BITS Pilani presentation

BITS Pilani
Hyderabad Campus

D. Powar
Lecturer,
BITS-Pilani, Hyderabad Campus

innovate achieve lead

BITS Pilani
Hyderabad Campus

SSZG527

Cloud Computing

Agenda:

- Multi-Tenancy
- 4 levels of multi tenancy
- Authentication
- Resource sharing
- Multi-tenant models for cloud services

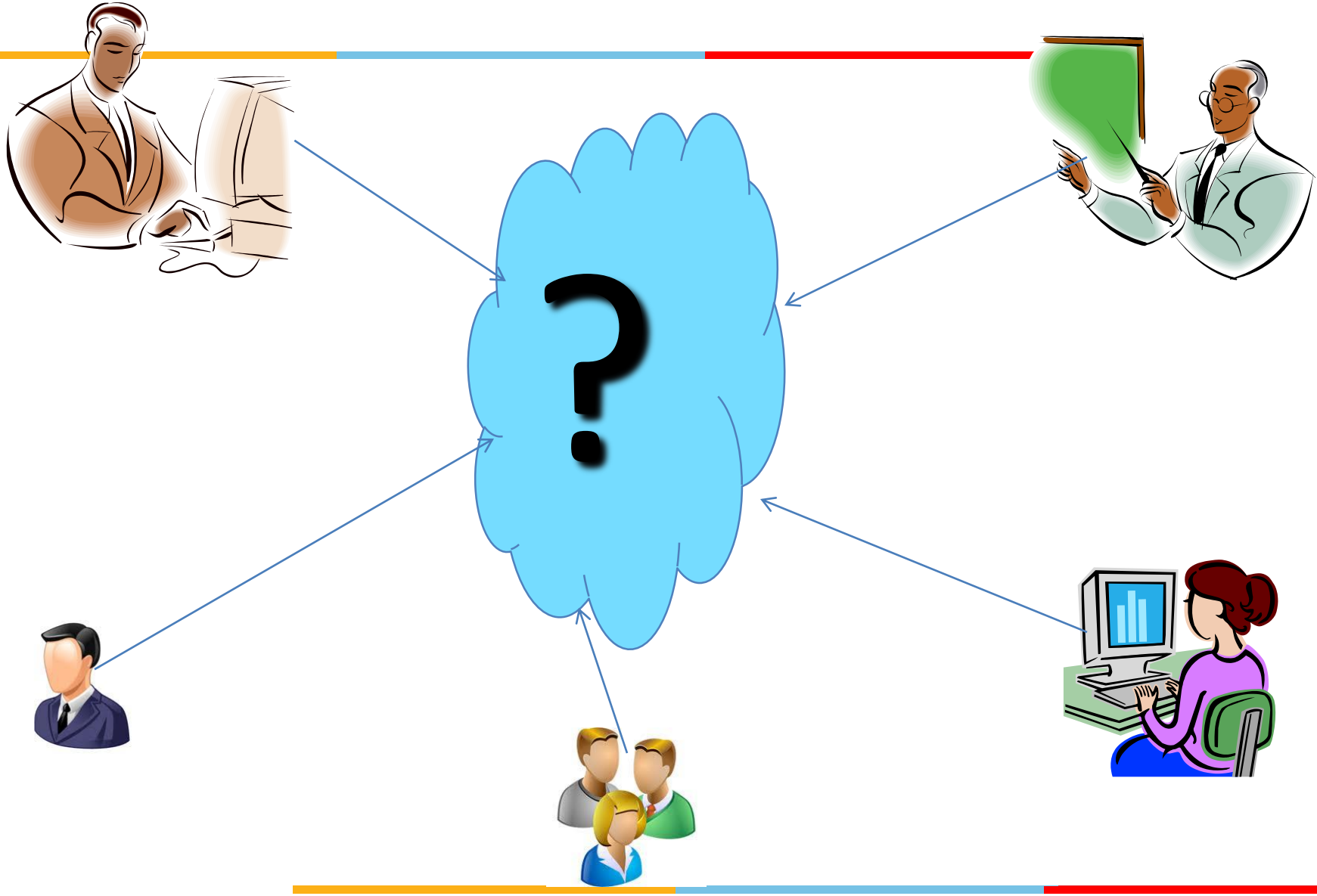
Multitenancy

- Multitenancy refers to a principle in software architecture where **a single instance** of the software runs on a server, **serving multiple client organizations** (tenants).
- Multitenancy is contrasted with a multi-instance architecture where separate software instances (or hardware systems) are set up for different client organizations
  - wiki

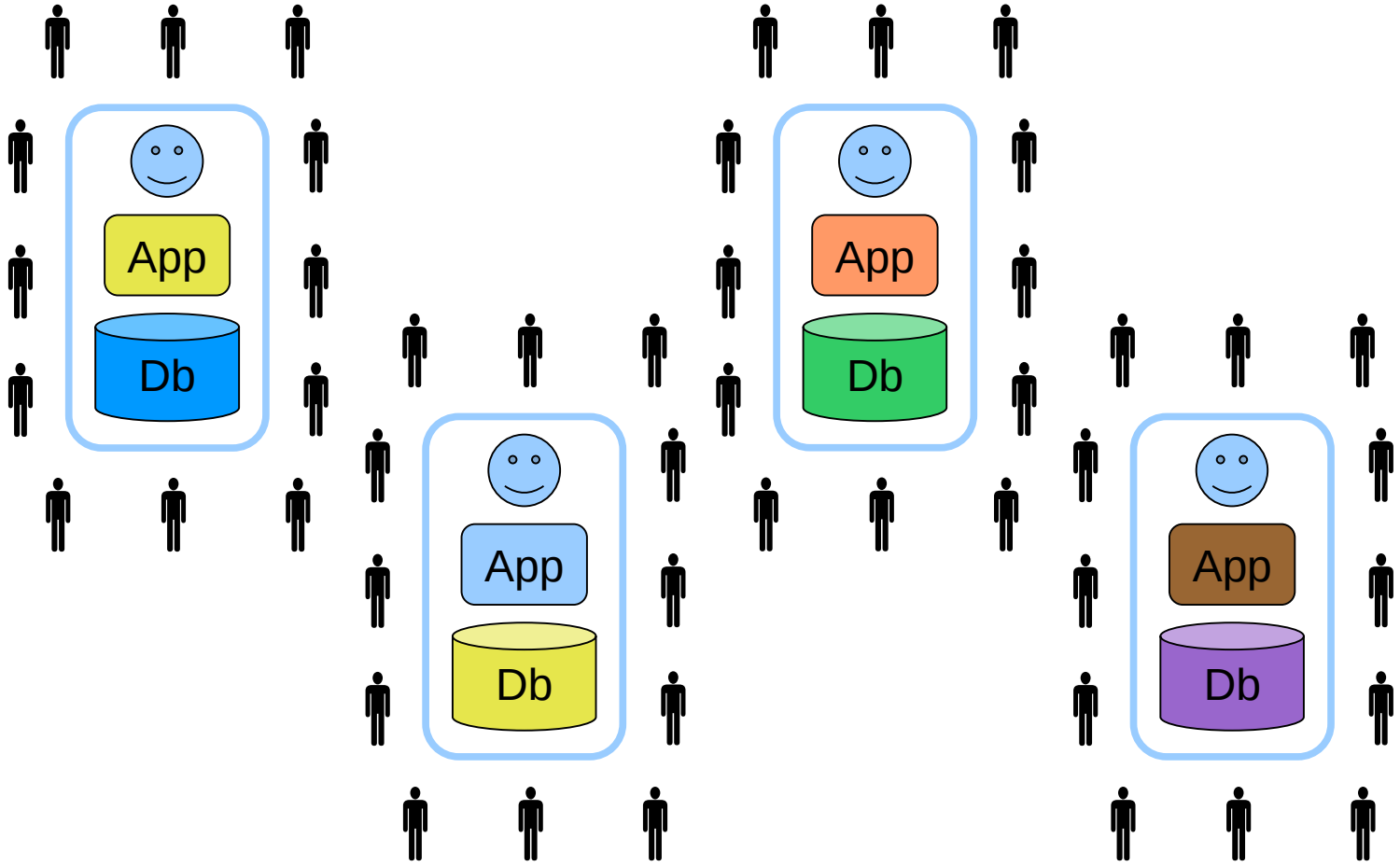
Multitenancy

- Multi-tenancy is a critical technology to allow one instance of application to serve multiple customers by sharing resources.
  - ✓ Multi - multiple, independent customers are served
  - ✓ tenant is any legal entity responsible for data and is provided on a contractual basis. Tenant is the contract signee
- Applications : IaaS, PaaS, SaaS

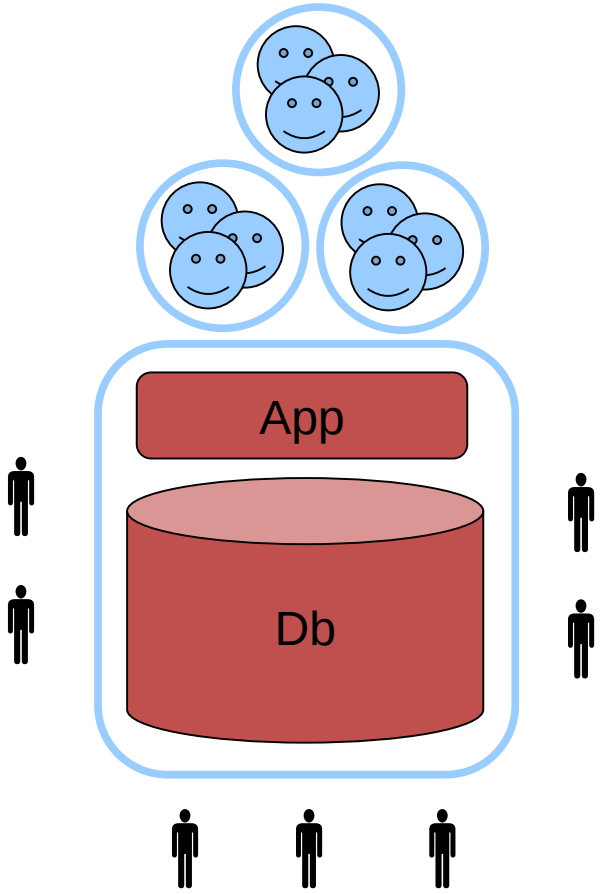
Requirement of Multitenancy



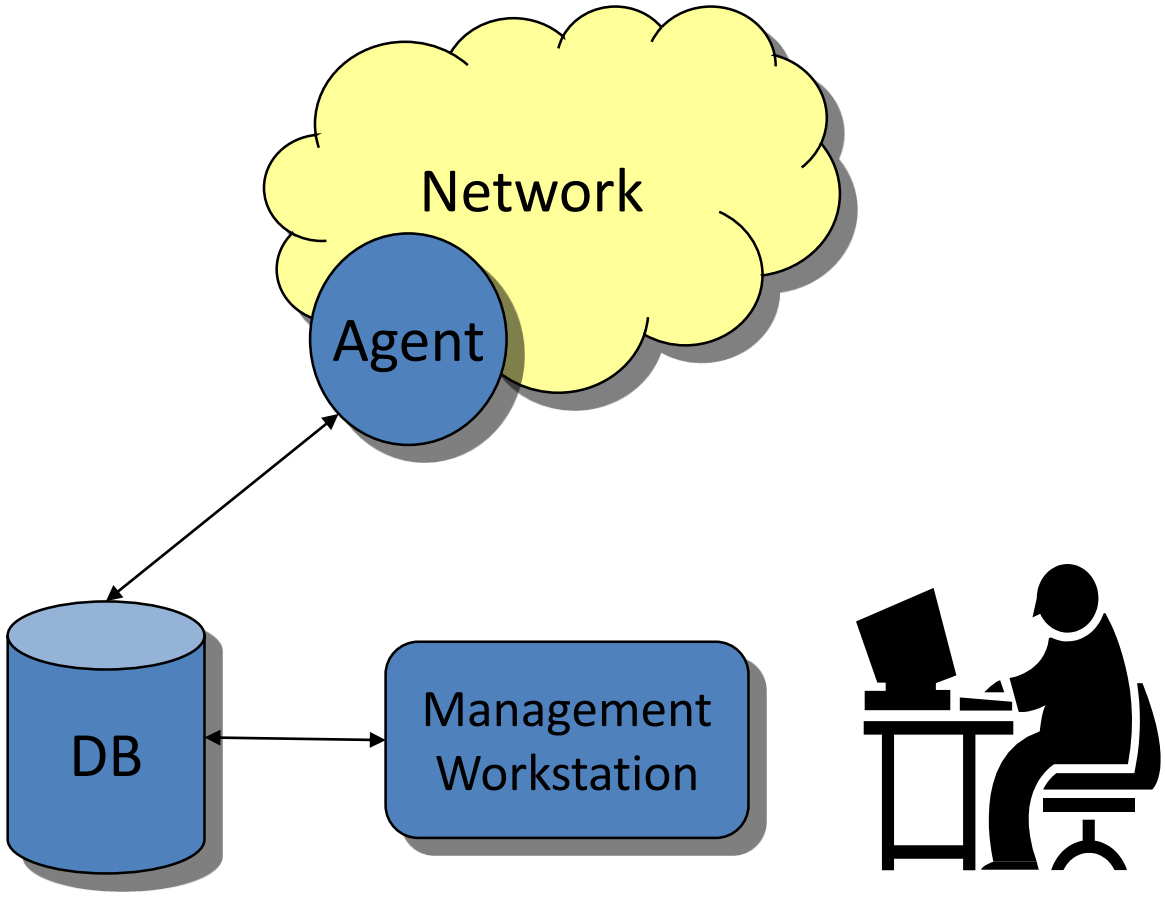
Single tenant applications: lots of waste



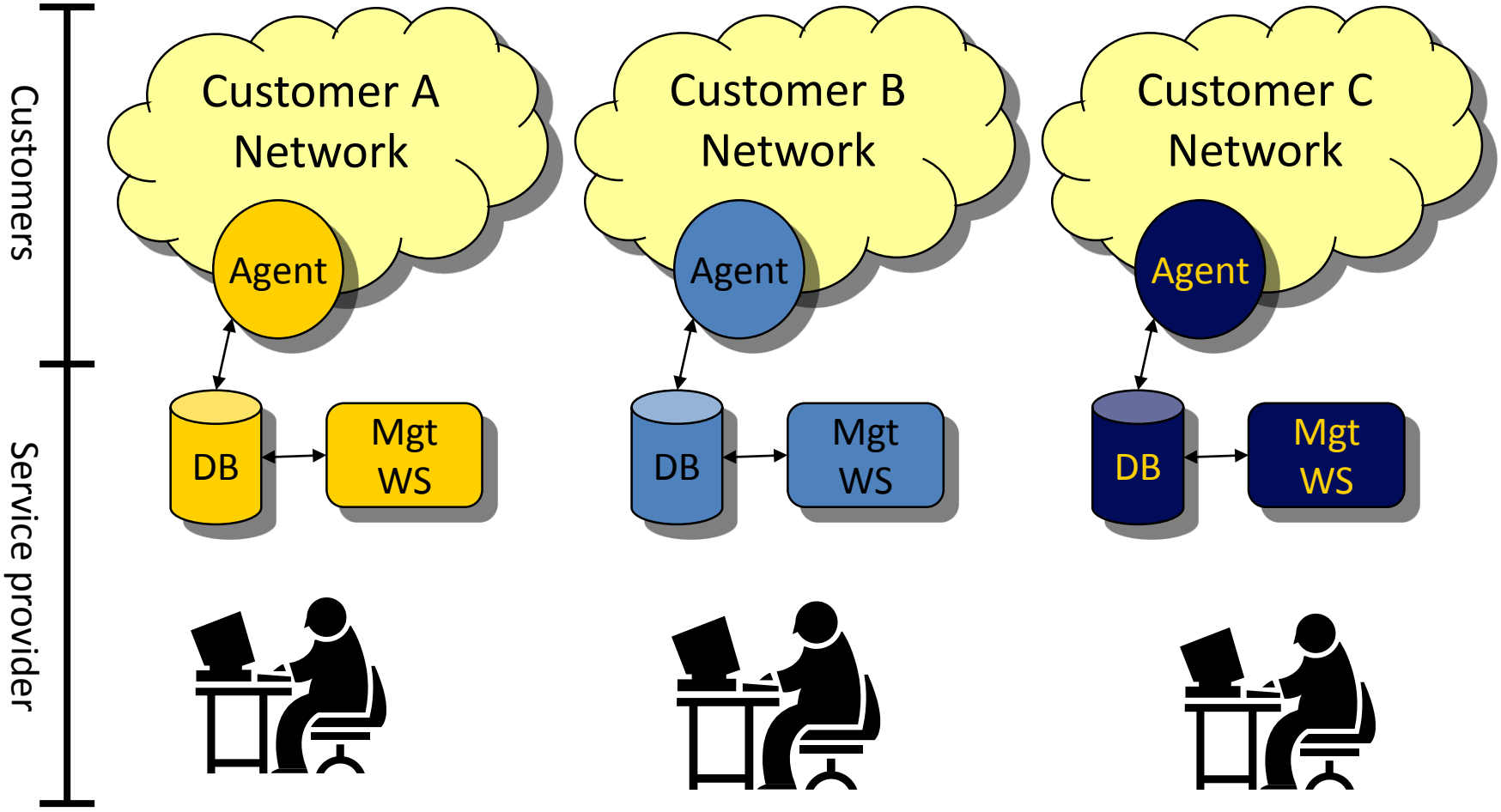
Multi-tenancy benefits are self-evident
But isolation is much easier said than done…



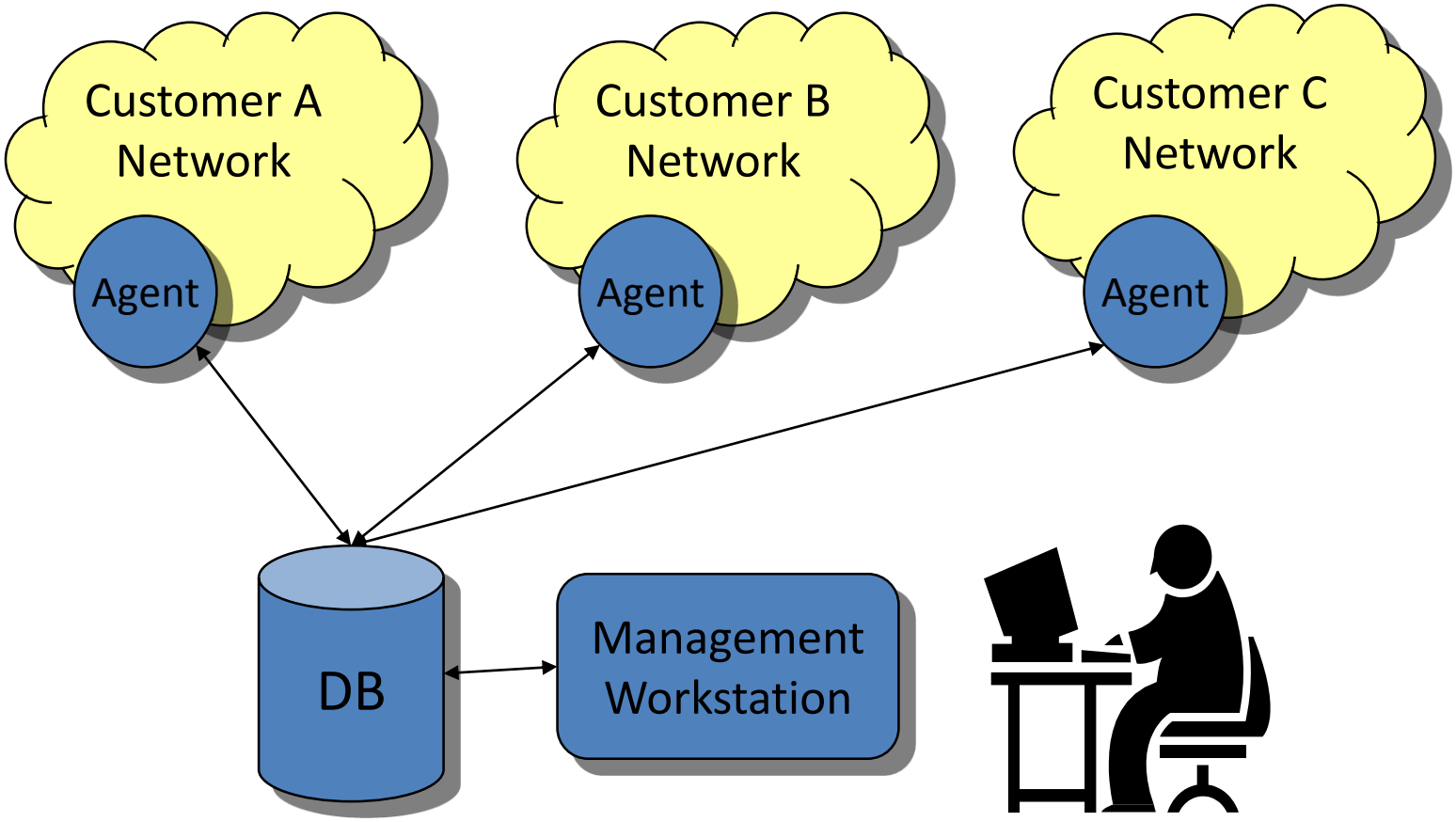
Typical Network Monitoring Infrastructure



Monitor Multiple Customers Using Typical Infrastructure



Multi-Tenant Network Monitoring Infrastructure



Goals of Multi-tenancy

- Sharing – maximize the resource sharing across multiple tenants.
- Isolation – hide the facts: others are also in the same server.
  - **Execution** – enforce security. Make sure one tenant can't call other tenants executable logic.
  - **Data** – make sure one tenant can't see other data
  - **Performance** - make sure performance is not affected by existence of other tenants.
- Scale
  - Server is distributed and it can handle larger load by adding more nodes.

Experiment by Jacob et al.,

- The experiment compares 3 methods of creating a database shared among multiple customers
  - ✓ Shared machine
  - ✓ Shared process
  - ✓ Shared table

Shared machine

- Each customer was given their own database process and tables on a shared machine

Shared process

- Each customer had their own database tables, but there was only one database process which executed instructions on behalf of all customers

Shared table

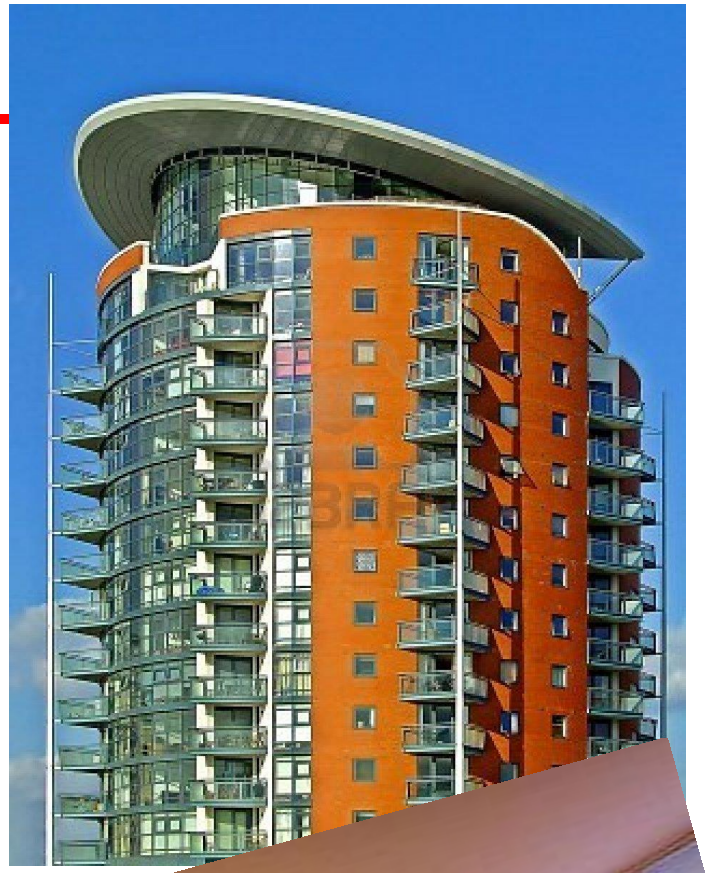
- In addition to the customers sharing the database process, the data was stored on shared tables
  - ✓ Each row being prefixed with the customer id to indicate the customer to which the row belonged

Shared table example

Account

TenId	AcctId	Name	...
17	1	Acme	
17	2	Gump	
35	1	Ball	
42	1	Big	

single instance, mult-tenant



multi-tenant efficient
customisable
scaleable

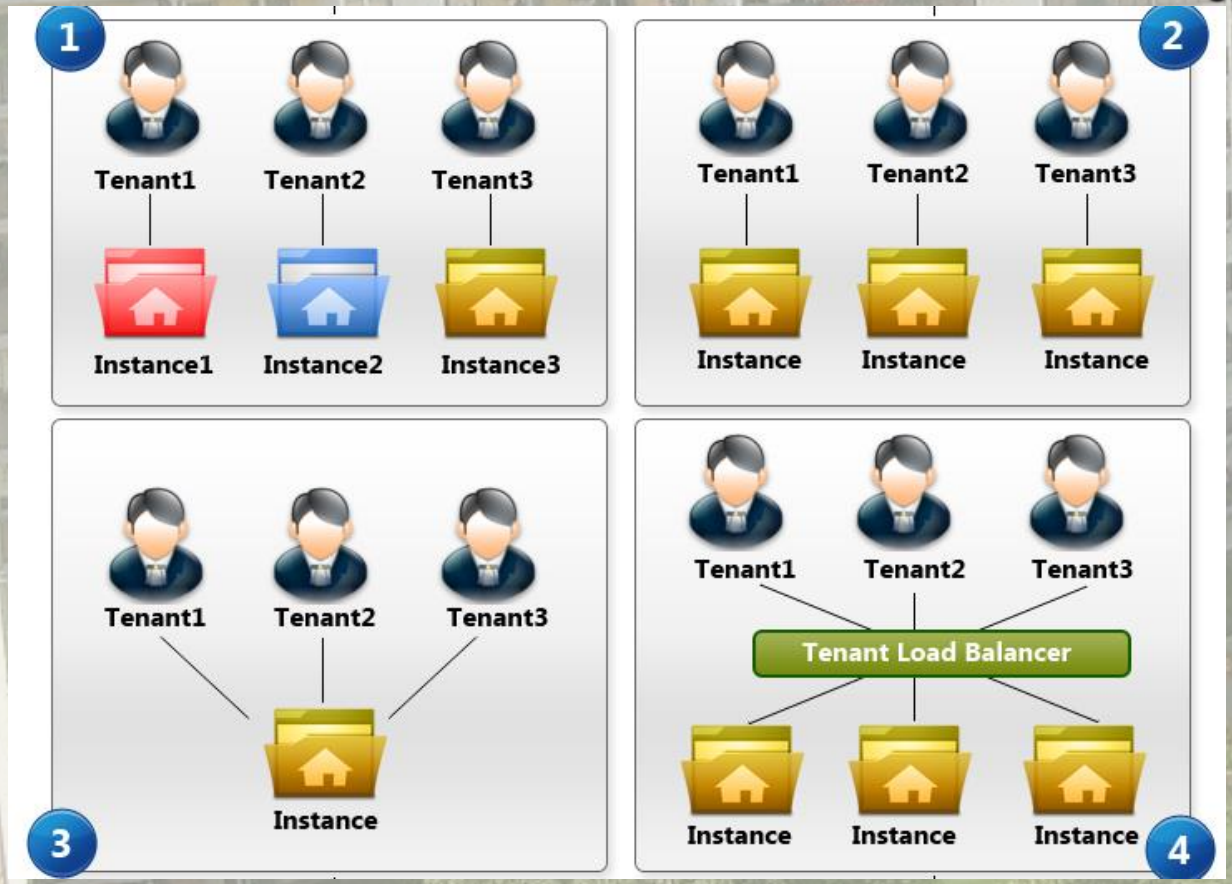
4 levels of multi tenancy

1. Ad-hoc /customizable instances
2. Configurable instances
3. Configurable multi – tenant efficient instances
4. Scalable, configurable, multi-tenant efficient instances

For any given resource in a cloud system, the appropriate level could be selected

Basic SaaS maturity model

1. ad-hoc /custom



2. configurable single tenant

3. configurable multi tenant

4. configurable multi tenant (scalable)

Ad-hoc /customizable instances

- Each customer has their own custom vision of the software
- Represents a enterprise data center where there are multiple instances and versions of the software
- Each customer would have their own binaries, as well as their own dedicated processes for implementation of the application
- Disadv: Difficulty in Management: Each customer would need their own management support

Configurable instances

- All customers share the same vision of the software (one copy for each customer)
- adv: Easy Management: Single copy of the software

Configurable multi-tenant efficient instances

- All customers share the same version of the software (only single copy among all customers)
- adv: Easy Management: running of only single instance

Configurable multi-tenant efficient instances (scalable)

- All customers share the same version of the software (only single copy among all customers)
- Software is hosted on a cluster of computers
- Hence, allows the capacity of the system to scale almost limitlessly
- Thus, increase in no. of customers and capacity as well
- Ex: Gmail, yahoo mail, etc
- Disadv: Shared storage problem

vs

share

isolate

business model (can I monetise?)

architectural model (can I do it?)

operational model (can I guarantee SLAs?)

meta-data

access control

Authentication

- Unlike traditional computer systems, the tenant would specify the valid users, and cloud service provider would authenticate them
- Two basic approaches are used
  - Centralized authentication
  - Decentralized authentication

Authentication (contd..)

Centralized authentication:

- Authentication is performed using a centralized user database
- Cloud admin gives the tenant's admin rights to manage user accounts for that tenant
- Multiple (two) sign-on service (Sign on the CAS, in addition to the tenant's AS)
- Given self service nature of the cloud, it is more generally used

Decentralized authentication:

- Each tenant maintains their own user database, and needs to deploy a federation service that interface between that tenant's authentication framework and the cloud system's authentication service
- Single sign-on service

Resource sharing

- Two major resources that need to be shared are storage and servers
- **Sharing storage resources (two types)**
  - ✓ **File system**
  - ✓ **Databases**

- Since file system storage is well known mechanism, we will restrict our discussion to database storage

Database

There are two methods of sharing data in a single database

- ✓ *Dedicated tables per tenant*
- ✓ *Shared table*

Dedicated tables per tenant:

- ❖ Each tenant stores their data in a separate set of tables different from other tenants
- ❖ ex: www.mygarage.com portal
- ❖ Shows the way auto repair stores may store each table as separate file

Dedicated tables per tenant:

Best garage

Car license	Service	Cost

Friendly garage

Car license	Service	Cost

Honest garage

Car license	Service	Cost

Shared table:

- The data for all the tenant is stored in the same table in different rows.
- One of the column in the table identifies a tenant to which a particular row belongs
- It is more space efficient than previous approach
- A auxiliary table, called a metadata table, stores information about the tenants

Shared table (contd..)

Data table 1

Tenant ID	Car license	Repair	Cost
1			
2			
2			
1			
3			
2			

Metadata table 1

Tenant ID	Data
1	Best garage
2	Friendly garage
3	Honest garage

Data customization

- It is important for the cloud infrastructure to support customization of the stored data, since it is likely that different tenants may want to store different data in their tables
- In *Dedicated table* method, each tenant has their own table, and therefore can have different schema
- Difficulty is with shared table approach
- Three method used
  - ✓ Pre-allocated columns
  - ✓ Name-value pair
  - ✓ XML method

Pre-allocated columns

- Space is reserved in the tables for custom columns, which can be used by tenants for defining new columns
- Salesforce.com reserves 500 columns
- Some of the tenants may not use these columns

Disadv: There could be a lot of wasted space

Pre-allocated columns

Tenant ID	Car license	Service	Cost	Custom1	Custom2
1					
2					
2					
1					
3					
2					

Data table 1

Tenant ID	Tenant name	Custom1 name	Custom1 type
1	Best garage	Service rating	int
2	Friendly garage	Service manager	string
3	Honest garage		

Metadata table 1

Name-value pair

- The standard table will have an extra column which is a pointer to a table of name-value pair, which indicates additional custom fields for a record
- The table name-value pair is also called as a pivot table
- This method overcomes the deficiencies of storage wastage from previous method

Name-value pair (contd..)

Tenant ID	Car license	Service	Cost	Name-value pair record
1				275
2				
2				
1				
3				
2				

Data table 1

Name-value pair	Name ID	Value
275	15	5.5

Data table 2

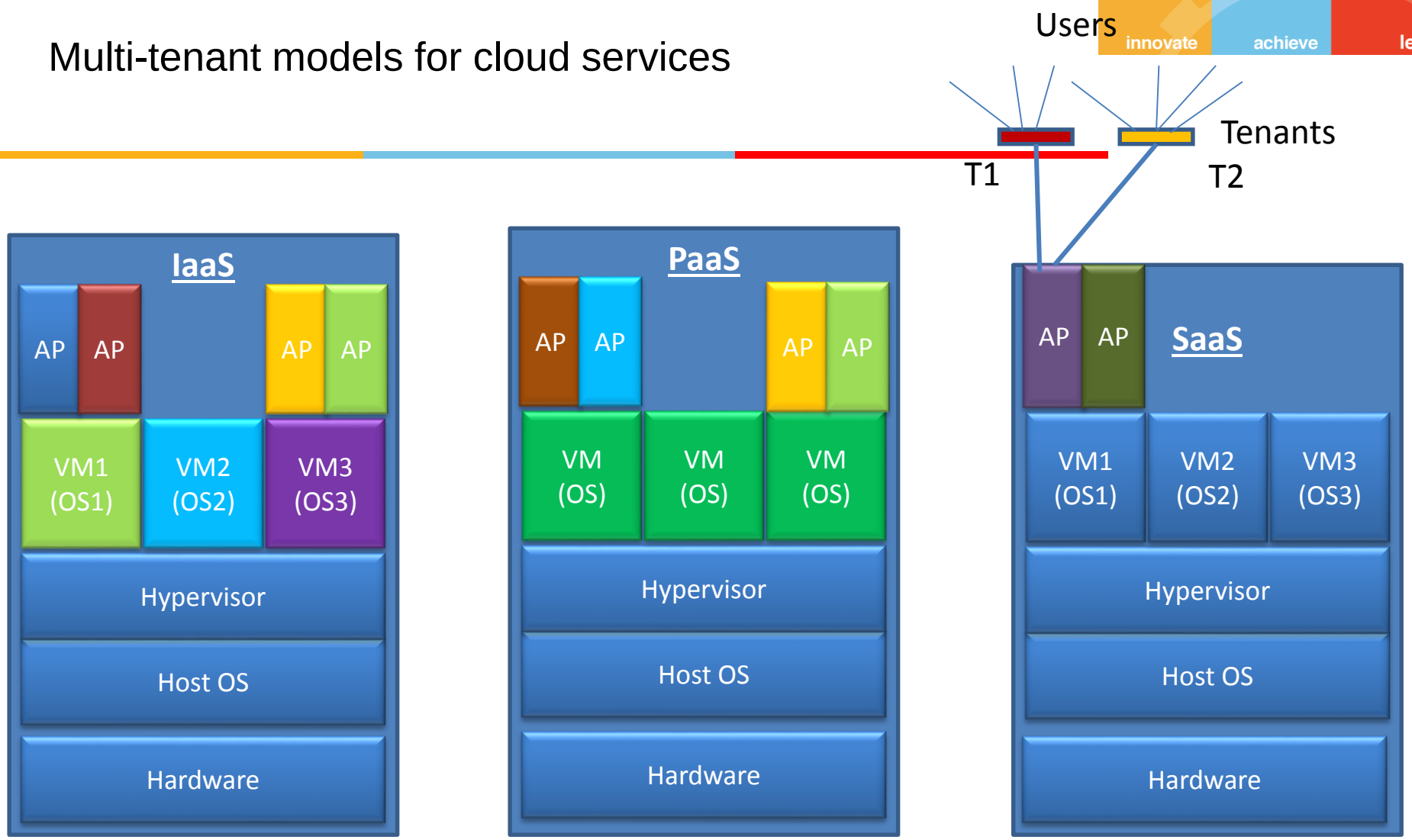
Metadata table 1

Name ID	Name	Type
15	Service rating	int
	Service manager	string

Metadata table 2

Tenant ID	Data
1	Best garage
2	Friendly garage
3	Honest garage

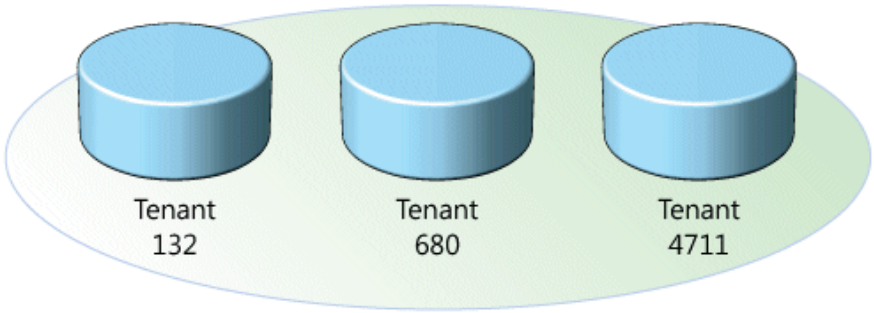
Multi-tenant models for cloud services



Private cloud/ IT center

Development center

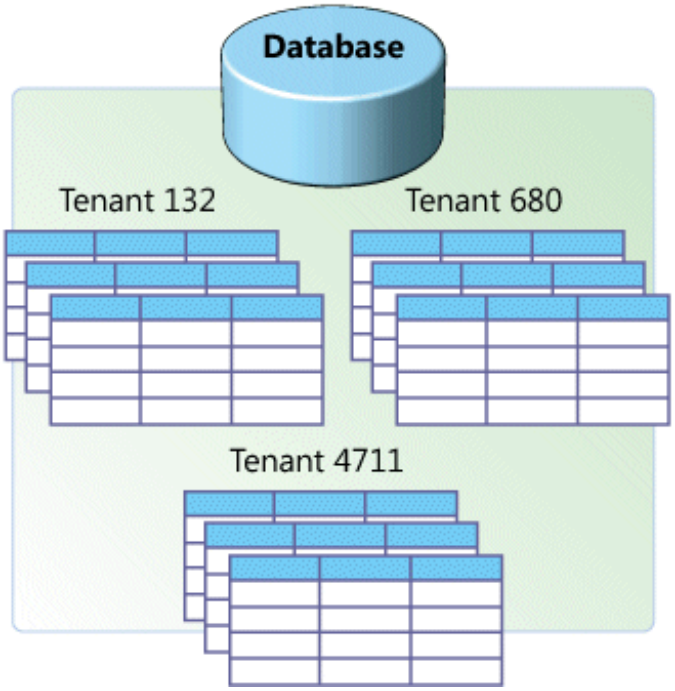
Data center



TenantID	CustName	Address

4	TenantID	ProductID	ProductNam

1	4	TenantID	Shipment	Date
6	1	4711	324965	2006-02-21
4	6	132	115468	2006-04-08
	4	680	654109	2006-03-27
		4711	324956	2006-02-23



Summary

- Multi-Tenancy
- 4 levels of multi tenancy
- Authentication
- Resource sharing
- Multi-tenant models for cloud services