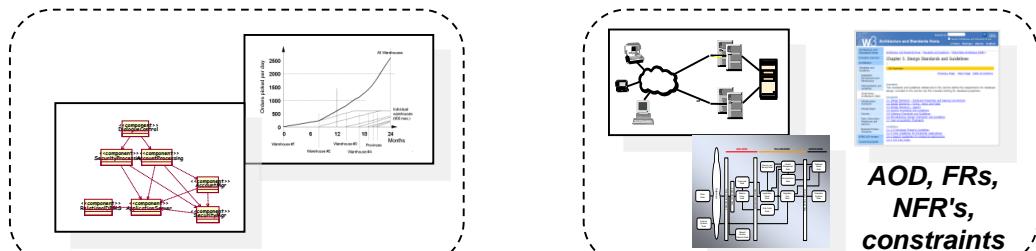


**Where does everything go?
(Operational Model)**



**Where can we put things?
(Location Model)**

These contained artifacts give us a way of “building up” the Operational Model...



**AOD, FRs,
NFR's,
constraints**

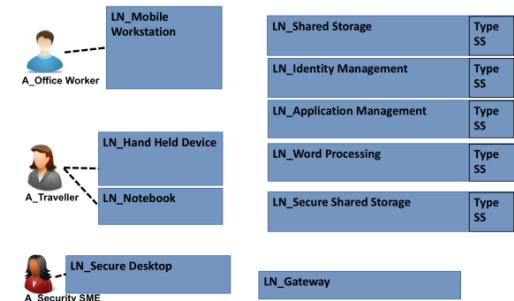
Component Model

Based on the CM

2

| Presentation DU | IT Skill level | Location | Frequency | Type |
|---------------------|------------------------------|-----------------|-----------|------------|
| U_Inventory | Can be trained | Mobile | N/A | N/A |
| U_SMS | N/A | N/A | Weekly | Batch |
| Data DU | Scope | Volatility | Size | Lifetime |
| D_Inventory | Store | High | Small | <= 1 day |
| D_Cat | Entire product range | Low | Large | Long lived |
| Execution DU | Frequency | Processing Load | | |
| E_Submit New Update | Daily (once for every store) | Light | | |
| E_Consolidate | Daily | Heavy | | |

**What do we need to deploy?
(Deployment Unit Model)**



**Where should we
deploy things?
(Node Model)**



Logical Location

LL_Office (5)



LL_Cloud (1..n)



LL_Remote (10M)



LL_Secure (300)



LL_Secure DMZ (1)



Presentation

Entry point used by actors to access a components functionality e.g. live web interface.

Data

The data used by components e.g. user profile.

Execution

The runtime execution aspect of a component e.g. 3 instances of a word processor running in 3 different user accounts on the same mobile workstation.

Installation

Files required to install the component e.g. executable, configuration and data.



List deployment units for each component

| Component (from the CM) | Presentation Deployment Unit | Data Deployment Unit | Execution Deployment Unit |
|----------------------------------|---------------------------------|-------------------------|---|
| Customer Relationship Management | U_CRM | D_CRM_Data | E_Update_CRM_Data E_Consult_CRM_Data |
| Stock Management | U_SMS U_Inventory | D_Inventory_Upd | E_Submit_Inv_Update E_Consolidate_Inv_Update |

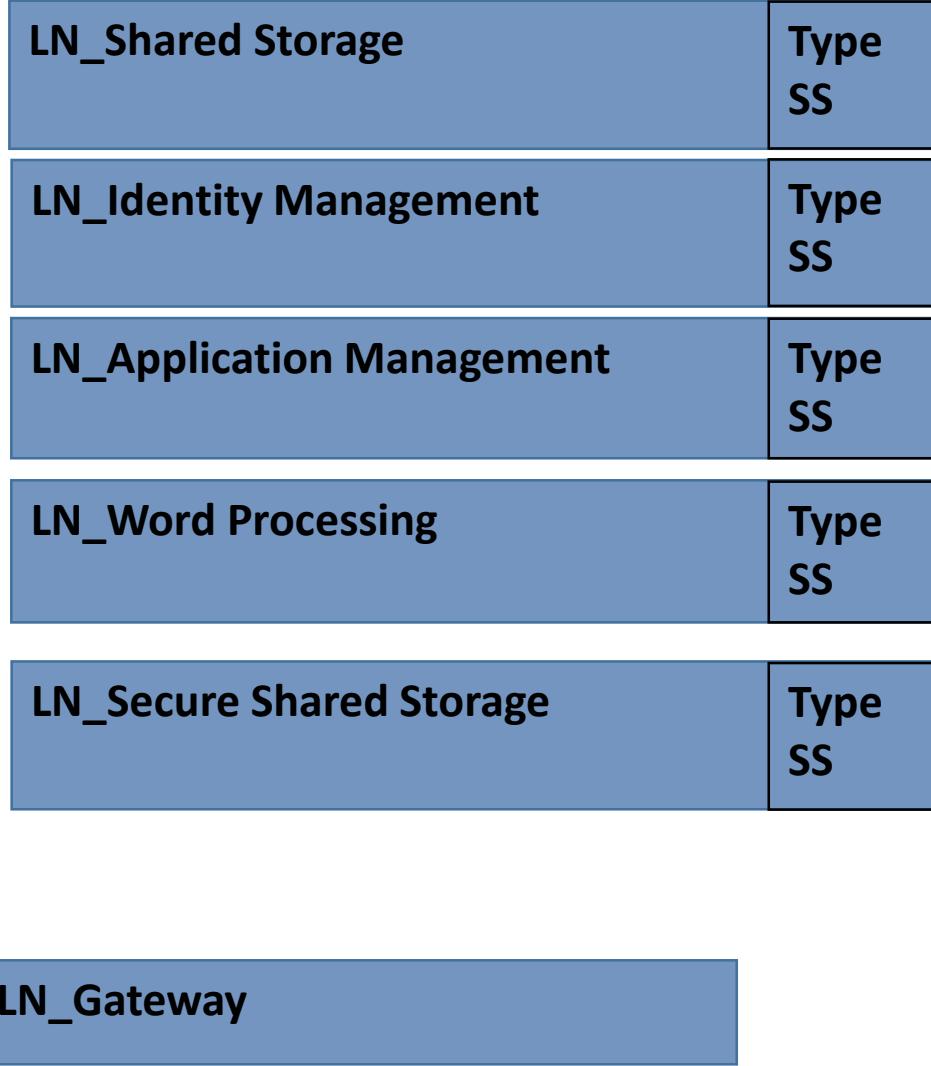
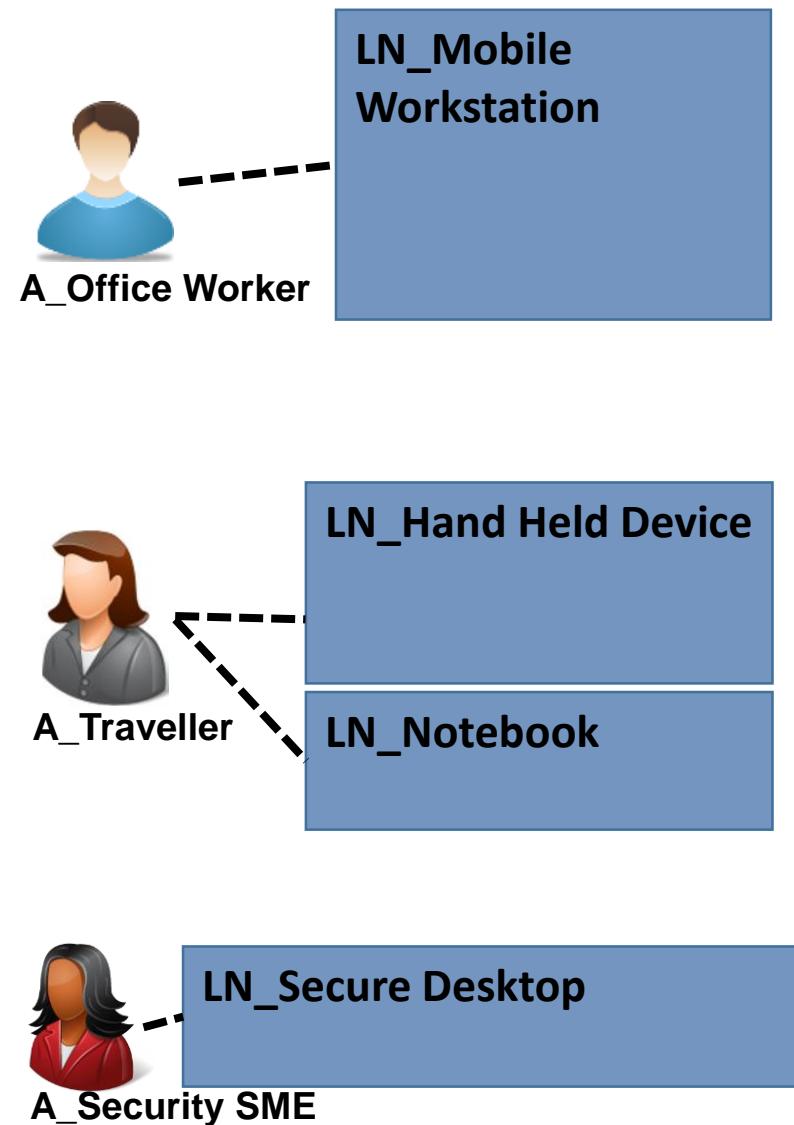
DUs need to be described regarding their non-functional requirements which determine their placement and grouping.

| PresentationDU | Actor | IT Skill level | Location | Frequency | Type |
|----------------|-------------------|----------------|----------|-----------|-------|
| U_Inventory | A_Inventory_Check | Can be trained | Mobile | N/A | N/A |
| U_SMS | A_SMS | N/A | N/A | Weekly | Batch |

| Data DU | Scope | Volatility | Size | Lifetime | Master or Copy |
|-------------|----------------------|------------|-------|------------|----------------|
| D_Inventory | Store | High | Small | <= 1 day | N/A |
| D_Cat | Entire product range | Low | Large | Long lived | M |

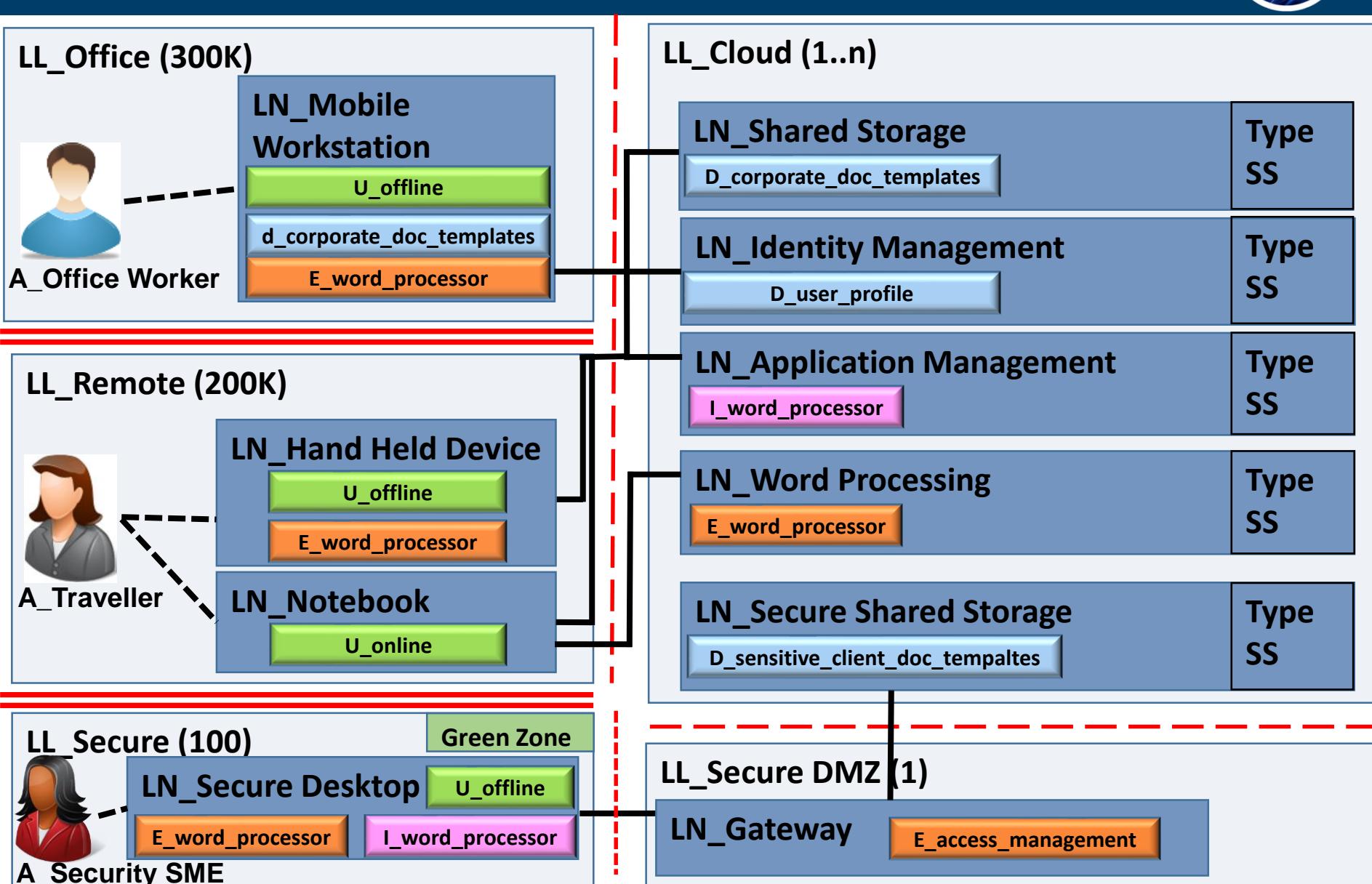
| Execution DU | Frequency | Processing Load |
|---------------------|------------------------------|-----------------|
| E_Submit_Inv_Update | Daily (once for every store) | Light |
| E_Consolidate | Daily | Heavy |

Logical Node Model





Logical Operational Model - shows the static relationships between nodes, DUs, connections and interactions on a geographic background





Topic outline

- 1. Understand the non-functional requirements**
 - What NFRs need to be met ?
- 2. Conceptualize from an application perspective (**ALOM**¹):**
 - **Identify** application level nodes and their placement.
- 3. Conceptualize the **Logical** Operational Model (**LOM**):**
 - **Determine** where all of the logical application components are deployed.
 - **Determine** what logical **technical** components are required and where they will be deployed to support the logical application components.
- 4. Transform into the **Physical** Operational Model (**POM**)**
 - **Determine** real technologies to deliver all of these specifications.

1M transactions per hour

Finance

Warehouse Management

Customer Relationship

ESB

Load balancer

Database

SAP

Message Broker

F5 BIG-IP LTM-4200v

DB2

Application

Logical

Physical

¹*Application Logical Operational Model (ALOM) is not discussed in this lecture – see the Operational Modelling class for details on ALOM*

The LOM documents the placement and interaction of the application's DUs onto nodes distributed across the system's locations



The standard technique involves:

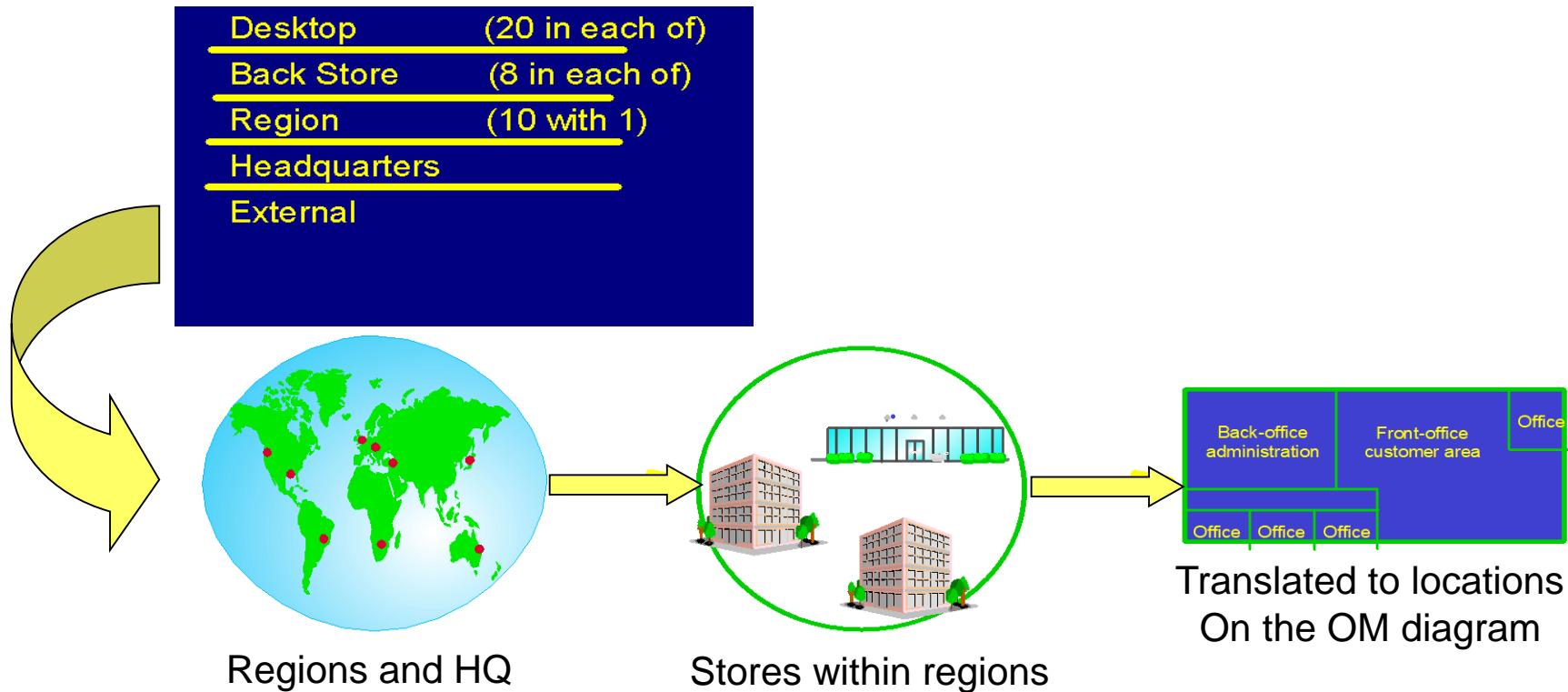
- Identifying potential locations and nodes where deployment units might end up being placed, and establishing where the system's actors will be located
- Placing deployment units onto application logical nodes in locations, based on a variety of considerations such as service management
 - Presentation DUs
 - Data DUs
 - Execution DUs
 - Installation DUs
- We derive connections from the relationships (interactions) between deployment units
- (Confirming the design using architecturally significant walk-throughs)



Step 1: The overall structure of the LOM is largely based on the locations over which the solution will be deployed.

The location is based on the information found in:

- Business Roles & Locations
- System Context
- Current IT Environment





Step 2: Then identify Candidate Logical Nodes

Inputs

System Context

Use Cases

NFRs

Component Model

Key Questions to ask for node discovery

- Is there different functionality or placement across locations?
- Are applications accessed by actors in different locations?
- Are there groups of applications with different NFRs/SLAs?

These are NOT logical nodes

Hardware devices e.g. routers

Named hardware products

LANs and WANs



Output

| | | |
|-----------------|---------------------------|---------|
| A_Office Worker | LN_Mobile Workstation | Type SS |
| | LN_Shared Storage | Type SS |
| | LN_Identity Management | Type SS |
| | LN_Application Management | Type SS |
| | LN_Word Processing | Type SS |
| | LN_Secure Shared Storage | Type SS |
| A_Traveller | LN_Hand Held Device | |
| | LN_Notebook | |
| A_Security SME | LN_Secure Desktop | |
| | LN_Gateway | |



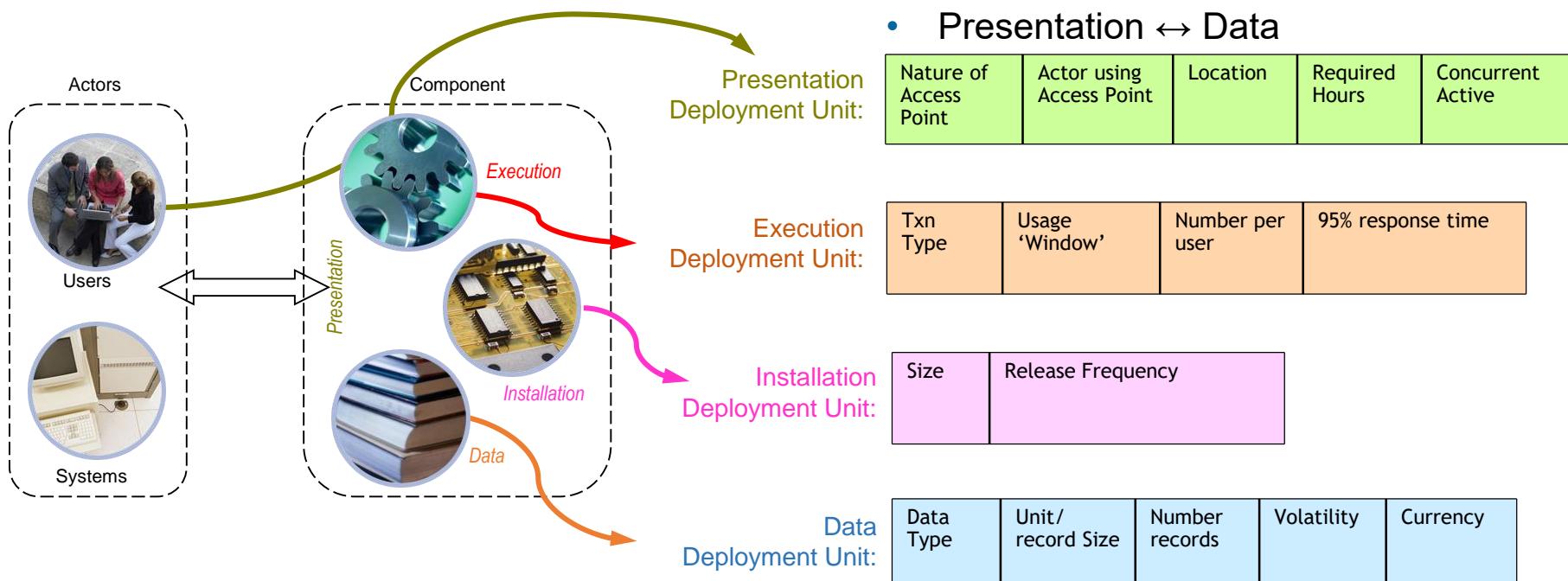


Step 3: We have to understand what it is we need to deploy, via the DEPLOYMENT UNIT Model

- DUs give us a mechanism for:
 - Understanding the nonfunctional requirements placed on the system's components
 - Deciding where best to place the various aspects of the system's components
 - Tracing the system's requirements to its design

The nonfunctional requirements of the component's interactions are described in terms of DU – DU relationships.

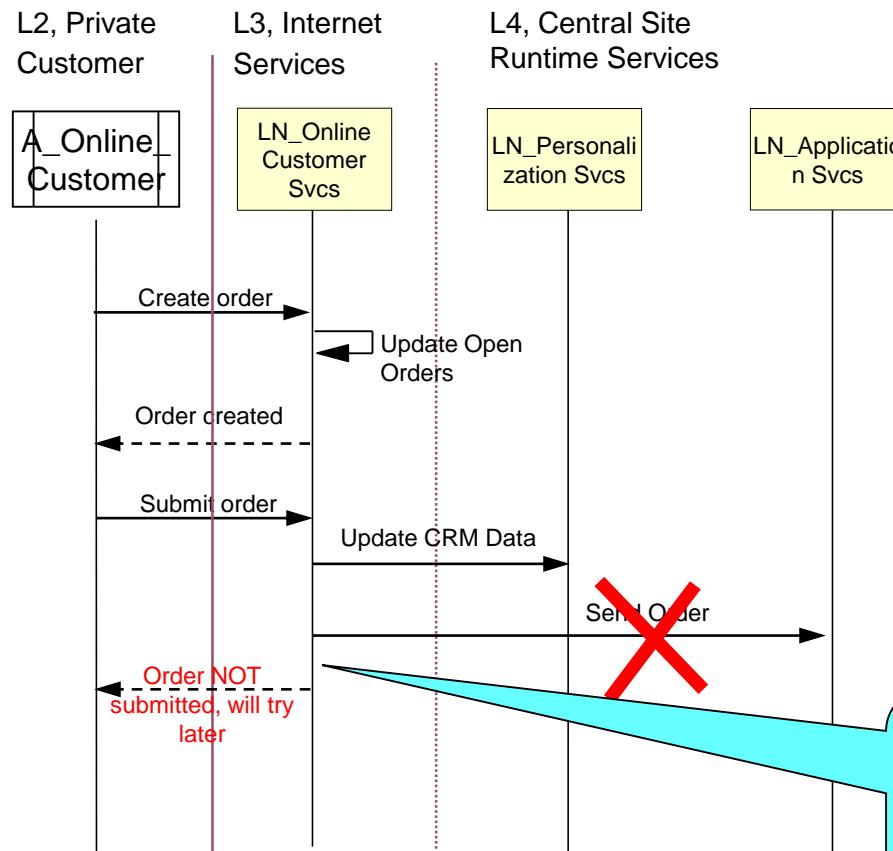
- Typical relationships analyzed include:
 - Presentation ↔ Execution
 - Execution ↔ Execution
 - Execution ↔ Data
 - Presentation ↔ Data



Step 4: Operational Modeling helps ensure the system's non-functional requirements are delivered, within all constraints...



A dynamic walkthrough technique can be used to demonstrate that various Use Cases are indeed supported by the OM and often able to trigger improvements to business processes.



Highlights the need for LN_Online Customer Services to hold a temporary master of the submitted order – NOT shown in the OM relationship diagram!



Time for a question



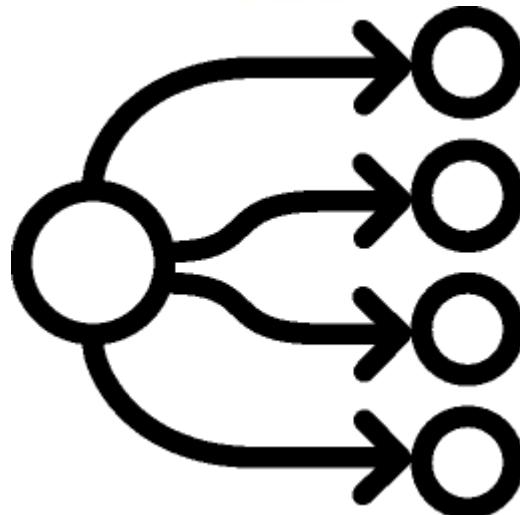
The typical views of an Operational Model include the Physical Operational Model, and the:

- A. Logical Operational Model
- B. Component Operational Model
- C. Technical Operation Model
- D. Deployment Operation Model



Example:

Where and how do we “deploy” a Word Processor ?





What are the use cases that need to be met ?

Story 1: Sales Manager working from a IBM office creates a statement of work using an IBM approved template.



Story 2: Project Manager while traveling creates a Project Change Request using a IBM approved template.



Story 3: Architect working from a IBM office in the public sector creates a confidential design using a template from an IBM asset that is not allowed to be stored outside of local country



Simple
Right ?
Lets see...

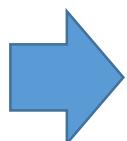
First we model the landscape from a logical perspective and then
from a physical perspective



Logical Examples

Functional Component

Word Processor

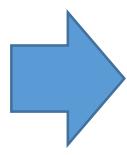


Physical Examples

MS Word 2016

Technical Component

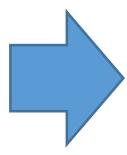
Identity Management



MS Active Directory

Location

Employee Office



IBM Head Office

Deployment Unit

Data: SOW Template



IBM Master SOW
Template.docx

Node

Mobile Workstation



Macbook

Word Processor Example - Logical Location Model



LL_Office (5)



LL_Cloud (1..n)



LL_Remote (10M)



LL_Secure (300)



LL_Secure DMZ (1)



Word Processor Example - Logical Node Model



A_Office Worker

LN_Mobile Workstation



A_Traveller

LN_Hand Held Device

LN_Notebook



A_Security SME

LN_Secure Desktop

LN_Shared Storage

LN_Identity Management

LN_Application Management

LN_Word Processing

LN_Secure Shared Storage

LN_Gateway



Everything on the PC...

Option 1



In this case we have 16 options in minimum to place DUs. In general this is:
 $(\#components * \#locations)^{\#DUs}$

Option 2



Presentation
Execution

Data

How do we do this for complex systems in a structured manner?

Operational Modeling

Option 3



Presentation
Data

Execution
Installation

Word, running on Citrix, with local data



Presentation



Execution
Data
Installation

Word, running in thin client with remote file serving



Based on our use cases we get these deployment units?

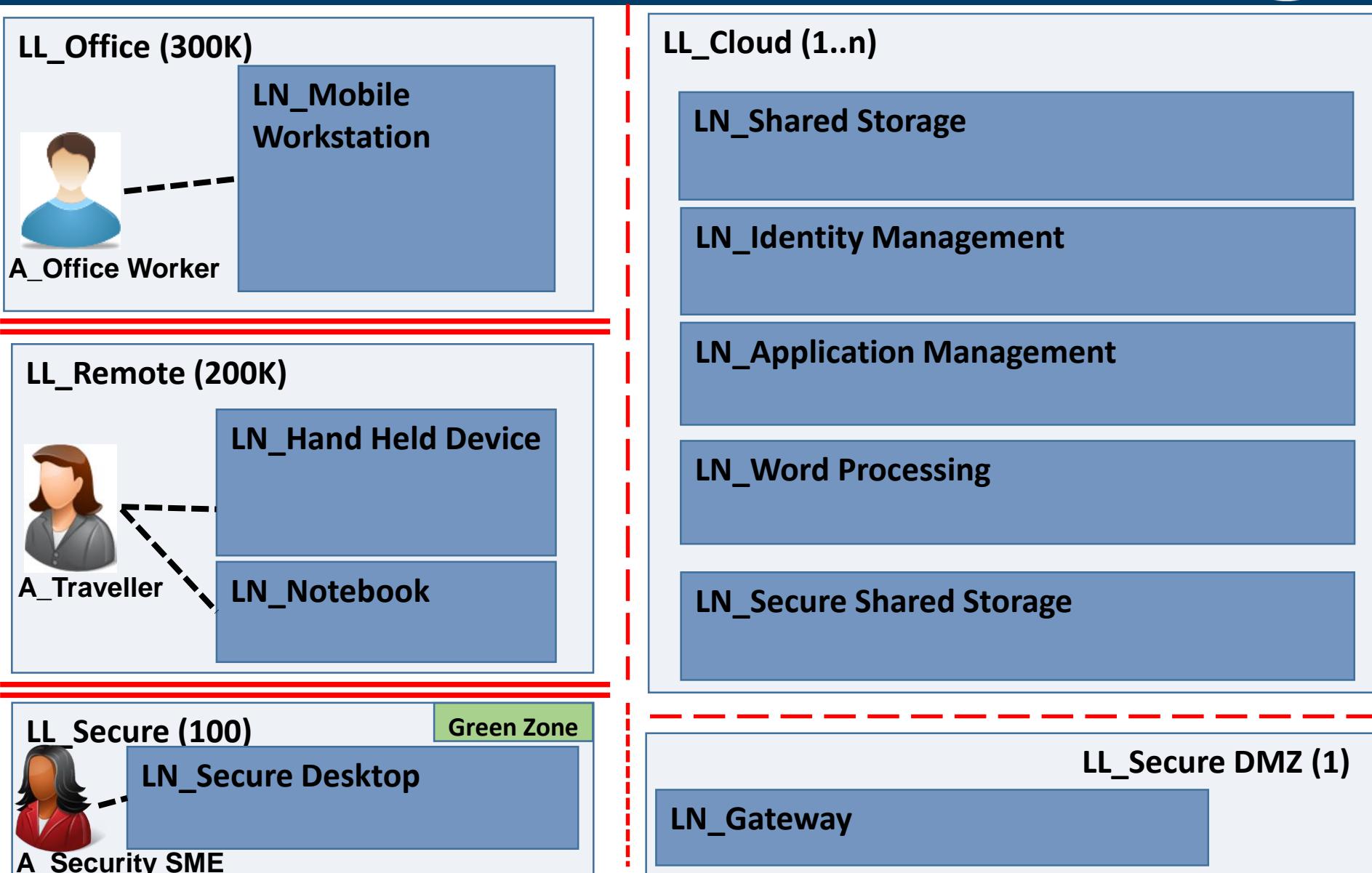
Deployment Units

Word Processor

| Presentation | Data | Execution |
|------------------|----------------------------------|---------------------|
| U_offline | D_corporate_doc_templates | E_word_processor |
| U_online | D_sensitive_client_doc_templates | E_access_management |
| | D_user_profile | |
| Installation | | |
| I_word_processor | | |

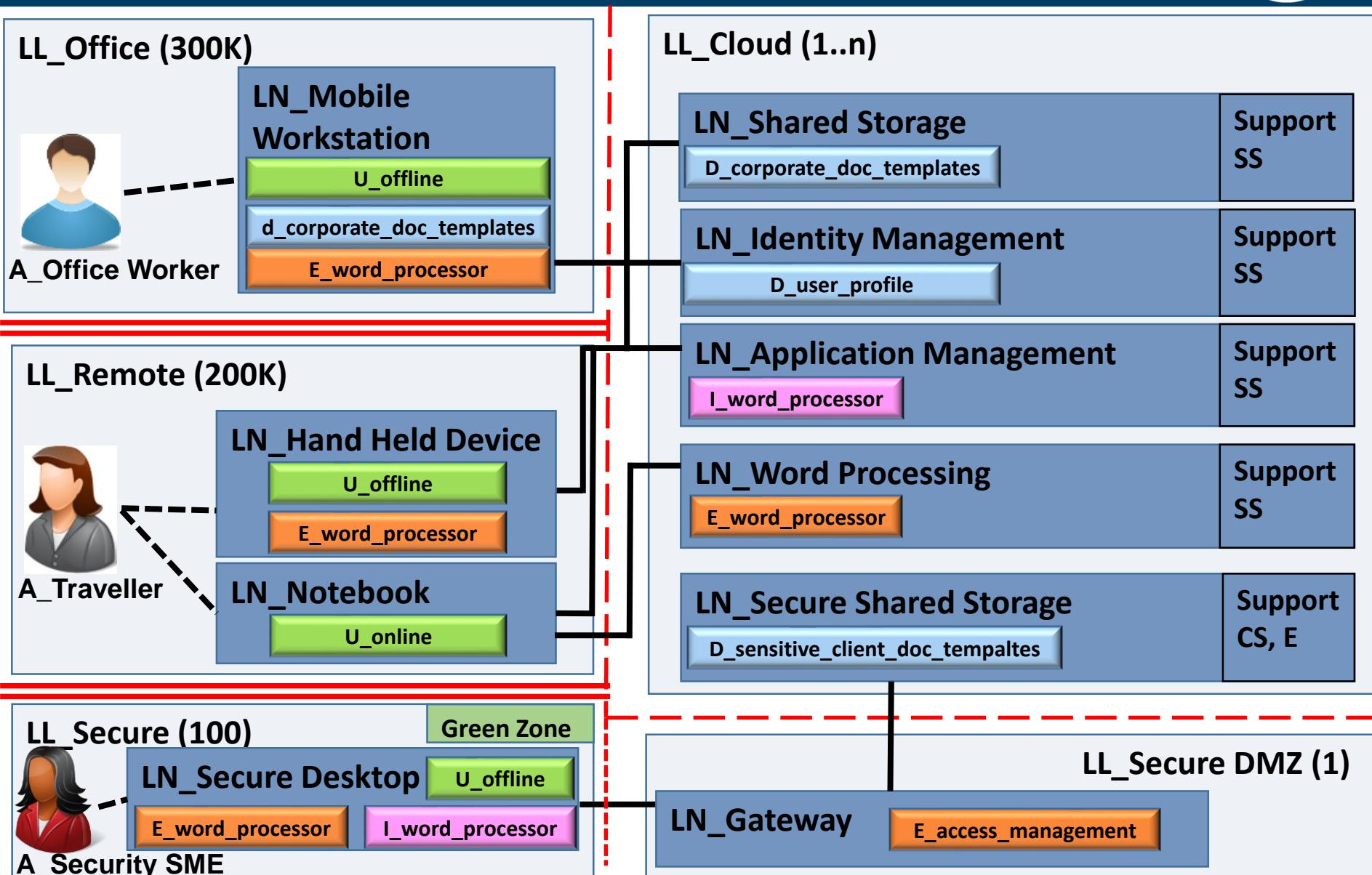
More
things to
consider

Worked Example – Word Processor Logical Operational Model





Worked Example – Word Processor Logical Operational Model





BREAK



Topic outline

- 1. Understand the non-functional requirements**
 - What NFRs need to be met ?
- 2. Conceptualize from an application perspective (**ALOM**¹):**
 - **Identify** application level nodes and their placement.
- 3. Conceptualize the **Logical** Operational Model (**LOM**):**
 - **Determine** where all of the logical application components are deployed.
 - **Determine** what logical **technical** components are required and where they will be deployed to support the logical application components.
- 4. Transform into the **Physical** Operational Model (**POM**)**
 - **Determine** real technologies to deliver all of these specifications.

1M transactions per hour

Finance

Warehouse Management

Customer Relationship

ESB

Load balancer

Database

SAP

Message Broker

F5 BIG-IP LTM-4200v

DB2

Application

Logical

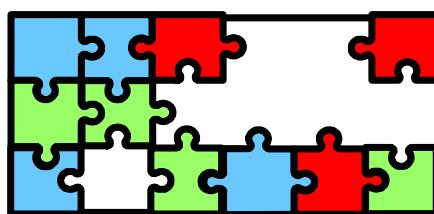
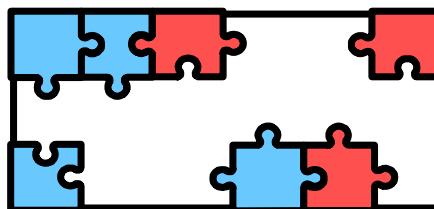
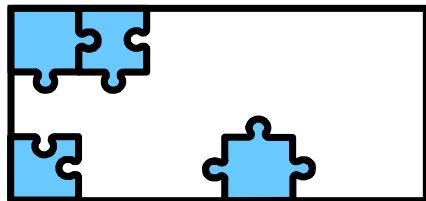
Physical

¹*Application Logical Operational Model (ALOM) is not discussed in this lecture – see the Operational Modelling class for details on ALOM*



The standard technique involves four steps:

- Identifying your role in the product selection process and the definition of the selection criteria
- Performing the necessary product selections for the nodes and connections
- Defining the solution details and describing how technologies will be put together to deliver the specific SLAs
- Rationalizing the solution architecture to get to a sized POM



✓ **1. Identify the “Givens”**

- In a client engagement
 - Usually defined in the client's IT standards, existing Technical Architectures or current IT environment. In all cases, from the beginning you will have ensured the LOM is capable of being implemented in these technologies
- In an IBM proposal
 - IBM product, *if* they meet the requirements!

✓ **2. Spot the “Obvious” products**

- “Easy to choose”, such as:
 - “Only one product/vendor meets the specs.”
 - “We all *know* this will be product/vendor ABC”
 - “We *could* do a full vendor evaluation study, but it is more efficient to choose this product/vendor now”

•

•

✓ **3. Look at the “Adjacent” components**

- The ones that are the most difficult to choose



What is captured in a POM?

Physical Nodes

- Hardware technologies and products
 - Overall hardware configuration
 - Hardware specifications, such as processor speed, memory, disk configuration, etc.
 - for example, SystemP, SystemX...
- Software technologies and products
 - Software product specifications for operating system, database, middleware, directory services, security, etc., including versions
 - for example AIX Version X.x, WebSphere Application Server Liberty Profile, etc...
 - Detailed configuration of software, such as the need for multiple instances of a software product on a computer

Physical Connections

- Technology specific, including
 - Physical networks, protocols, network bandwidth, latency, etc.
 - for example, 100BaseT, Fiber Optic, T1, T3, 1/10/100 Gbps Ethernet, TCP/IP, etc.



What is captured in a POM?

Cloud Services

- Cloud deployment model
 - **Public:** Shared infrastructure that lowers the cost per unit for all customers.
 - **Private:** Provides dedicated infrastructure configured for flexibility and scalability.
 - **Managed:** Takes care of infrastructure management and maintenance tasks.
 - **Hosted:** Provides flexibility to self source infrastructure management and maintenance support.
 - **Hybrid:** Mix of using shared infrastructure and dedicated infrastructure depending on NFR's.
- Cloud services model
 - **IaaS** : Infrastructure as a Service - provides raw infrastructure resources in a virtualised environment e.g. IBM Bluemix.
 - **PaaS:** Platform as a Service - provides an container/runtime environment with supporting services to build solutions e.g. IBM Bluemix.
 - **SaaS:** Software as a Service – provides application functionality as a service e.g. IBM Verse.
 - **BPaaS:** Business Process as a Service – provides a means to increase organization efficacy by outsourcing business processes to a automated runtime e.g. through the use of IBM Blueworks Live



Traditional LOM to POM transformation

A distinction can be drawn between Physical Level Operational Models that emphasize



A distinction can be drawn between Physical Level Operational Models that emphasize:

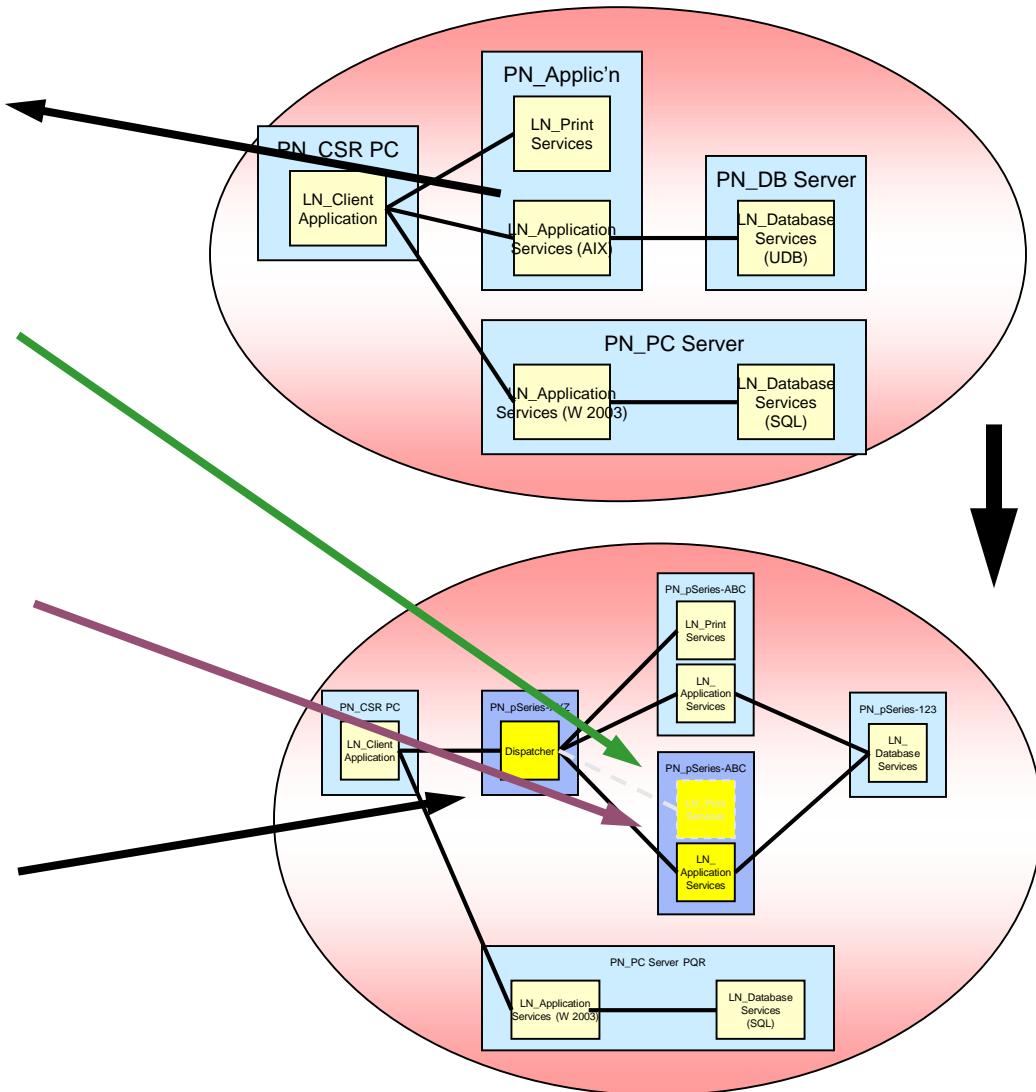
- Component Selection (Unsized POM): Identifying the technologies and products, and the overall (qualitative) way in which they will be put together:
 - Identify the “givens”
 - Spot the “obvious” products
 - Look at the “adjacent” components
- Configuration (Sized POM): Defining the (quantitative) details of the solution, including the way these technologies will be put together in order to deliver the specific SLRs of the OM’s specification:
 - Performance and capacity
 - Availability
 - Security
 - Service management

Rationalizing the design involves a process of SIZING the POM,
so that performance & other characteristics can be achieved.



For example, the “PN_Applic’n” physical node will deliver:

- The availability requirements of the Print Services, by being implemented as two physical pSeries machines, with a failover capability between the “live” and “standby” machines
- The throughput requirements of Application Services, by being implemented as TWO physical pSeries machines, together with a “dispatcher” physical node.



This, of course, has an influence back on the POM, through the introduction of a “LN_Dispatcher” logical node



Cloud LOM to POM transformation



- 1. Plan** - Review the Cloud reference architectures and determine if one is a good match for the NFR requirements or tailor as required.
- 2. Define** - the type of cloud deployment model (CDM) i.e. public, private, hybrid, that meets the solution NFR's and constraints.
 - Logical Service Models e.g. IaaS, PaaS, SaaS, BPaaS
 - Logical Services e.g. Middleware, Database, Storage
- 3. Select** – a cloud provider that meets NFR and CDM requirements with a sizing/pricing model that provides the best match for the solution.
- 4. Configure** – Using NFR's, LOM and cloud reference architecture as inputs
 - Configure the required physical cloud services offered within the chosen providers CDM.
 - Configure performance, capacity, and availability by configuring physical cloud services sizing/service level requirements (SLR's)
 - Understand licensing model including enterprise level agreements for software/services installed within IaaS and PaaS.



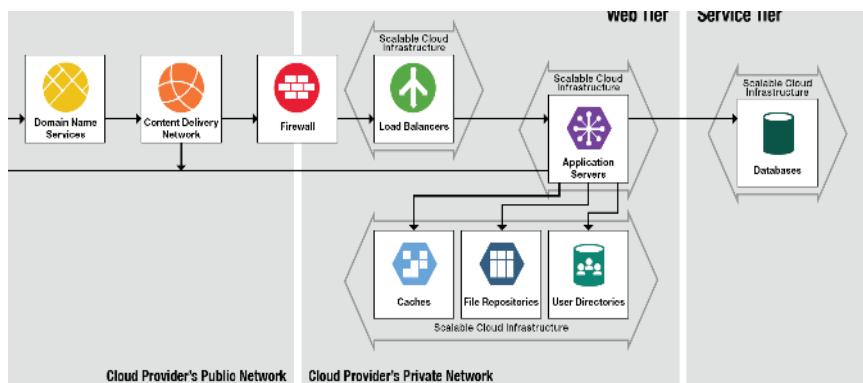


Cloud@POM – Plan what kind of cloud is required for the solution

Selecting a cloud reference architecture helps leverage lessons learnt and best practices but you may still need to tailor to your needs or in the case none fit the solution define your own

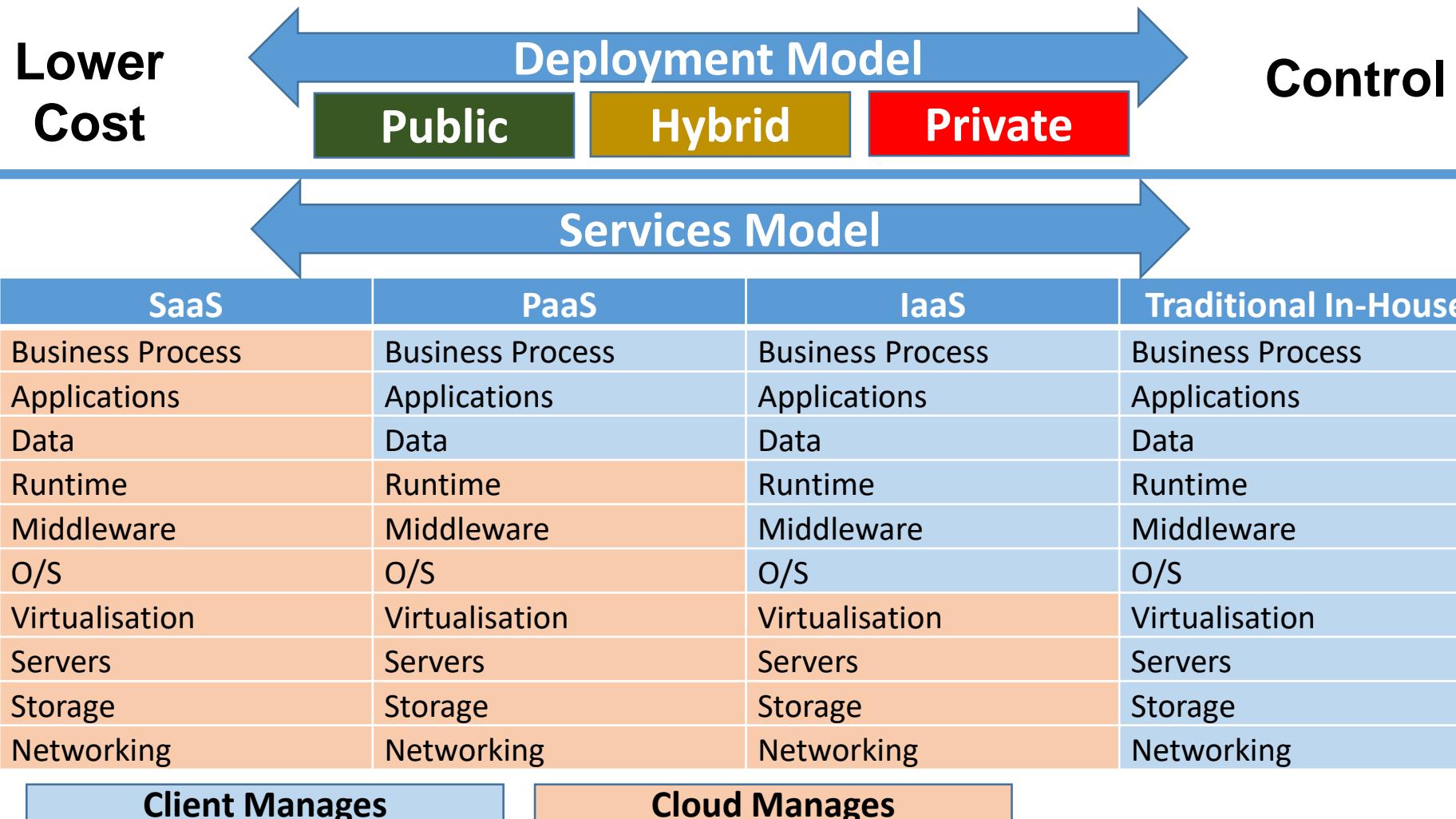
Cloud Council Reference Architectures

- [Cloud Customer Architecture for Securing Workloads on Cloud Services](#)
- [Cloud Customer Architecture for API Management](#)
- [Cloud Customer Architecture for Hybrid Integration](#)
- [Cloud Customer Architecture for Enterprise Social Collaboration](#)
- [Cloud Customer Architecture for e-Commerce](#)
- [Cloud Customer Architecture for IoT](#)
- [Cloud Customer Architecture for Big Data and Analytics](#)
- [Cloud Customer Architecture for Mobile](#)
- [Cloud Customer Architecture for Web Application Hosting](#)





Architectural decisions need to be made to select the required deployment and service model(s) based on the NFR's and constraints while balancing value for the Customer.





- Some considerations include:
 - Availability of the cloud platform within the client's jurisdiction.
 - Enterprise strength of the services offered by cloud providers, in particular security, availability, backup and disaster recovery.
 - The level of support offered by the cloud provider
 - Network connectivity to the cloud provider's data centre. One of the disadvantages of a cloud based solution is the potential for increased latency.
 - Content of the cloud provider's service catalogue (e.g. operating system, databases, middleware, ect).



hp Cloud Services
Many others!





1. Need to select a physical service for each logical services identified in the LOM i.e. Database -> DashDB
2. Define the size of the cloud services and as you can see the major cloud providers use a T-shirt sizing model and you need to consider the NFR's to select the "T-shirt" the best fits the solution while keeping customer value in mind

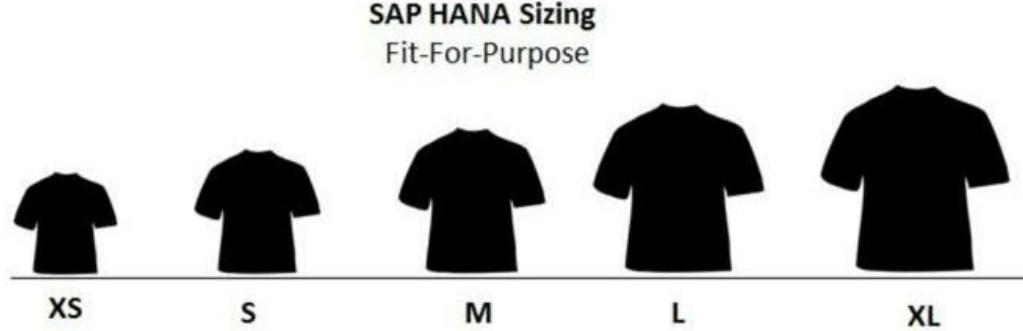
Microsoft Example

| SKU Family | ACU/Core |
|-------------------|----------|
| ExtraSmall | 50 |
| Small-ExtraLarge | 100 |
| A5-7 | 100 |
| Standard_A1-8v2 | 100 |
| Standard_A2m-8mv2 | 100 |
| A8-A11 | 225* |
| D1-14 | 160 |

IBM Example

| T-Shirt | vCPU | RAM (GB) | HD (GB) | Price/Hr |
|---------|------|----------|---------|----------|
| S | 1 | 2 | 12 | \$0.21 |
| M | 2 | 4 | 25 | \$0.42 |
| L | 4 | 8 | 50 | \$0.84 |
| XL | 8 | 16 | 100 | \$1.68 |
| XXL | 16 | 32 | 200 | \$3.36 |

SAP Example



The review process leverages transaction walkthroughs to test the solution's robustness



At every level of the OM, “transaction walkthroughs” using the OM walkthrough diagram are a powerful means of verifying whether the OM will work

- Analyze each transaction from start to finish following the flow of information through each physical node and connection
- Visualize what would happen if there was a failure on some part of the system:
 - Would the failure be detected?
 - Would the end user be notified of the failure?
 - How will the transactions and data be recovered?
 - Who will be responsible for repair of the component?
- Analyze how the failure would impact Service Level Agreements



Module outline

Introduction and objectives

What is Operational Modeling (OM) and why do we need it?

How does OM evolve and how is it represented?

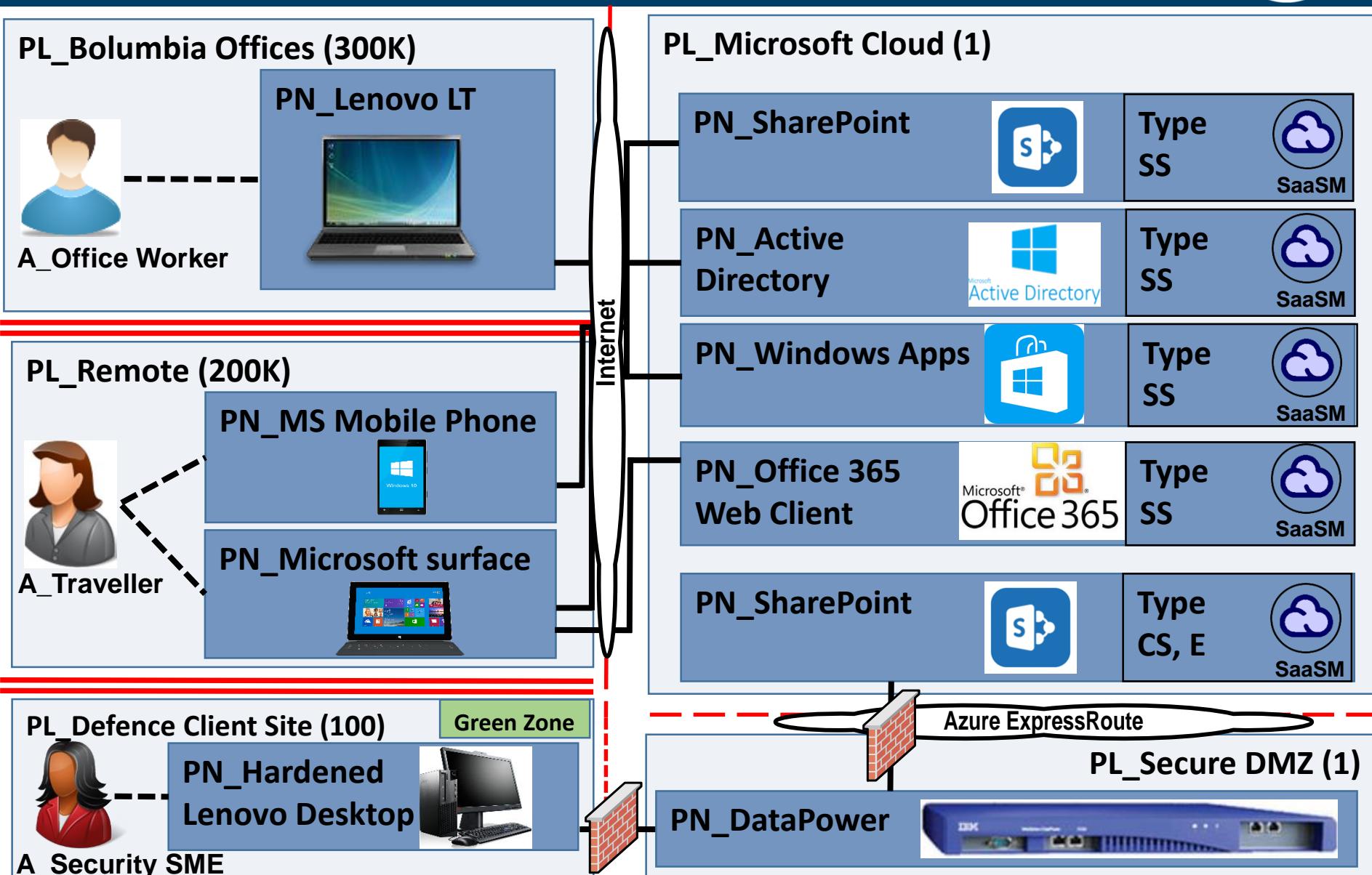
► **Constructing an Operational Model**

Summary and references



Lets transform our LOM into a POM

Worked Example – Word Processor Physical Operational Model





Module outline

Introduction and objectives

What is Operational Modeling (OM) and why do we need it?

How does OM evolve and how is it represented?

Constructing an Operational Model

► **Summary and references**



The operational aspect of architecture:

- Focuses on the placement of the solution's DUs across the organization to achieve the business requirements of performance, availability, and other nonfunctional requirements
- Is addressed through the creation of the Operational Model at 2 different viewpoints:
 - The Logical Operational Model, which focuses on the distribution style of the application and the technical components required to support the application components
 - The Physical Operational Model, which details which combination of products, hardware and or cloud based services would deliver on the specification defined through the LOM



References

Enterprise Solutions Structure – ESS

- http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5387080

Cloud Computing

- <https://www.ibm.com/developerworks/learn/cloud/index.html>

Cloud council

- <http://www.cloud-council.org/>

THANK YOU



IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "[Copyright and trademark information](#).“ Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and its affiliates.

Linux is a trademark of Linus Torvalds in the United States, or other countries, or both. Microsoft, Excel, PowerPoint, Project, Visio, Word, and Windows are trademarks of the Microsoft Corporation in the United States, other countries, or both. UML is a registered trademark of Object Management Group, Inc. in the United States and/or other countries. TOGAF and UNIX are registered trademarks of The Open Group in the United States and other countries. Other company, product, and service names may be trademarks of IBM or other companies. Citrix is a trademark of Citrix Systems, Inc. and one or more of its subsidiaries, and may be registered in the United States Patent and Trademark Office and in other countries.