



BITS Pilani

Software Architecture

Architecting for the Cloud

Viswanathan Hariharan

Contents



- Assignment 1 & 1
- Recap
- Architecting for Cloud

Assignment



A1 - Build an Architecture for an App

- The App should at minimum include the technologies Web, Mobile, IOT, cloud and analytics.
- Each team will select an application.
- The submitted architecture will be evaluated by the TA
- The team will be evaluated for their developed architecture

A2 – Research paper on real life architecture / latest trends etc

- Each team has to select a topic
- Final Paper should be submitted as per the agreed upon template and schedule

Refer to the template for Assignment #1

Assignment - 2 Sampe



1. Performance - at least 3 popular applications
2. Scalability - at least 3 popular applications
3. Facebook - scalability
4. Agile and Architecture
5. Micro Services
6. Uber / Ola etc
7. Architecture For IOT
8. Architecture For Mobile
9. Architecture For Machine Learning
10. Architecture for Conferencing Platform
11. AWS Lambda
12. Architecture and NoSql DB
13. Architecting for cloud - trends
14. Compare cloud - Amazon / Azure / Google
15. YouTube / Netflix

Guidelines

- **You are not building an architecture**
- **You are studying what is happening around**
- **You must read many papers / case studies**
- **You can do a Literature survey paper**
- **Key words**
 - Study
 - Analyze strengths / weakness
 - Compare
 - Trend - what is driving the trends
- **Abstract**
- **Intro**
- **Scope / objective**
- **Contents**
- **References**

Assignment - 2 Guidelines



Guidelines

- You are not building an architecture
- You are studying what is happening around
- You must read many papers / case studies
- You can do a Literature survey paper
- Key words
 - Study
 - Analyze strengths / weakness
 - Compare
 - Trend - what is driving the trends

Guidelines for the paper

- Intro
- Scope / objective
- Contents
- References

Evolution of SW Architecture



We design and implement information systems to solve problems and process data.

As problems become larger and more complex and data becomes more voluminous, so do the associated information systems

- Structured programming, Data Structure, Higher Level languages, software engineering, Object Oriented etc

Computing become Distributed, on the cloud, Mobile as a front end

As the problem size and complexity increase, algorithms and data structures become less important than getting the **right structure** for the information system.

Specifying the right structure of the information system becomes a critical design problem itself

< Example from Construction Industry >

Importance of Quality attributes & Tactics



- Functional requirements help us to define the modules
- Quality attributes help us to **structure** the system

Availability

Modifiability

Performance

Security

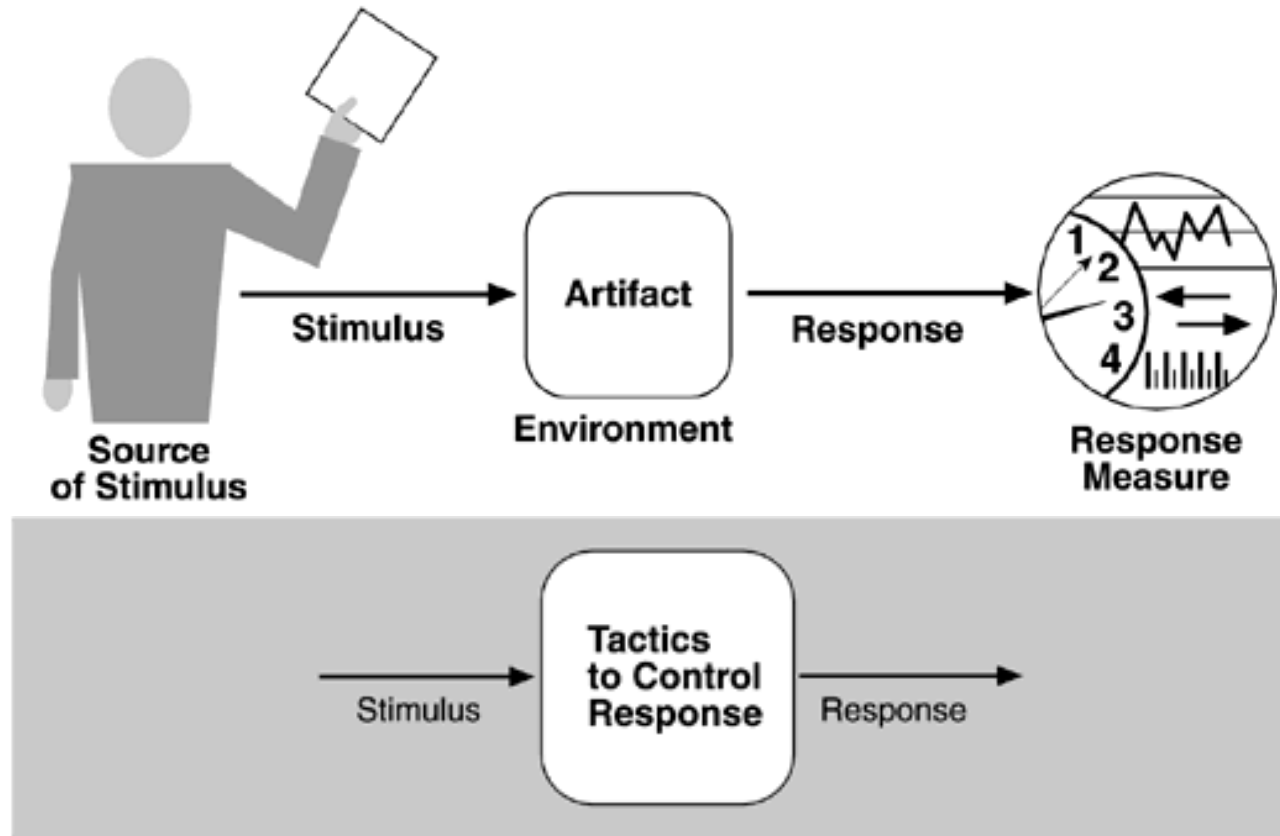
Usability

Interoperability

Scalability

Testability

Architecturally
significant
requirements



What is a Pattern



1. Addresses a recurring design problem
2. Documents existing, well proven design experience
3. Pattern identify and specify abstractions that are above the level of single classes and instances or of components
Typically, a pattern describes several components, classes or objects, and details their **responsibilities** and relationships, as well as their **cooperation**.
All components together solve the problem more effectively than the pattern addresses

Because patterns are (by definition) found repeatedly in practice, one does not invent them; one discovers them.

Patterns



1. Layer
2. Pipe & Filter
3. MVC
4. Publish & Subscribe
5. Client & Server
6. P2P
7. Shared Data
8. **Broker**
9. Map-Reduce
10. Multi-tier
11. SOA

Reference Architecture - Microsoft, AWS, IBM

Attribute-Driven Design



Purpose , Quality Attributes , Functionality, Architectural concerns , Constraints

1. Review Inputs
2. Establish iteration Goal - Selecting drivers
3. Choose one or more elements of system to refine
4. Choose **design concept(s)** that satisfy the drivers
 - Proven methods, reference architecture, Patterns, tactics, externally developed components - products (SQL DB), library, framework, Platform (java, .net), Past experience, Best Practices, Expert Knowledge
 - For a specific problem, one will combine different types of design concepts
 - Making decision is a Process, Make Candidate Decisions then refine

Architecting



1. Judicial system Case Study for building the Architecture
2. Documenting the Architecture
3. ATAM - Tradeoff Analysis Method

Architecting for the Cloud



Intro to Cloud


- Definition
 - Virtualization
 - Architecting for cloud - what is the difference
 - Multi Tenancy
-


Cloud Definition



NIST

On-site	IaaS	PaaS	SaaS
Applications	Applications	Applications	Applications
Data	Data	Data	Data
Runtime	Runtime	Runtime	Runtime
Middleware	Middleware	Middleware	Middleware
O/S	O/S	O/S	O/S
Virtualization	Virtualization	Virtualization	Virtualization
Servers	Servers	Servers	Servers
Storage	Storage	Storage	Storage
Networking	Networking	Networking	Networking

 You manage

 Service provider manages

Cloud Definition






NIST



Cloud Comparison

Key benefits & drawbacks of cloud computing types

 Public Cloud	 Private Cloud	 Hybrid Cloud
No maintenance costs	Dedicated and secure	Policy-driven deployment
High scalability, flexibility	Regulation compliant	High scalability, flexibility
Reduced complexity	Customizable	Minimal security risks
Flexible pricing	High scalability	Workload diversity supports high reliability
Agile for innovation	Efficient	Improved security
Potential for high TCO	Expensive with high TCO	Potential for high TCO
Decreased security and availability	Minimal mobile access	Compatibility and integration
Minimal control	Limiting infrastructure	Added complexity

Why Cloud



- Soon a corporate with a 'no-cloud' policy will be as rare as a 'no-Internet' policy is today.
- Cloud is a Vehicle for agile, scalable and elastic solutions
- Cloud: Cost scales with use and enables deferred spending

Core Technology - Virtualization

Virtualization #1

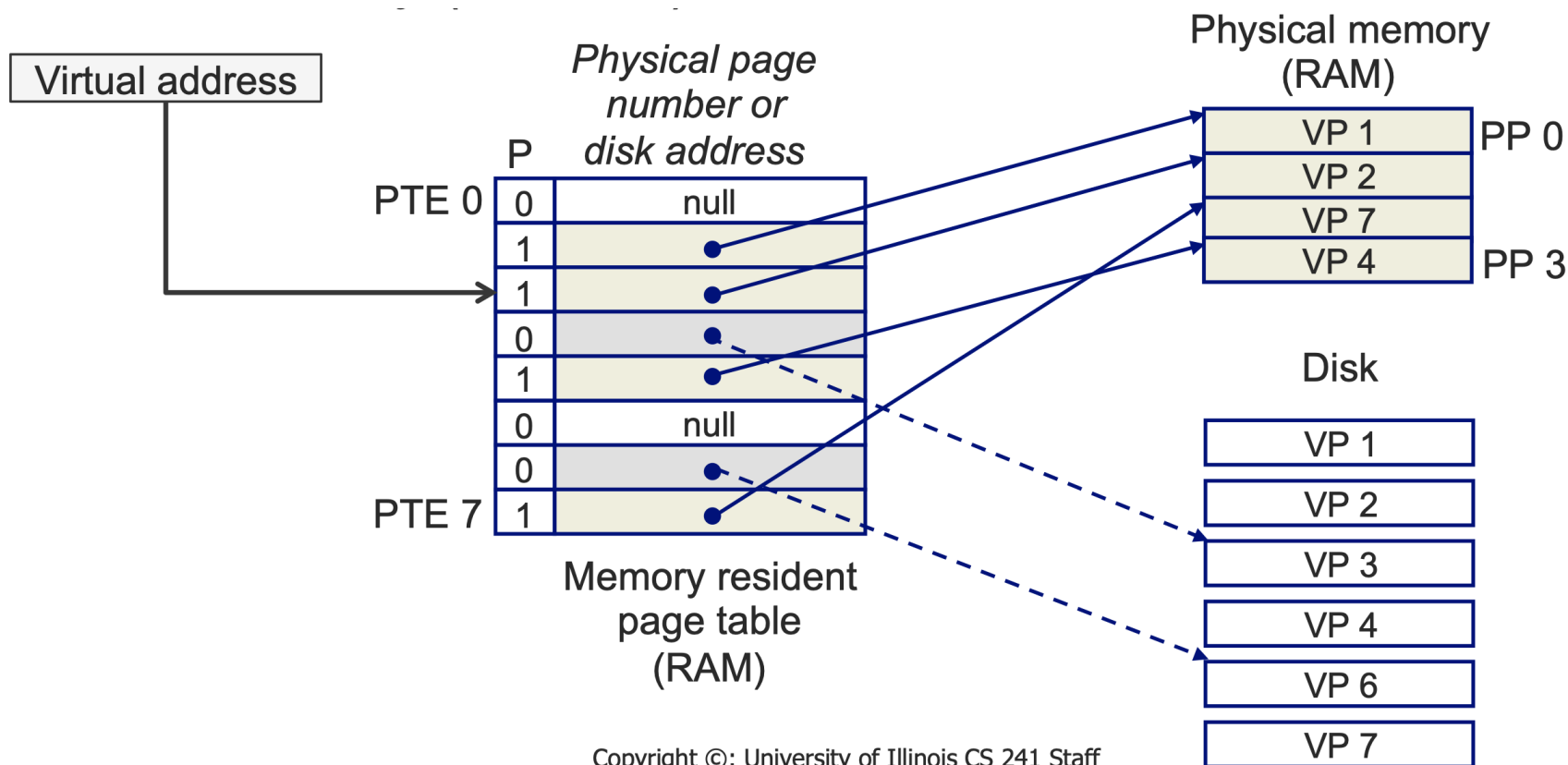


Java Virtual Machine ??

Virtualization #2



Virtual Memory Management with Page Tables



Virtualization #3



Multi Programming



Virtualization #4



Think of Multi Programming with Memory Management

Virtualization #5



Hardware Abstractions - Computer with various configurations

Virtualization #6



A virtual machine consists of several files that are stored on a storage device. The key files are the configuration file, virtual disk file, NVRAM setting file, and log file

Both Data and Executable code - OS & VM tools (Device drivers etc)

It is a Software Computer -

Roughly equivalent an OS Image in a computer...

When instantiated on a physical Hardware it behaves like a computer on which One can run applications...
