

ETH  SOFT

# **LIMS in the Cloud**

Bill Pingpank, *VP Client Services*  
*Ethosoft, Inc*

# Presentation Overview

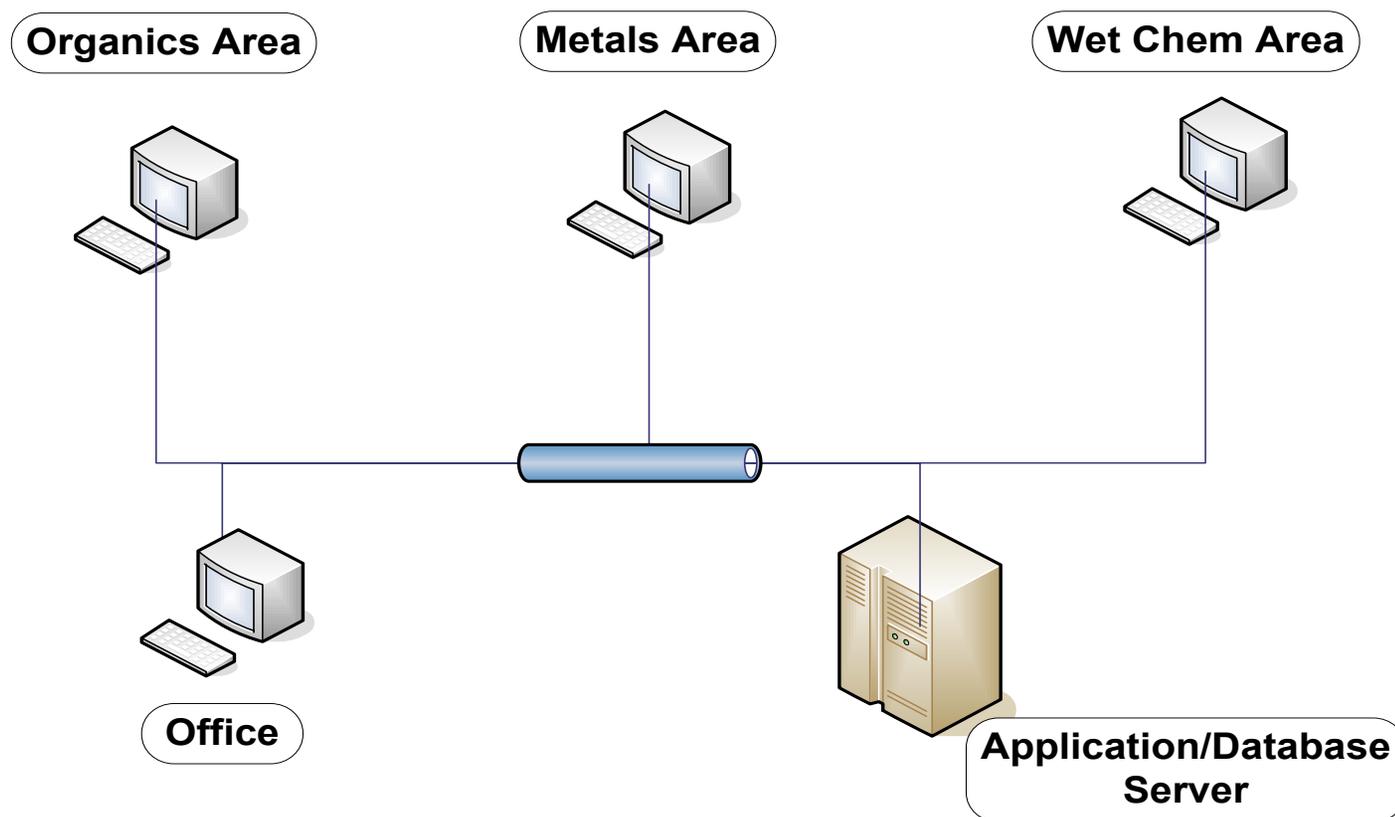
- Computing Infrastructure Evolution
  - Terminals to PC Client/Server to Thin Client
- Cloud Current Users
- Cloud Deployment Models
  - Dedicated
  - Semi-Shared
  - Shared-Separate Tables
  - Shared-Single Set of Tables
- Securing Your Environment
  - Site Security
  - Access and Connectivity

# Software Components

- Software can be broken down into 3 components
  - User Interface
    - No matter where your application is running you have to have this in front of you
  - Data Storage
    - Component rarely seen by end users
    - Good candidate for off site location since it's rarely seen
  - Reporting Engine
    - Component rarely seen by end users
    - Also a good candidate for off site location

# History of Data Management: Terminal

- Stand Alone Instrumentation/Terminal Entry

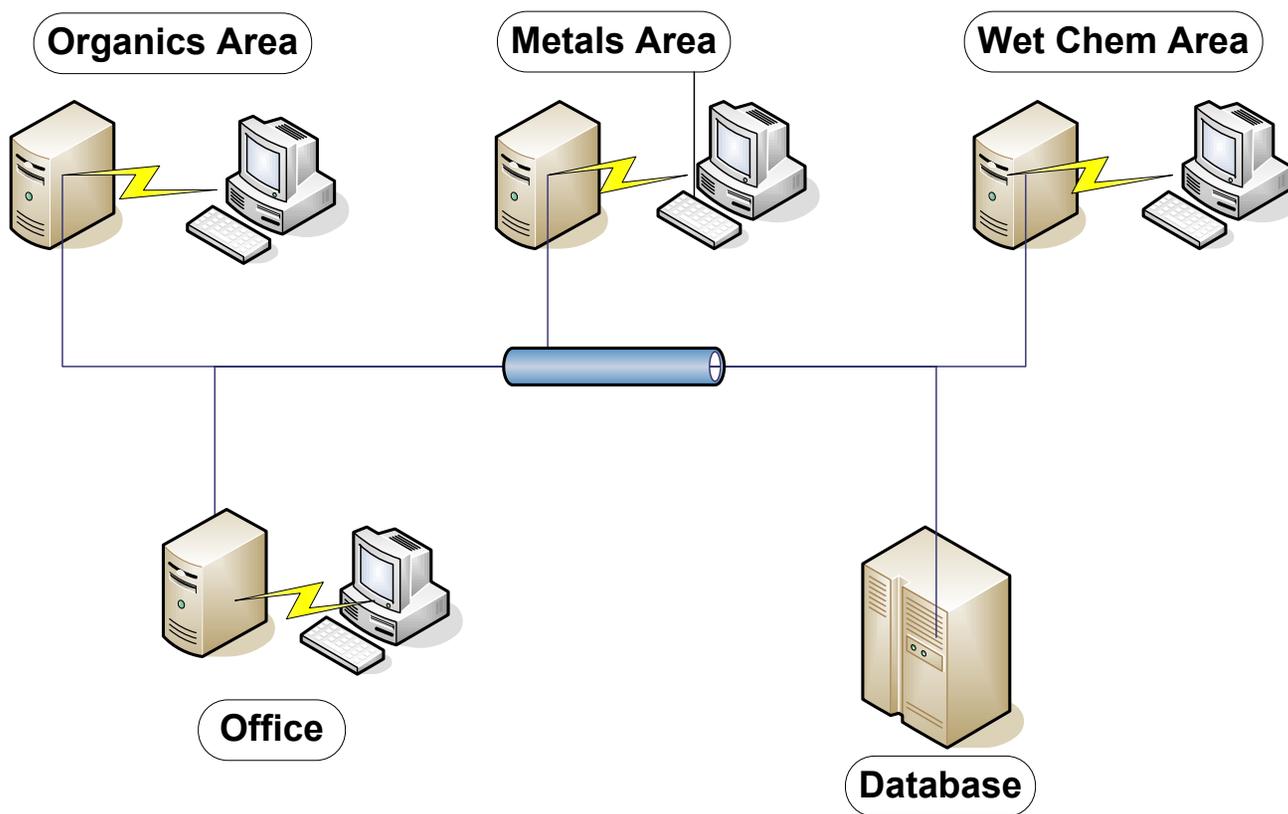


# Pros and Cons of Stand Alone and Terminals

- Pros
  - Terminals were Inexpensive
    - At least in comparison to PC based computing
  - Allows for distributed Data Entry and Access
  - Data Centrally stored on Server
- Cons
  - Terminals were dumb, unable to perform any functions or programs
  - Instruments were unable to communicate with terminals so you still needed a PC
    - This required data re-entry and increased potential error
  - Requires a robust server and connectivity within the operation

# History of Data Management: Client/Server

- Client / Server Diagram

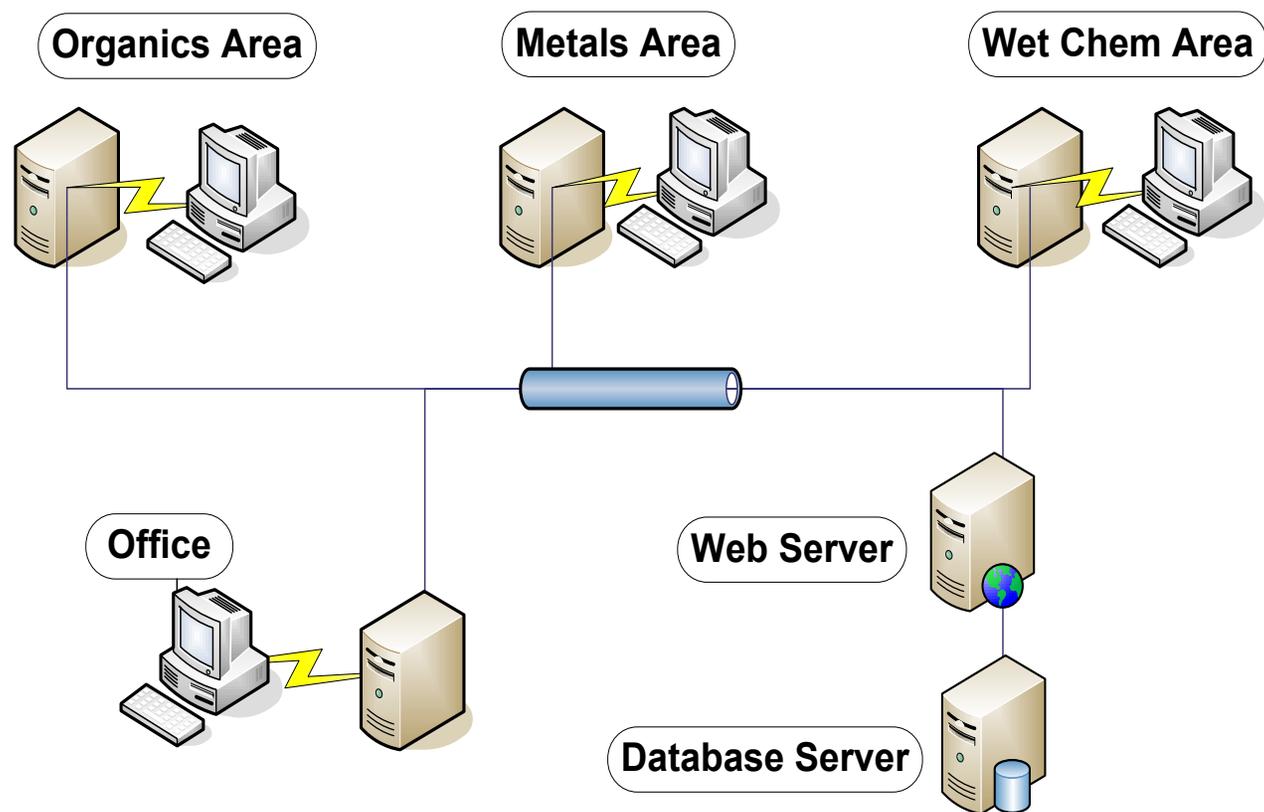


# Pros and Cons of Client / Server

- Pros
  - PCs are inexpensive
  - Distributed processing power
  - Instrumentation and PCs could communicate over networks
  - Numerous Programs available for network or client specific install
- Cons
  - Have to install programs on the PC
    - Occasionally lead to software incompatibility
    - Updates require deployment to clients which can lead to incompatibilities
  - Security information often resides on the Client
  - Requires a server and connectivity within the operation

# History of Data Mgmt: Thin Client/Server

- Thin Client /Server Diagram



# Pros and Cons of Thin Client / Server

- Pros
  - PCs are inexpensive
  - Distributed processing power (may be limited)
  - Instrumentation and PCs could communicate over networks
  - Numerous Programs available for network or client specific install
  - Potentially no client installs/updates as it uses the browser for its interface.
  - No stored security credentials on the client.
- Cons
  - More server intensive processing as client processing may be limited
  - Still Requires a server and connectivity within the operation

# Enter Cloud Computing

- A software or data storage solution that resides outside of local network
  - Other terminology
    - Cloud Computing/Hosted Solution/SaaS
- Servers are Virtualized Machines
  - Can be reconfigured quickly
- Very similar to thin client / server
  - Same browser based client
  - Hardware does not exist within domain

# Cloud: Virtualization

- Allows the use of existing Server Hardware in order to create >1 virtual Operating System Environments
  - Most efficient use of existing Hardware
  - Adjust on the fly
- Applies to the subsequent deployment models

# Cloud Users

- More than likely, everyone currently uses The Cloud today.
  - Individuals
    - Hosted email: Hotmail, Yahoo, Gmail
    - Google docs
    - Financial Management
      - On-Line Banking
      - Brokerage Accounts
  - Business
    - Office 365, Google Docs

# Expectations of The Cloud

- Advantages
  - Cost
    - Possibly no hardware to purchase/install/network
    - No IT overhead to manage the hardware
  - Fast Deployment
    - With internet connectivity you can use the application
  - Reliability
- Disadvantages
  - Where is my data?
  - Limited Control
  - Security

# Cloud Application Deployment Models

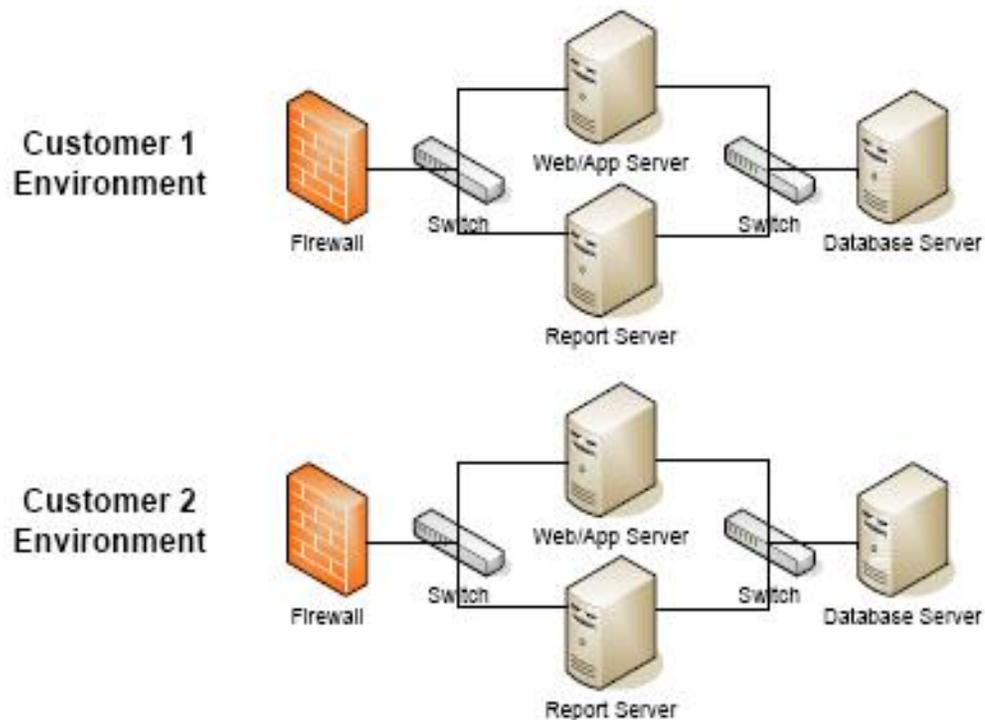
## Large Scale Applications(LIMS)

### Virtualization Models:

- Dedicated
- Semi-Shared
- Shared-Separate Tables
- Shared-Single Set of Tables
- Virtualized

# Cloud: Dedicated Host

- A separate set of servers is setup for each customer. These are the same set of servers that the you would have installed internally. The administration is done by locating facility. Also called Co-Locating (CoLo)



# Dedicated Host Pros and Cons

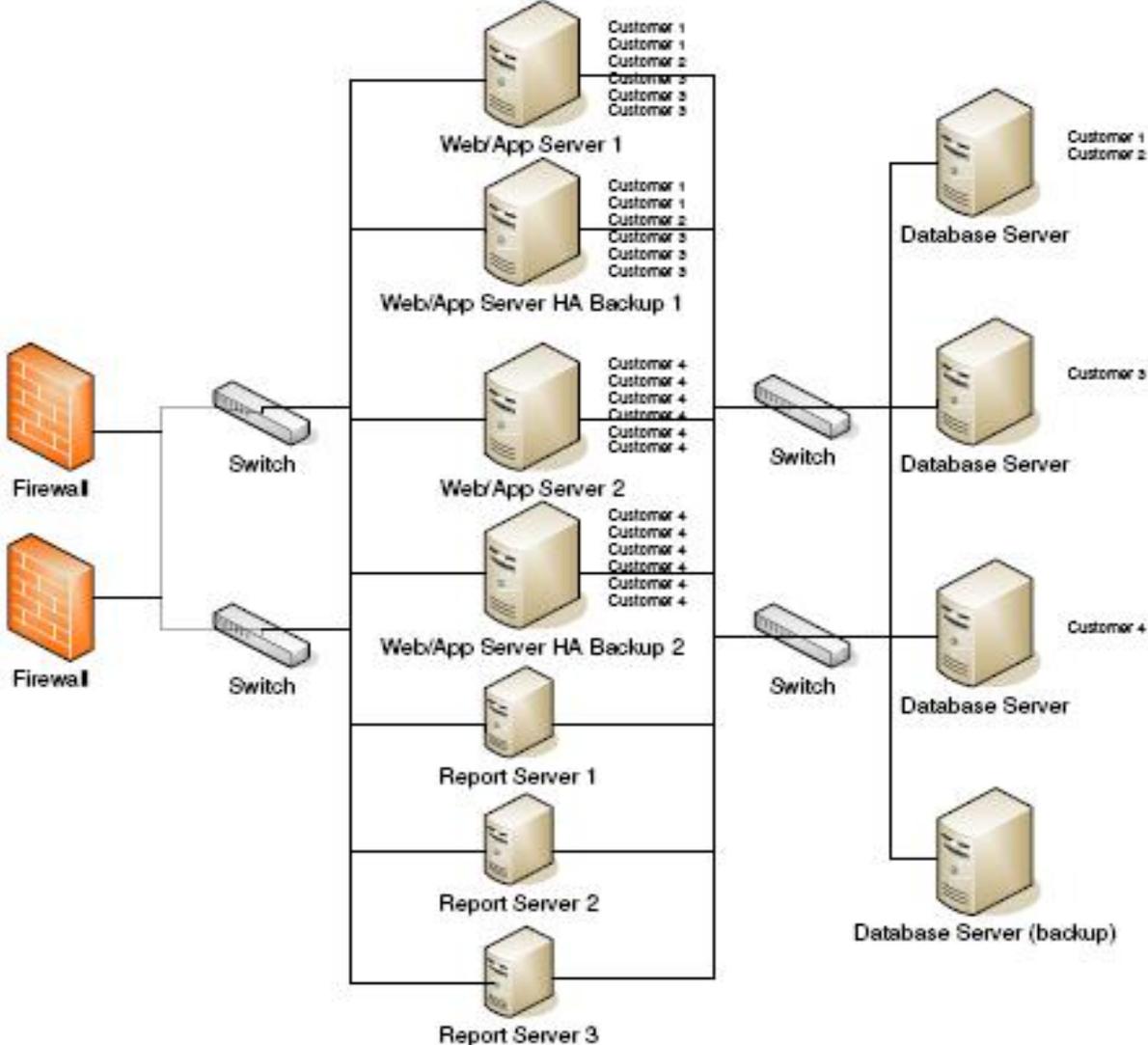
- **Pros**

- Customers avoid having to administer servers
- Customers can control the timing of upgrades
- Customers can control the timing of patches
- Easier to support application customization

- **Cons**

- The Price/Cost is Very High
  - Same Servers + Data Center + Server Hardware Administration + Server
  - App Administration + Margin = Very High Price
  - Not an economical solution
- Scalability is dependent on hardware
- Potentially Subject to reliability problems

# Cloud: Semi-Shared Diagram



# Semi-Shared Pros and Cons

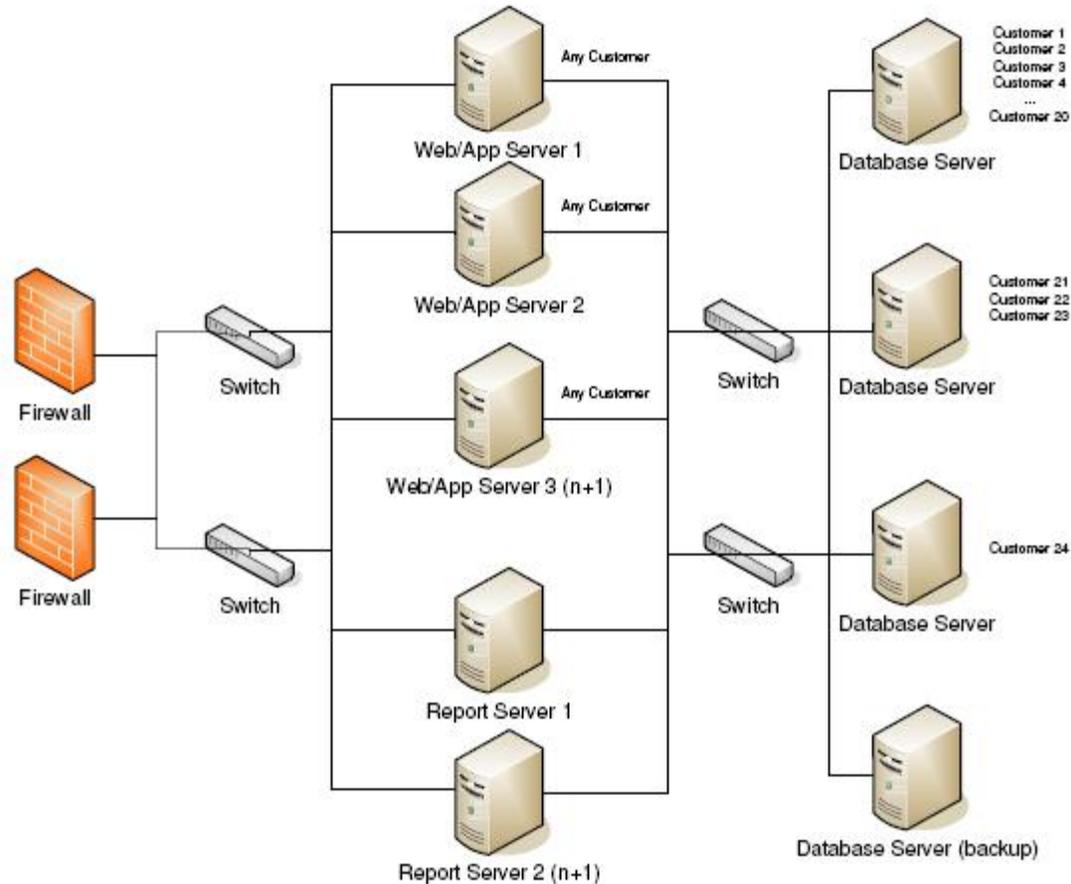
- **Pros**

- Slightly more efficient than the Dedicated model leading to a decreased cost
- Efficiency may allow for additional scalability
- Customers can control the timing of upgrades and application patches to some degree

- **Cons**

- Most Expensive Shared options
- More redundant servers are required for high availability
- Multiple software versions must be supported since each customer installs, upgrades and patches at different times
- Due to the diversity, less opportunity for automation
- More administration manpower is required than with other shared models

# SaaS: Shared Separate Tables Diagram



# Shared Separate Tables Pros and Cons

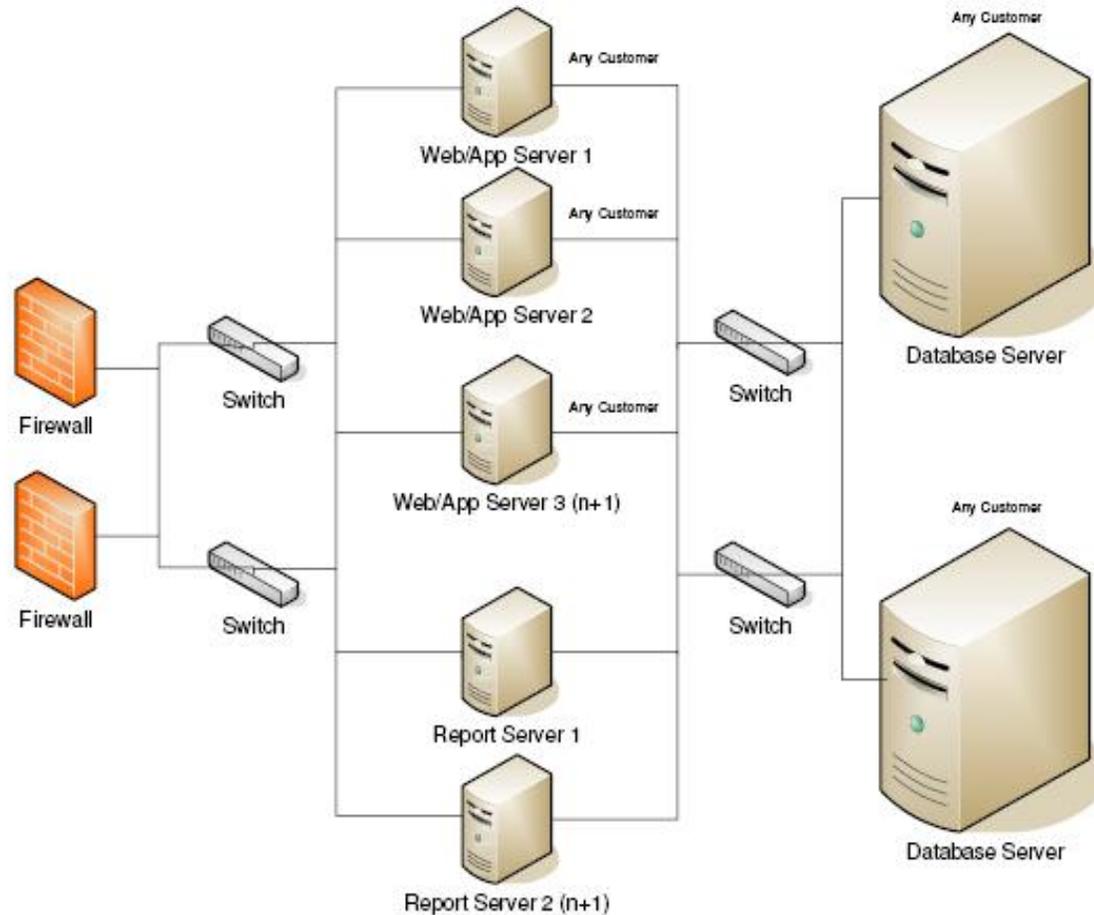
- Pros

- More efficient than the Semi-Shared model
- Requires very few administrators
- Allows for support of just a few versions, usually 3 or less
- Avoids customer security concerns over commingling data in the same set of tables

- Cons

- Multiple versions are supported making support more difficult
- Not as efficient as having multiple customers in the same set of tables
- For many applications, customization can be difficult or impossible to support.

# SaaS: Shared Single Table Set



# Shared Single Table Set Pros and Cons

- **Pros**

- Efficiency allows for true scalability at the lowest cost
- Requires fewest administrators
- Allows for support of just one version keeping code base small reducing error potential

- **Cons**

- Potential security concerns with data held in the same set of tables as other customers
- Upgrading customers all at one time carries higher risk
- Customers have no control over the timing of upgrades
- Customization is normally impossible to support.
- Backing up data uniquely for customers is more cumbersome in more complex applications – laboratory data is a good example

# Securing Your Environment

- The Goldilocks Principle
  - Too Many Hoops vs. Too Few
- Data Transmission and Encryption
  - RSA: The basis of HTTPS / SSL / TLS
    - Ron Rivest, Adi Shamir, and Leonard Adleman at their Manischewitz Moment<sup>(1)</sup>
    - Authentication and Encryption
    - Secure Channel over Insecure Network
    - 3<sup>rd</sup> Party Certificate Validation

Source: (1) Michael Calderbank (2007), "The RSA Cryptosystem: History, Algorithm, Primes"

# Securing Your Environment (Cont.)

- Application Access Controls
  - Inherent Operating System
  - IP Whitelisting
  - Client Certificates
  - Application Specific Security

# Questions and Answers

- Q&A