# SYNERGY2015

## SEATTLE, WA, USA

# SYNERGY 2015

SEATTLE, WA, USA

# Multi-Tenant SQL

Presented by Eddy Kleinjan

How to handle multi-tenancy in applications.
*Are you ready to take your application to the next level?*

**Data**Access WORLDWIDE

# SaaS: Software as a Service

- ◦ "Software deployed as a hosted service and accessed over the Internet."
- ◦ Source: Architecture Strategies for Catching the Long Tail Frederick Chong and Gianpaolo Carraro, Microsoft



Customer 1

Internet
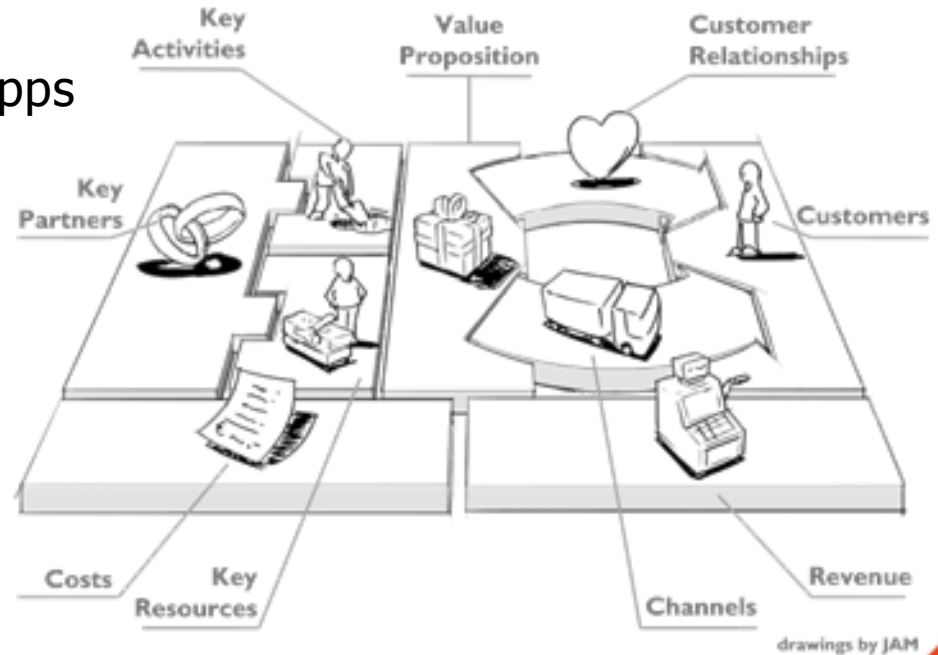
Server(s)

Customer 2
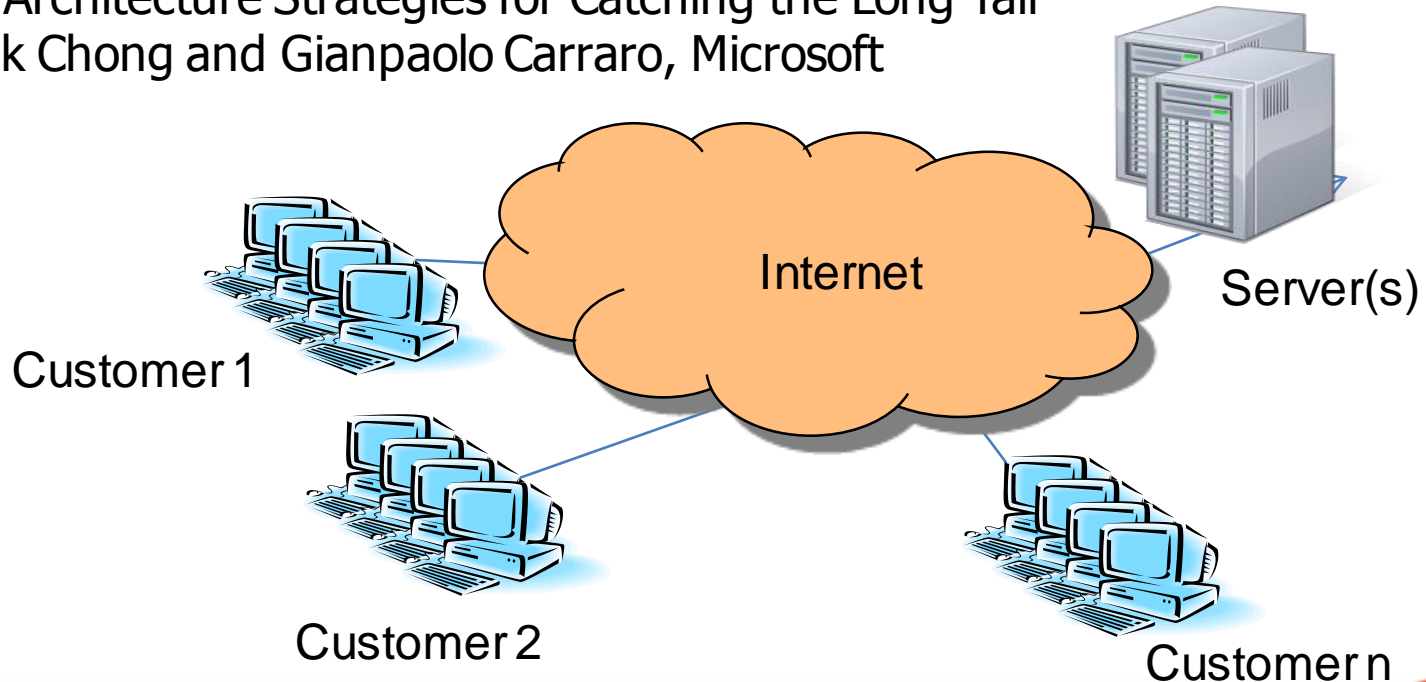
Customer n

# Client view: Reduce Costs

- Changing the Business Model
- Shifting "ownership" of the software from the customer to an external provider
- Reallocating responsibility from customer to provider
    - Software
    - Hardware
    - Professional Services
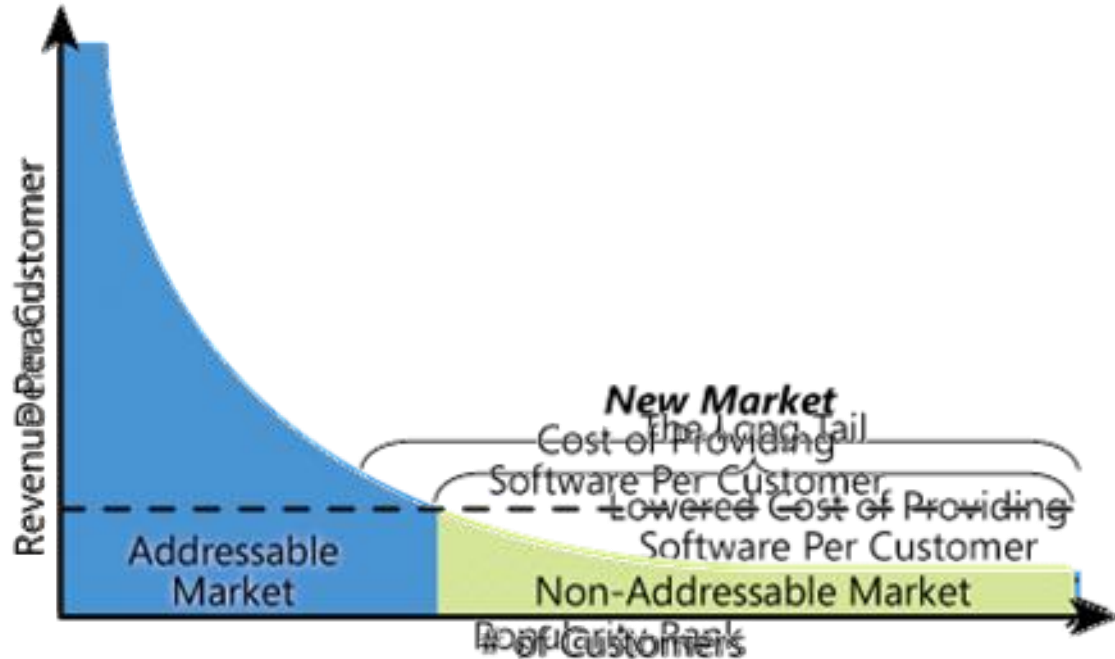- Reducing cost

**DataAccess** WORLDWIDE
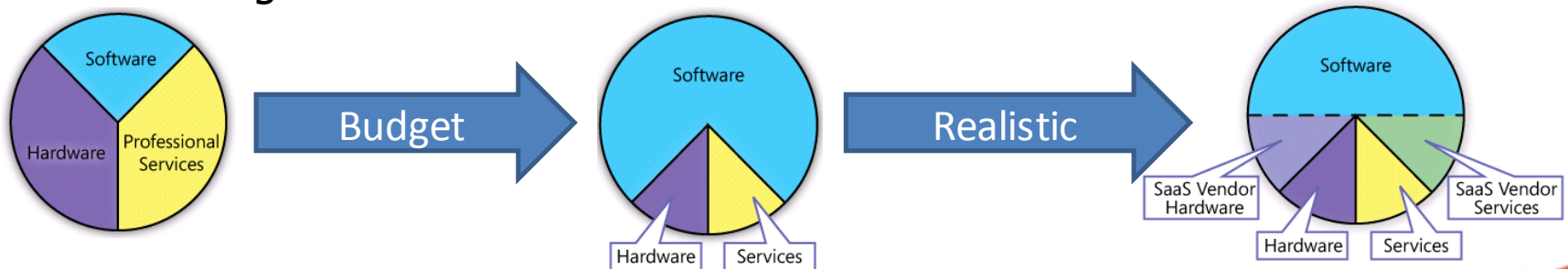
# Provider view: Business Models

- Outsource IT Hardware and Software to Service Providers reducing cost
- Single Version of the Software
- Applies to both WebApps and WinApps
- Charge per
  - Number of users
  - Tenant
  - Item (customer, invoice...)
  - Time (day, month, etc.)
- Reduce up-front costs



Key Activities

Value Proposition

Customer Relationships

Key Partners

Customers

Costs

Key Resources

Channels

Revenue

drawings by JAM

DataAccess WORLDWIDE

# The Long Tail

- ◦ Selling to the Long Tail, Chris Anderson *Wired* :
  http://www.wired.com/wired/archive/12.10/tail.html

# Trust

◦ Who is the service provider?

◦ Where is the data stored?

◦ What happens when one goes broke?

◦ Who takes care of back-up?

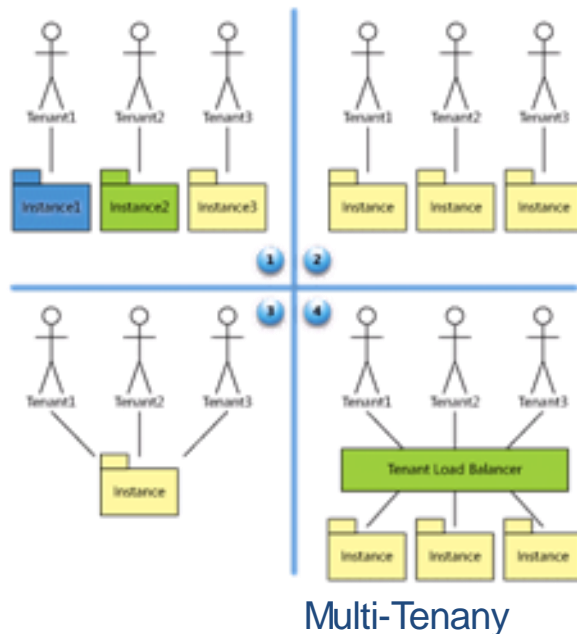◦ Can I move my data out when leaving?

◦ Etc...



The Trust Equation

$$\frac{\text{Weighted expectations} \times \text{Delivery on expectations}}{\text{Clarity of expectations} \times \text{Usefulness of feedback}} = \text{Trust}$$

2006 Jack Ricchiuto / www.DesigningLife.com

# Attributes of a Multi-Tenant Architecture



Scalability

Multi-Tenany

Configurability

# Scalability

- ◦ Process Pooling (Scale Up)
- ◦ Load Balancing (Scale Out)
- ◦ SPLF: Security, Performance, Load Balancing and Failover
- ◦ Use Client / Server Database Model
- ◦ IP Address Sharing
- ◦ Multiple Servers
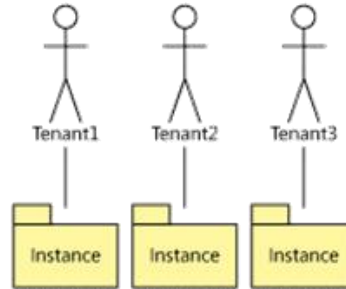- ◦ Database Partitioning
- ◦ Bandwidth

**DataAccess** WORLDWIDE
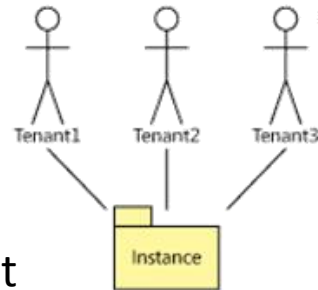
# Multi-Tenancy



Ad Hoc / Custom
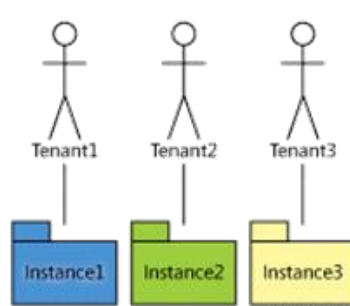
Configurable

Configurable,
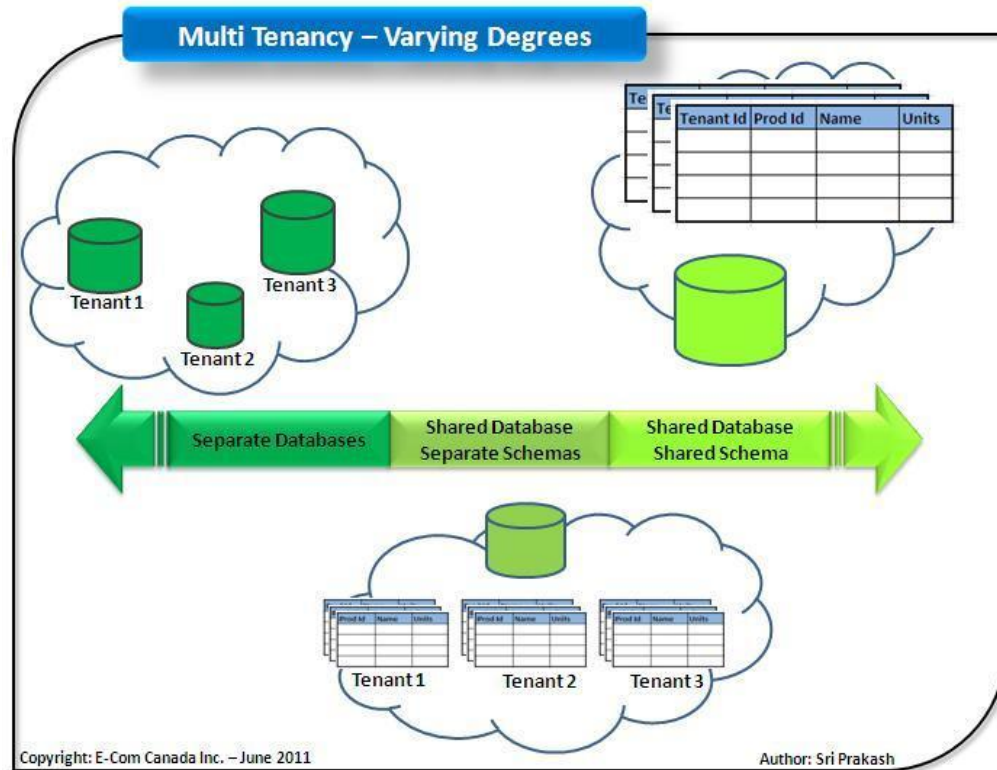Multi-Tenant-Efficient

Scalable,
Configurable,
Multi-Tenant-Efficient

DataAccess WORLDWIDE
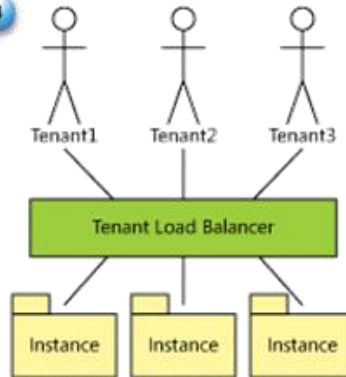
# Database Considerations

DataAccess WORLDWIDE

# How to design your database

# All data in one database

| TenantID Integer | FirstName String | BirthDate Date | RecordID Integer |
|---|---|---|---|
| 345 | Ted | 1970-07-02 | 893 |
| 777 | Kay | 1956-09-25 | null |
| 784 | Mary | 1962-12-21 | 564 |
| 345 | Ned | 1940-03-08 | 117 |
| 438 | Pat | 1952-11-04 | 301 |

| RecordID | Name | Value |
|---|---|---|
| 893 | Subscriber | Yes |
| null | Status | Gold |
| 564 | Expire | 2008-07-29 |
| 117 | Subscriber | No |
| 301 | Affiliation | Acme |

## Pros

- One database structure; easily maintained
- Create Cross-Tenant Queries Easily
- One backup procedure

| TenantID Integer | FirstName String | BirthDate Date | Custom1 Integer | Custom2 String | Custom3 Untyped |
|---|---|---|---|---|---|
| 345 | Ted | 1970-07-02 | null | Paid | null |
| 777 | Kay | 1956-09-25 | 23 | null | null |
| 784 | Mary | 1962-12-21 | null | null | null |
| 345 | Ned | 1940-03-08 | null | Paid | null |
| 438 | Pat | 1952-11-04 | null | San Francisco | Yes |

## Cons

- Size of Database can grow
- Harder to Partition Data
- Careful To Separate Data At All Times
- Harder to restore a database of a Tenant

# Each Tenant in Separate Database

## Pros

- Data Can Easily Be Taken Out
- Databases can be partitioned over multiple database servers
- Physical Separation of data may be a Requirement
- Database Accounts to Control Access
- Backup and Restore databases individually

## Cons

- Structure Changes Need To Be Rolled Out Over Each Database
- Backup and Restore databases individually



Tenant 132    Tenant 680    Tenant 4711

DataAccess WORLDWIDE

# Process / Application Considerations

○ One version of the software

○ One WebApp Application Process or Separate process for each Tenant
   • Use WebApp Administrator to setup separate Application per Tenant

○ Same URL for each Tenant or Separate



ONE version of the TRUTH.

DataAccess WORLDWIDE

# Separate the data in WebApp

○ Use Procedure OnAttachProcess as a hook to set Contraints

**Description**

When process pooling is used, this event is sent to the WebApp object. By default, it does nothing, but it can be used to perform any initialization for the newly attached process.

```
{ MethodType=Event }
Procedure OnAttachProcess
End_procedure
```

○ Use DF_FILE_SQL_FILTER to set a hard constraint

○ When using psSQLFilter, mind pbApplyGlobalSQLFilters!!!
  • It's false by default, overruling DF_FILE_SQL_FILTER

**DataAccess** WORLDWIDE

# Encryption

○ Use HTTPS / SLL for traffic encryption

○ Enforce it!

# Authentication

- User Id / Password
- Client Certificates
- Client Certificates for Smartcard
- IP Address
- Multi-Pass Authentication
  - Something you know
  - Something you have

# Authorization

- Manage Permissions
- Logging of access using Sessions
- Menu System
- View Authorization

**DataAccess** WORLDWIDE
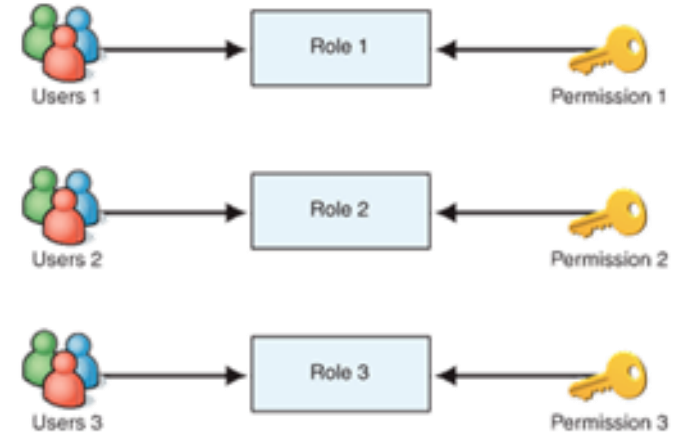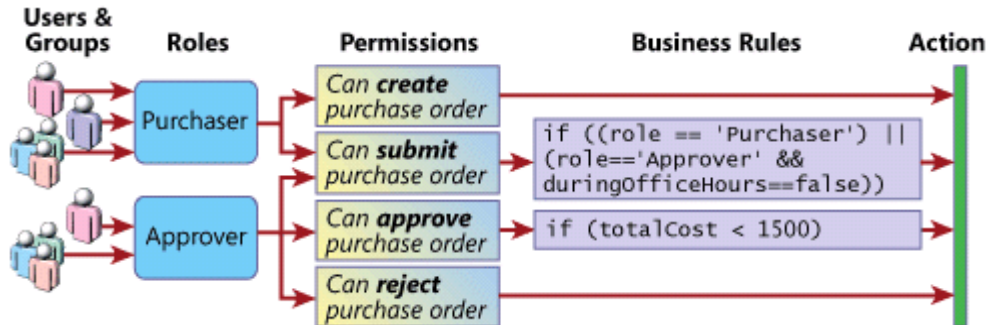
# Consider

- Legal Aspects
  - Where is the data located?
  - Who is responsible for data loss?

- SLA (Service Level Agreement)
  - Guaranteed up-time
  - Monitor Server Availability
  - Monitor Server Performance

DataAccess WORLDWIDE

# Where to go from here?

- [Architecture Strategies for Catching the Long Tail](http://msdn.microsoft.com/en-us/library/aa479069.aspx)
  http://msdn.microsoft.com/en-us/library/aa479069.aspx

- [Multi-Tenant Data Architecture](http://msdn.microsoft.com/en-us/library/aa479086.aspx#mlttntda_tde)
  http://msdn.microsoft.com/en-us/library/aa479086.aspx#mlttntda_tde

- Amazon RDS (Relational Data Services)
  - SQL Server, MySQL, Oracle, PostgreSQL

- Microsoft Azure SQL Database
  - Database as a Service

**DataAccess** WORLDWIDE

# Thank you.

Have a nice day!

SYNERGY 2015

SEATTLE, WA, USA