SaaS
(Software as a Service) On Cloud

Piyush Aghera and Sanjay Chaudhary

Outline

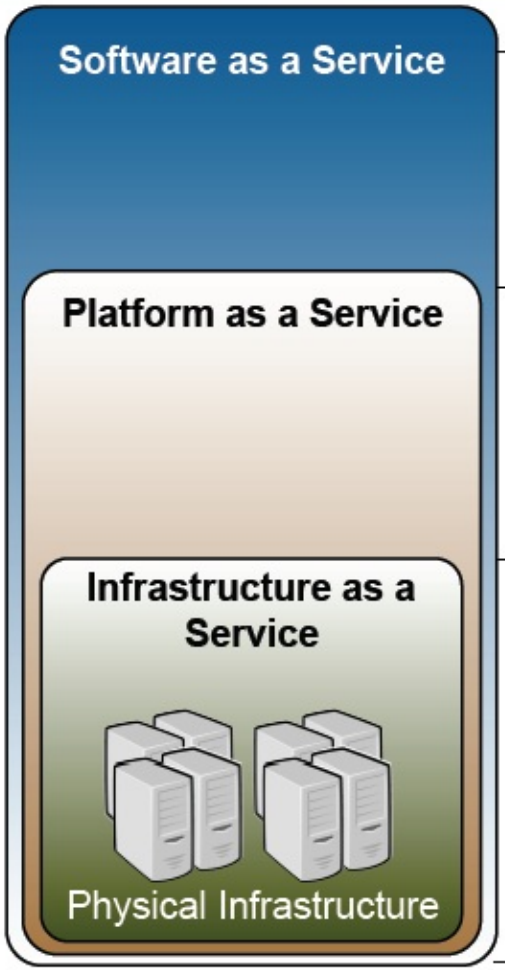
- Introduction
- Cloud Service Model
- Software as a Service (SaaS)
- Hosting vs SaaS
- SaaS Core functionality
- Product vs SaaS
- Myths of SaaS
- Domains that are more suitable for SaaS

Introduction

- What is Software ?

- What is Service ?

Cloud Service Models



	What	Who
Software as a Service	On-demand access to any application	End-user (does not care about hw or sw)
Platform as a Service	Platform for building and delivering web applications	Developer (no managing of the underlying hw & swlayers)
Infrastructure as a Service	*Raw* computer infrastructure	System Administrator (complete management of the computer infrastructure)

*http://www.ibm.com/developerwor/

Current Challenges

- Add new services quickly
- Expand your infrastructure !!
- Uncertainty of data
- Manage complex software and service
- Risk of developing
- Vendor locking

Software as a Service

- Software as a Service (SaaS) model is both a software delivery model and business model .
- The SaaS delivery model involves delivering an application as a service, over the Internet.
- No hardware or software to manage
- Service delivered through a browser
- Not required expert team to maintain software
- Pay as per you used

What is not SaaS ?

- What is not SaaS ?

  If a hosting company hosts 100 separate instances of a software package, there are 100 software instances to install, maintain and (worst of all) customize. This model is simply not scalable .

- Current hosting is not pay as you used. You have to pay whether you are using or not.

- Deliver service to one organize only

Why Hosting Need Changes

- Technology has evolved.
- People demand more control on infrastructure .
- Instant gratification
- In-House too costly in cost and human capital
- Hosting always need to best match service provider
- More rare required service also need to host forever.

Owner vs Hosted vs Cloud [For User View]

	Own Resource	Hosted Resource	Cloud Resource
Time	Week to Month	Day to week	Minute to Hour
Scalability	Slowest ,Rigid & Costly	Some what flexible	Instance-flexible, Pay per usage
Flexibility	Not flexible	Some what flexible	Fully flexible
Cost	Purchase cost	Rent rate on fix time.	Rent time as per usage
Pricing Unit	Buy server whether you use or not	Rent server whether you use or not	Pay iff you are using server

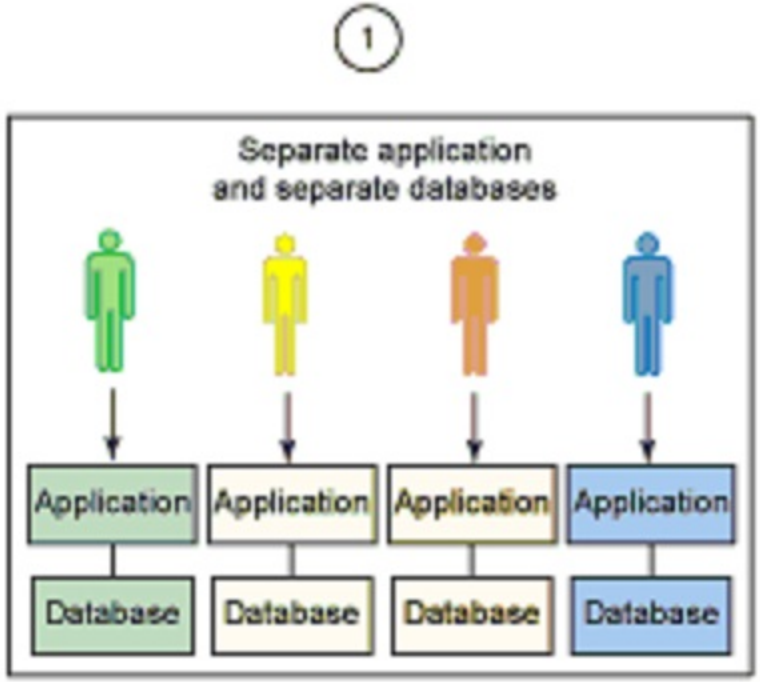
Core functionality that should be in SaaS

- Multi-tenancy
- Subscription
- Monitoring and billing
- Security and privacy
- Scalability, high availability and reliability
- Managing and administration for separate tenant
- Runtime per tenant customization .
- Customizable GUI
- Customizable business logic
- . . .

1. Multi-tenancy

If Not Multi-tenancy :

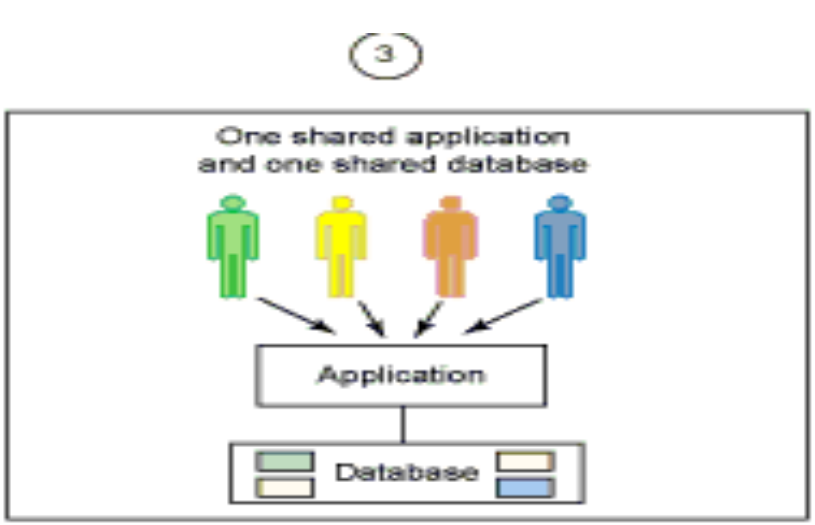
- Organization is an owner of the application and it has to maintain on their infrastructure
- Different Organization running just slightly different application for same purpose
- No Support for Scalability

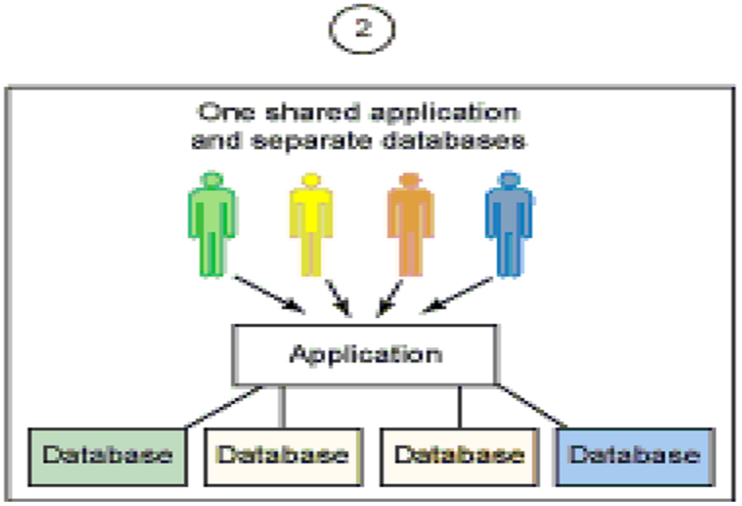


* http://www.ibm.com/developerworks/cloud/library/cl-multitenantsaas/

If Multi-tenancy:

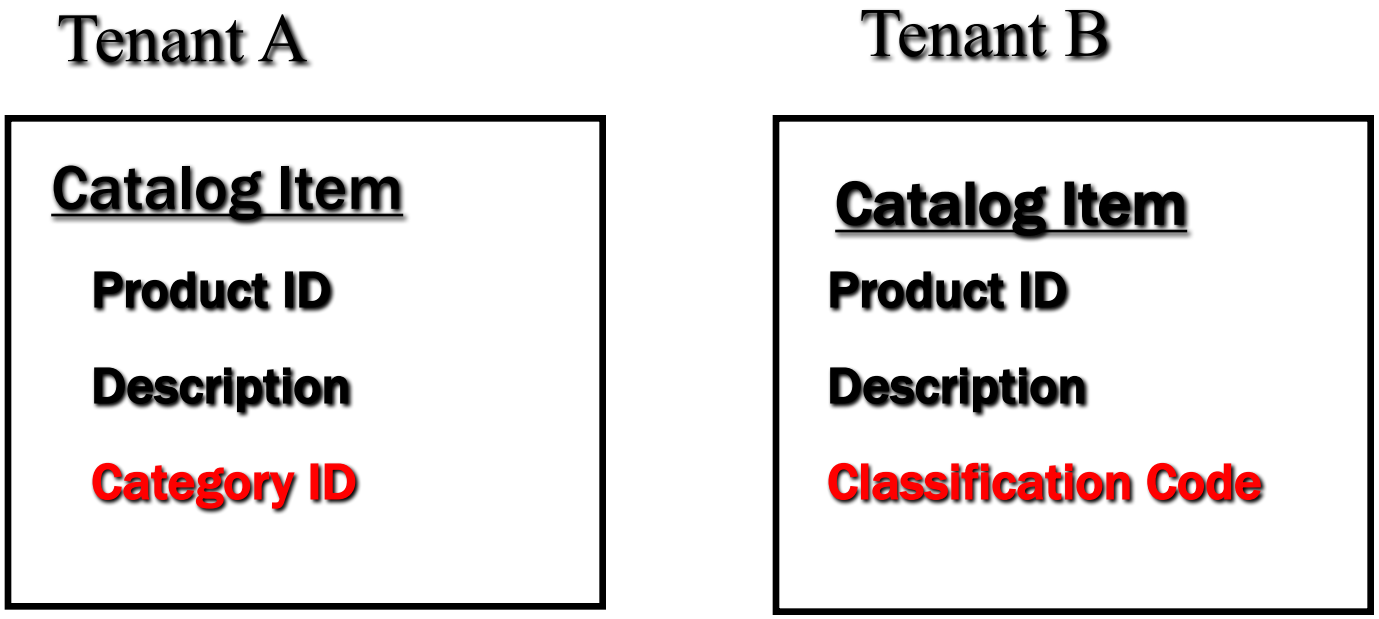
- Multi-tenancy is the key for SaaS efficiency
- SaaS and the cloud, many organizations will be using the application; they must all be able to allow all their users to access it, but the application must allow only each organization's own members to access the data for their organization.





* http://www.ibm.com/developerworks/cloud/library/cl-multitenantsaas/

Data Model Extension



Challenges:

- Defining custom fields and storing custom data for each tenant.
- Business logic that can handle custom fields
- Presentation logic that can handle custom fields

Simple Example

Multi-tenancy

Single Tenant

2. Subscription and billing

- Since SaaS applications by design involve a series of payments based on factors like number of users per tenant, application options, and perhaps usage duration, there must be a way to track and manage the application use and generate billing information that is accessible to the tenant administrators.

* http://www.ibm.com/developerworks/cloud/library/cl-multitenantsaas/

3. Security and privacy

- Most SaaS application architectures take data security measures that prevent one tenant from seeing another tenant's data as a baseline requirement
- Trust on service provider security
- User ID and authentication for tenant

* http://www.ibm.com/developerworks/cloud/library/cl-multitenantsaas/

4. Scalability, High Availability & Reliability

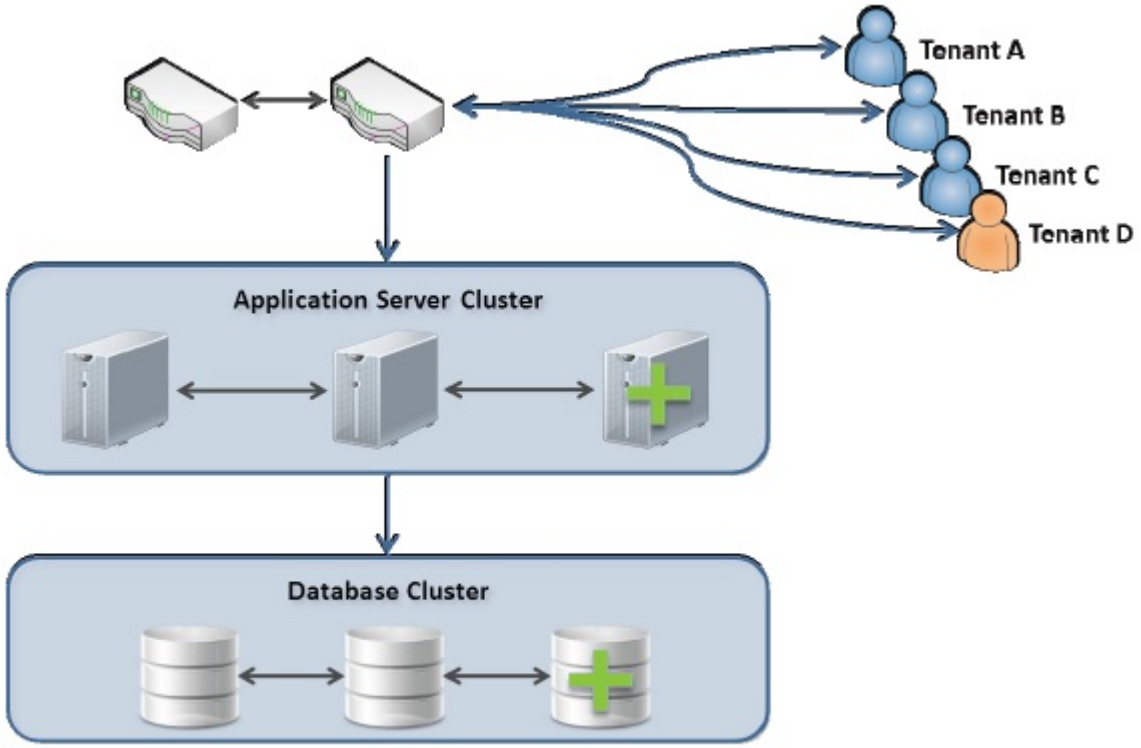


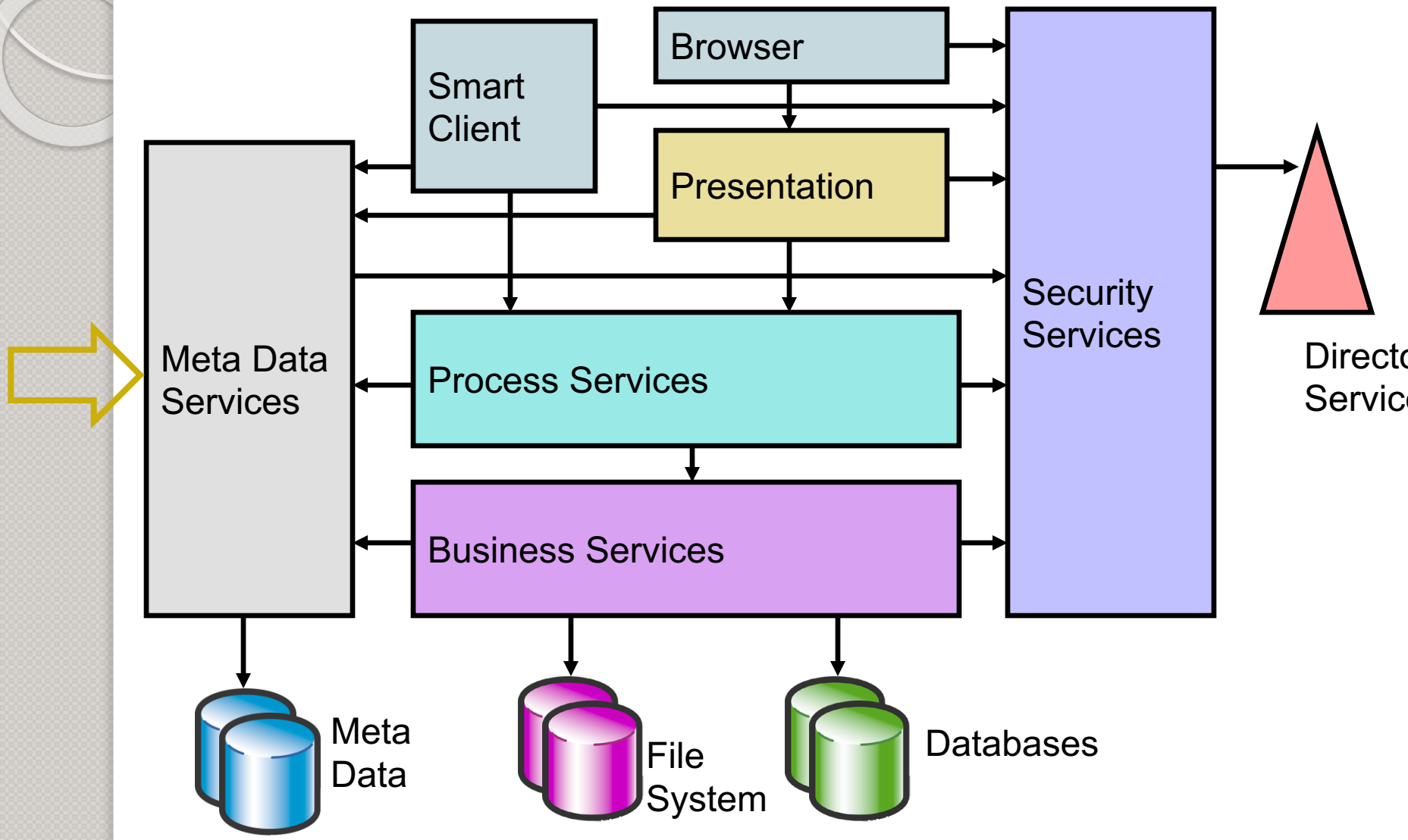
Figure 4. SaaS L4 Scalability & Availability

[4]

Others

- Managing and administration for separate tenants
- Runtime customization per tenant
- Customizable GUI
- Customizable business logic

High Level Application Architecture



Product vs Service [Developer's view]

	Software as a Product	Software as Service
Delivery	Designed for customer to install, manage and maintain	Designed for internet base service
Development	Longer cycle, Multiple coding for same application	Design to run thousands of different customers on single source code.
Pricing	License + maintenance	Subscription (All included)
Additional cost	Installation Maintenance Customization upgrades	Configuration according to changes only
Platform	Multi-version	Single Platform

Product vs Service

	Software as a Product	Software as Service
Add new Feature	Version control & update fee	Fixing a problem for one customer fixes it for everyone
Product demo	Required installation	Don't required installation
Mobility	Not possible	Anywhere, anytime, anyone
Success	New license revenue	No license problem
Updates	Not frequently possible	Shorter time possible
Feedback Cycle	Long	Short

Current SaaS

- Google Service
- Sale force CRM
- DESKAWAY

DESKAWAY
SIMPLIFY WORK

- https://www.deskaway.com/index.php
- https://lab213.deskaway.com/
- http://www.salesforce.com/

Domains that are more suitable for SaaS

- Education institute like College, School, University
- Hospital management
- E-Government agency
- Hotel , Tourism & restaurant management
- Banking System
- Industry having slightly different production
- More others

Risk in SaaS

- Nothing to buy
- Cancel immediately
- Change instantly,even operating systems
- Throw it out
- Rebuild it instantly testing

The complete list of myths on SaaS

- Fewer features
- Customer loses control
- Security is a problem
- Difficult to integrate
- SaaS is Risky
- Hosted is only good for small businesses and projects
- Costs more over time
- Service could be poor with a SaaS
- SaaS companies have an unproven business model

SaaS Challenges

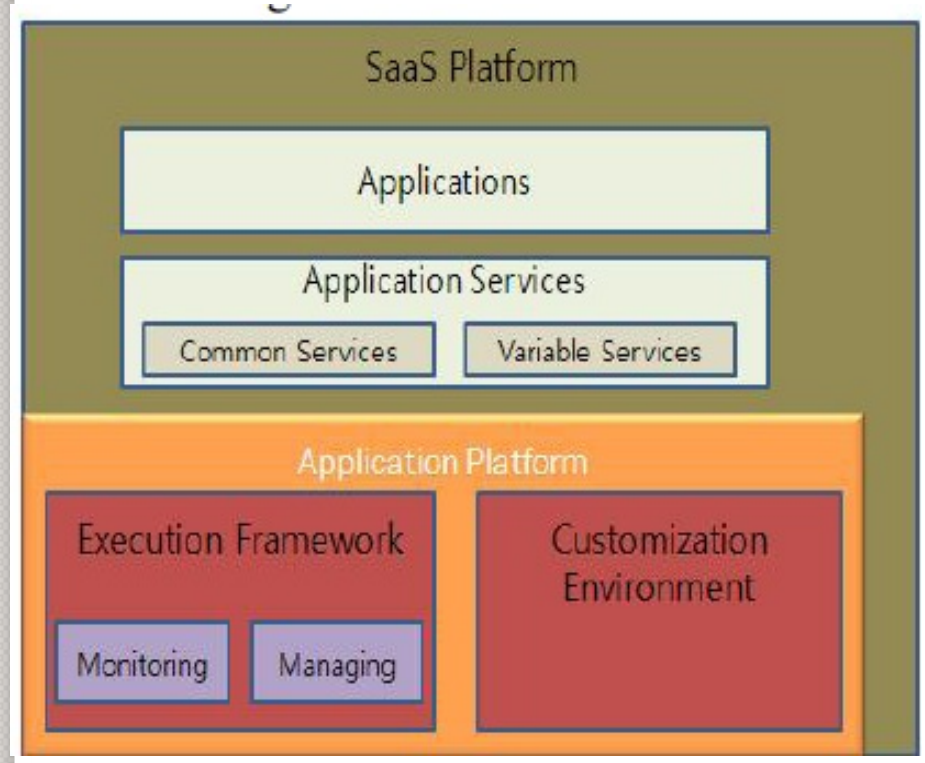
- SaaS development with all feature is very costly and time consuming. Technical expertise is required right from development to deployment on cloud.
- Cloud providers have their proprietary platform on which SaaS applications are deployed hence it is not easy to move a SaaS application from one cloud to another .
- Single tenant failure … it may affect all other services

SaaS Architectures

- Metadata Driven Architecture [15]
- SaaS with tenancy agency [12]

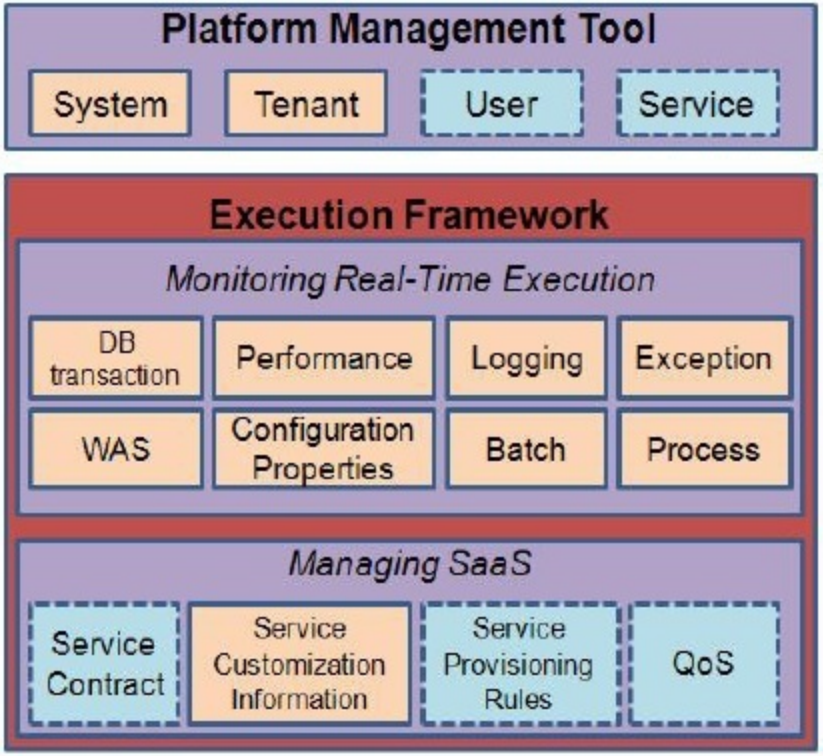
1-Metadata Driven Architecture[15]

- SaaS architecture is a set of technologies and service used to develop , deploy, integrate and deliver SaaS application .



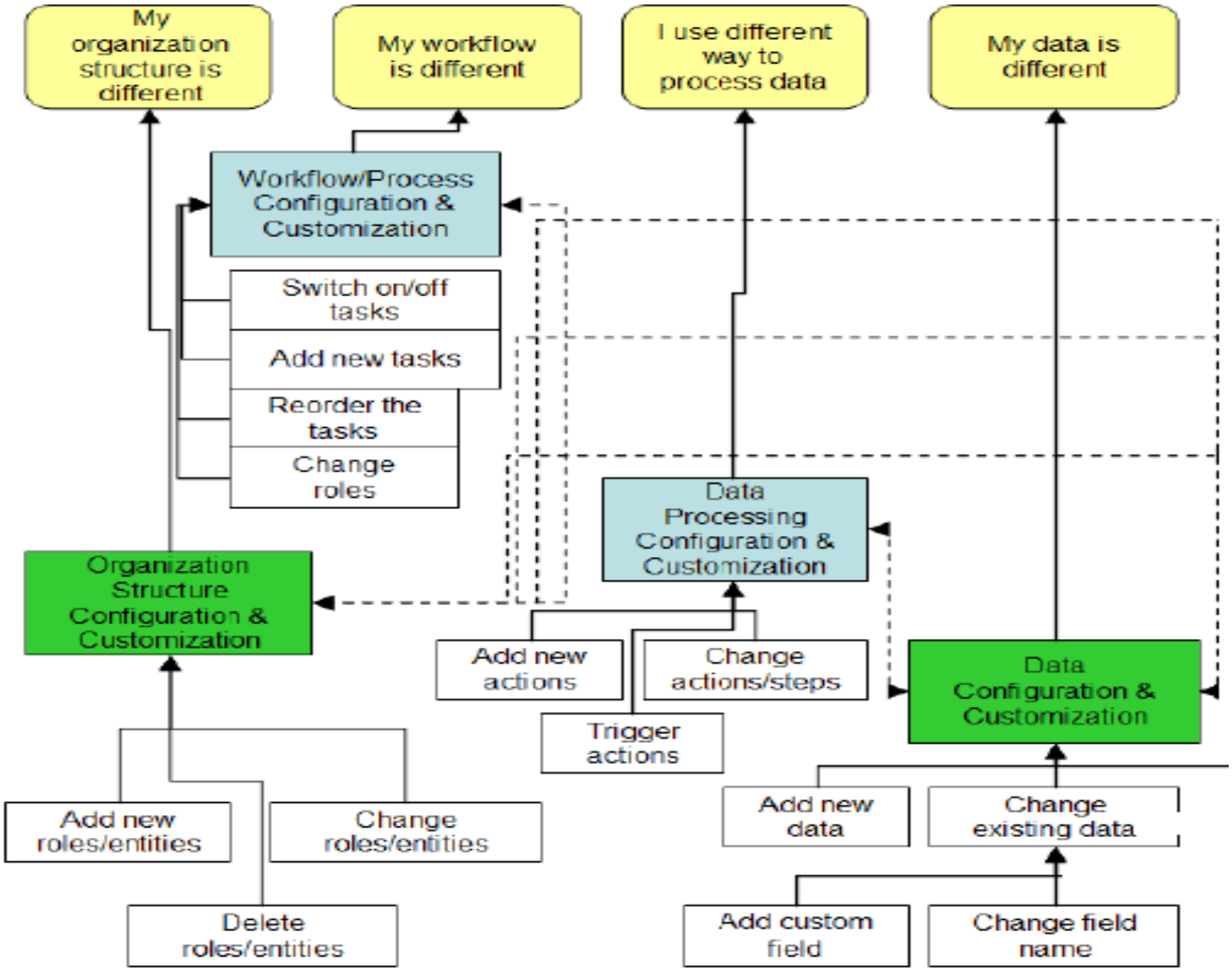
- Support GUI customization
- All are independently implemented .
- Customization environment help tenant specific execution
- Can be hosted on any cloud provider that support core development technology

Execution framework

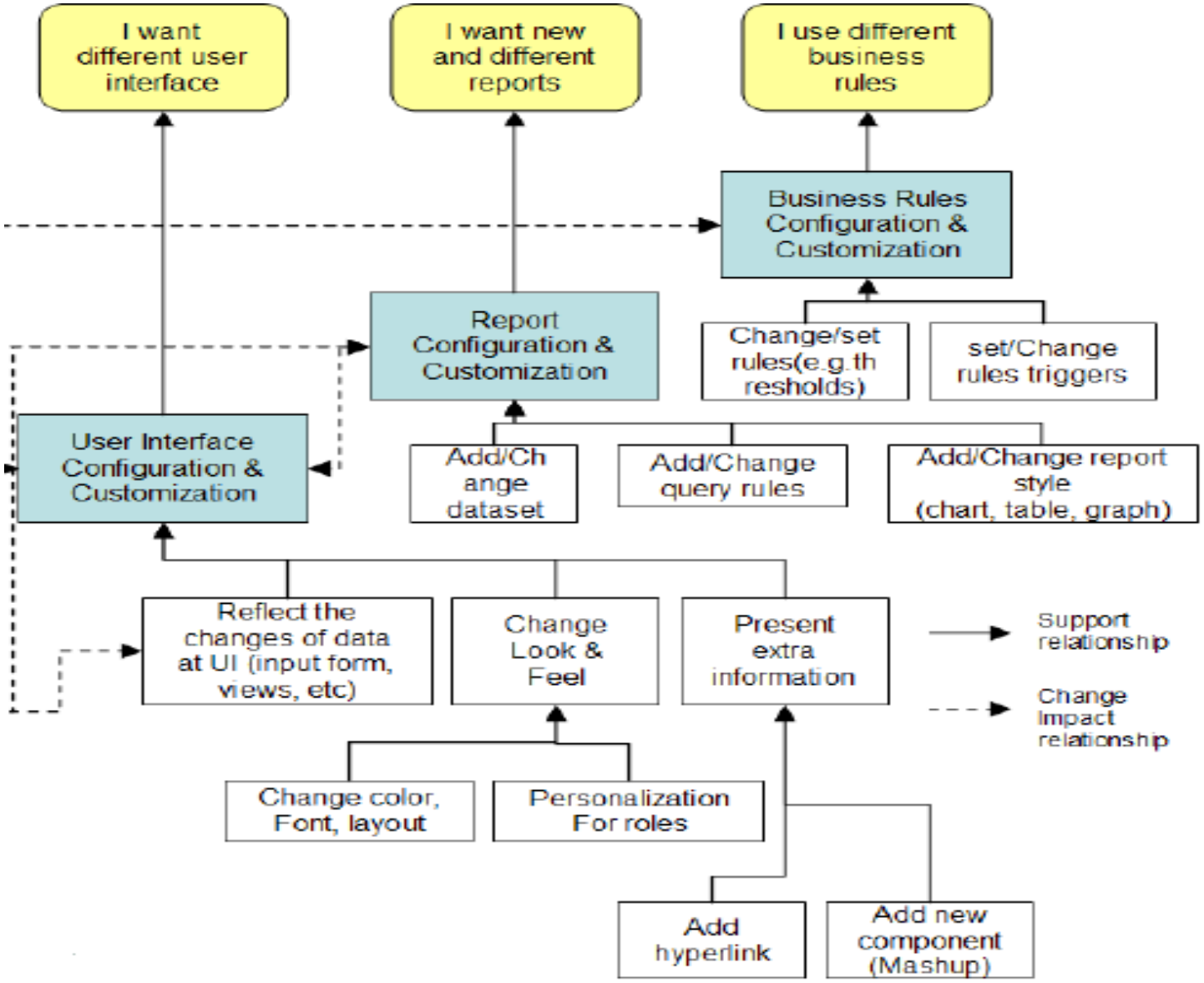


- All components are not implemented yet.
- Monitoring is required to monitor different parameters .
- Managing SaaS: control the SLA

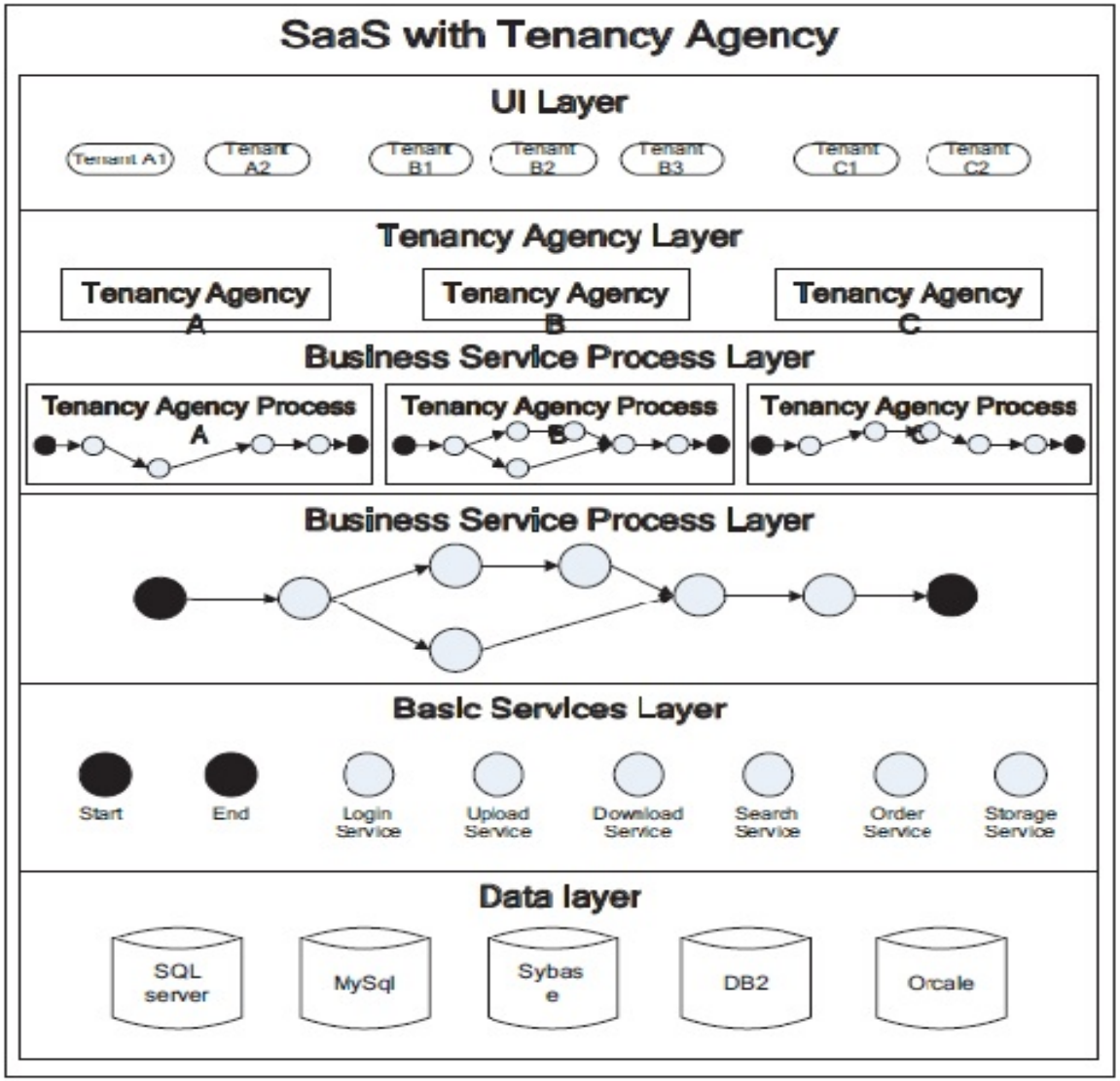
Possible Configuration Block



Cont..


I want different user interface
I want new and different reports
I use different business rules
Business Rules Configuration & Customization
Report Configuration & Customization
Change/set rules(e.g.th resholds)
set/Change rules triggers
User Interface Configuration & Customization
Add/Change dataset
Add/Change query rules
Add/Change report style (chart, table, graph)
Reflect the changes of data at UI (input form, views, etc)
Change Look & Feel
Present extra information
Support relationship
Change Impact relationship
Change color, Font, layout
Personalization For roles
Add hyperlink
Add new component (Mashup)

2-SaaS with tenancy agency [12]



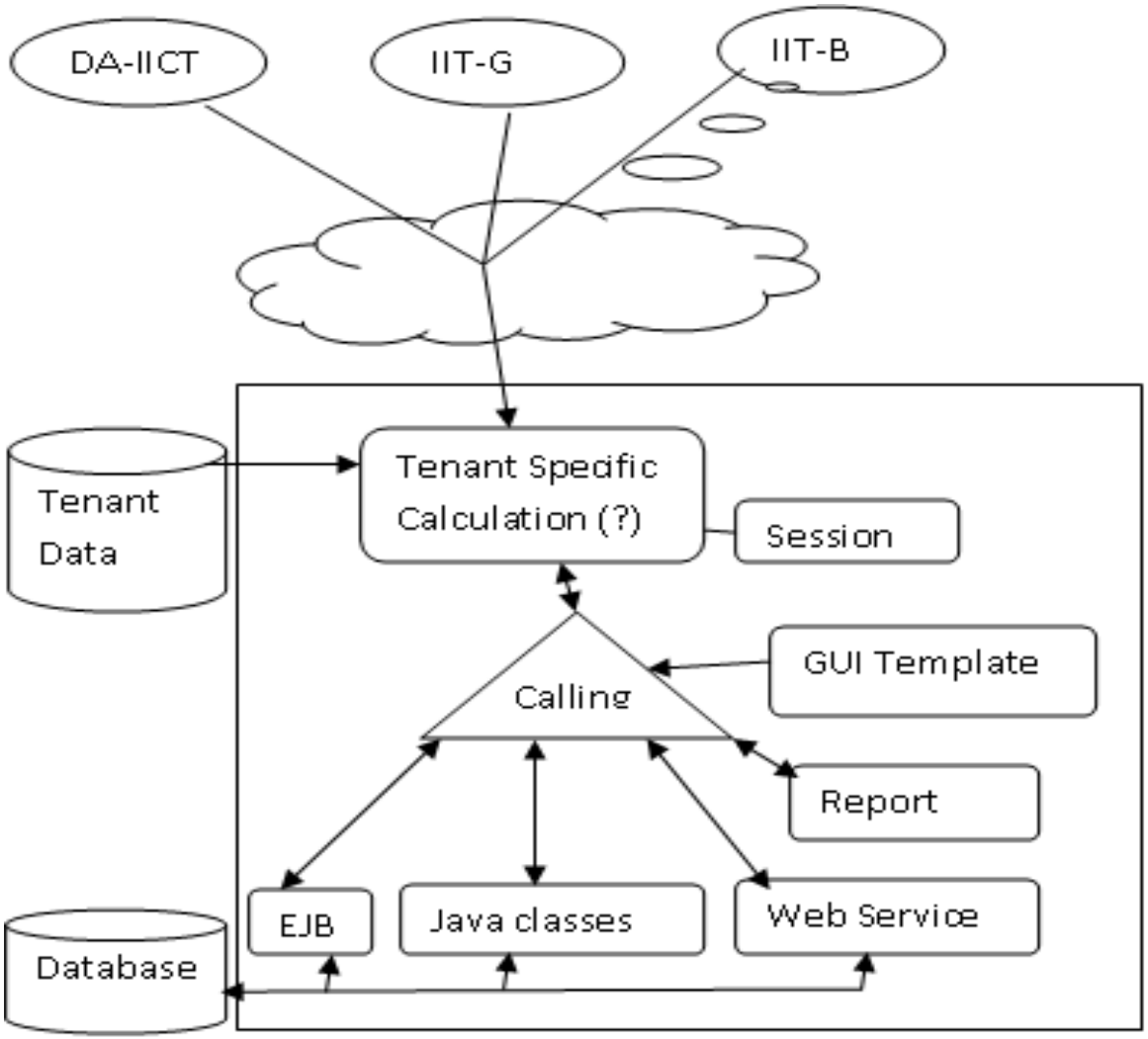
Tenancy agency

- It has mainly two layer

  1) Tenancy process layer

  2) Business process layer
- More SOA kind of architecture rather than dedicated SaaS
- In such a architecture agencies avoid being too huge and work with no interference.
- All processes are implemented as web services, use of WSDL or WSCL to explain about services.
- Less configuration allowed but more customization support is required.

University Architecture Components



Some functionality may change for tenants

- **Technical challenges**
- **Input form :**
  - Add/Delete new field in form for a tenant
  - Validations: requirement of a particular format
  - Whether it is mandatory or not

- **Work flow of application**
  - Student direct can register and management will approve it
  - Whether student can view profiles of other students or not.
  - Administrator hierarchy may be different

- **Business logic:**
  - Grading and evaluation scheme differs
  - Course structure differs

- **Report viewing / Rights management .**
  - Reports: only for selected employees
  - Student can not view grade sheets before final processing is completed.

References

[1] Peter Mell and Tim Grance, ″**The NIST Definition of Cloud Computing**″ Version 15, 10-7-09

[2] Robert L. Grossman **,” A Quick Introduction to Clouds”**, University of Illinois at Chicago and Open Data Group October 29, 2008

[3] “**Getting Started with Cloud Computing “,** Amazon EC2 on Red Hat Enterprise Linux

[4] Softserve Inc, Web: www.softserveinc.com ,White Paper, **“ SoftServe SaaS Development Kit For Open Source Java**”

[5] Krzysztof Sledziewski1, Behzad Bordbar1 and Rachid Anane2 **, “A DSL-based Approach to Software Development and Deployment on Cloud”,** IEEE International Conference on Advanced Information Networking and Applications, 2010

[6] Review Business and Technology Series | www.cumulux.com **, “Which cloud platform is right for you? comparing cloud platforms”**

References

[7] Ignacio M. Llorente , "**OpenNebula Cloud Case Studies for Research and Industry**", DSA-Research.org Distributed Systems Architecture Research Group Universidad Complutense de Madrid

[8] Michael Armbrust,Armando Fox,Rean Griffith ," **Above the Clouds: A Berkeley View of Cloud Computing ",** Electrical Engineering and Computer Sciences University of California at Berkeley , February 10, 2009

[9] Mladen A. Vouk , "**Cloud Computing – Issues,Research and Implementations**" , Journal of Computing and Information Technology - CIT 16, 2008, 4, 235–246 ,doi:10.2498/cit.1001391 Department of Computer Science, North Carolina State University, Raleigh, North Carolina, USA

[10] A SaaS-Attack White Paper ,"**Top 10 Design Issues on SaaS**"

[11] A SaaS-Attack White Paper **," SaaS – Application Migration Technical Assessment"**

References

[12] Wenyu Liu, Bin Zhang, Ying Liu, Deshuai Wang Yichuan Zhang "**New Model of SaaS: SaaS with Tenancy Agency**" College of Information Science and Engineering, Software College Northeastern University, NEU Shenyang, China

[13] Wei Sun,Zing Zhang,Chang jie Guo,Pei Sun, Hui Su , "**Software as a Service: Configuration and Customization Perspectives**" , IBM China Research Lab,Beijing 100094, P.R China.

[14] Jiehui JU1,Ya Wang3,Jianqing FU2,Jiyi WU3,4,Zhijie LIN1,3, "**Research on Key Technology in SaaS**", School of Information and Electronic Engineering, Zhejiang University of Science and Technology Hangzhou Normal University Hangzhou, P.R.China

[15] Jihyum Lee , Sung jin Hur, " **Level-2 SaaS Platform and Platform Management Framework"**, Cloud Computing Research Department- ETRI , South Korea.

Thank you for your attention