From Garage-Band to World Tour: Technical, Security, and Scalability Challenges of Migrating a Web-Based Program Management Tool from Workgroup-Level to Enterprise-Class in 24 Months

ICCRTS - Sept 2004

Helen Rico & Fred Hall
Air Force Research Laboratory
Information Directorate
Rome, NY

Mike Maciolek
Northrop Grumman IT
Rome, NY

Presentation Outline

- Introduction
- Evolution of the Web-Based Architecture
- Security Considerations
- Application Scalability
- Conclusion

Information Directorate Background

- Headquartered in Rome, NY
- Formerly Rome Air Development Center, then Rome Laboratory, before becoming part of AFRL
- Mission:

  The advancement and application of Information Systems Science and Technology to meet Air Force unique requirements for Information Dominance and its transition to aerospace systems to meet warfighter needs.

- Our Business is Science

Introduction

- There was a need to:
  - Report information accurately and timely without retyping
  - Electronically create Laboratory Management Review forms
  - Have engineers and scientists return to R&D tasks in lieu of admin type duties
- The goal is to make reporting quick and easy for the Program Manager (scientists/engineers) and provide secure access to needed effort or program information.

Introduction (cont'd)

- Web-Based Program Management Tool "JIFFY"
- Accessible via any Web browser capable of 128-bit encryption
- Pulls Data from AF Standard Systems
- Accessible by non-.mil Domains
- Two and one-half year transition from Workgroup to Enterprise Level

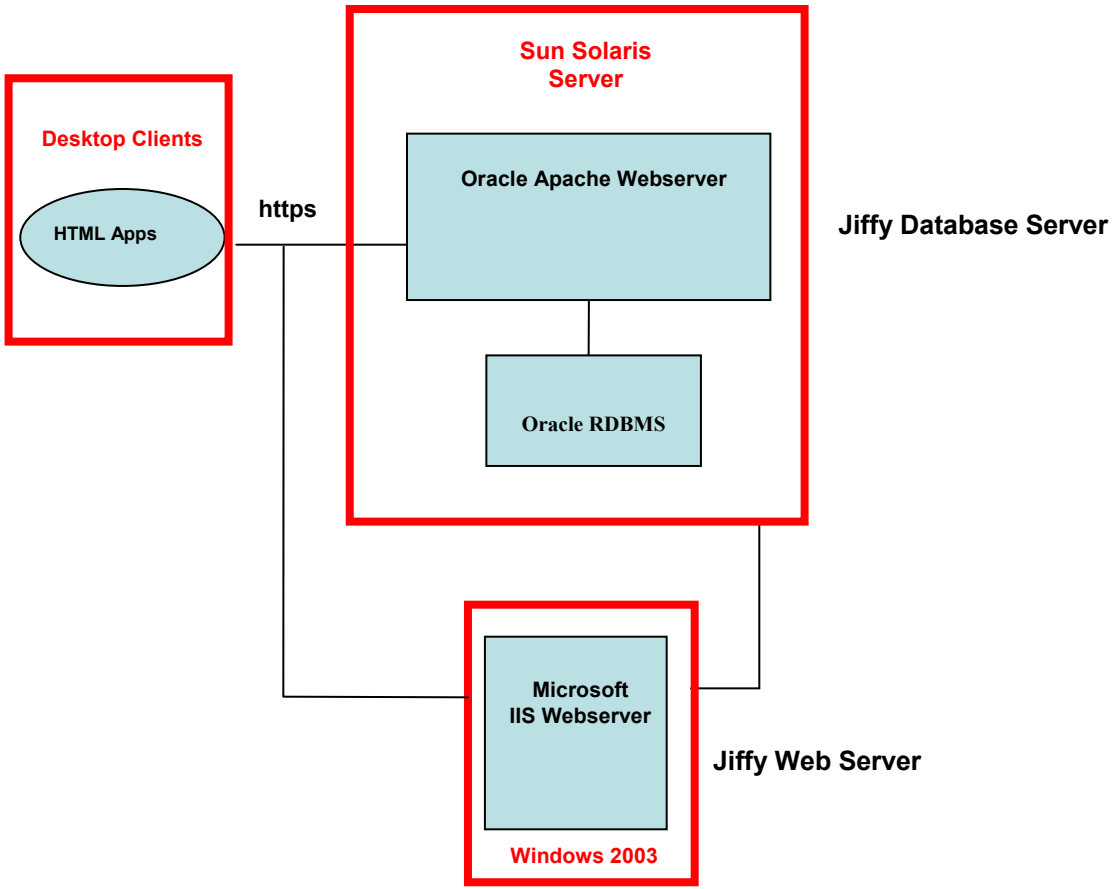
Presentation Outline

- Introduction
- Evolution of the Web-Based Architecture
- Security Considerations
- Application Scalability
- Conclusion

Architecture Basics

Two Main Pieces

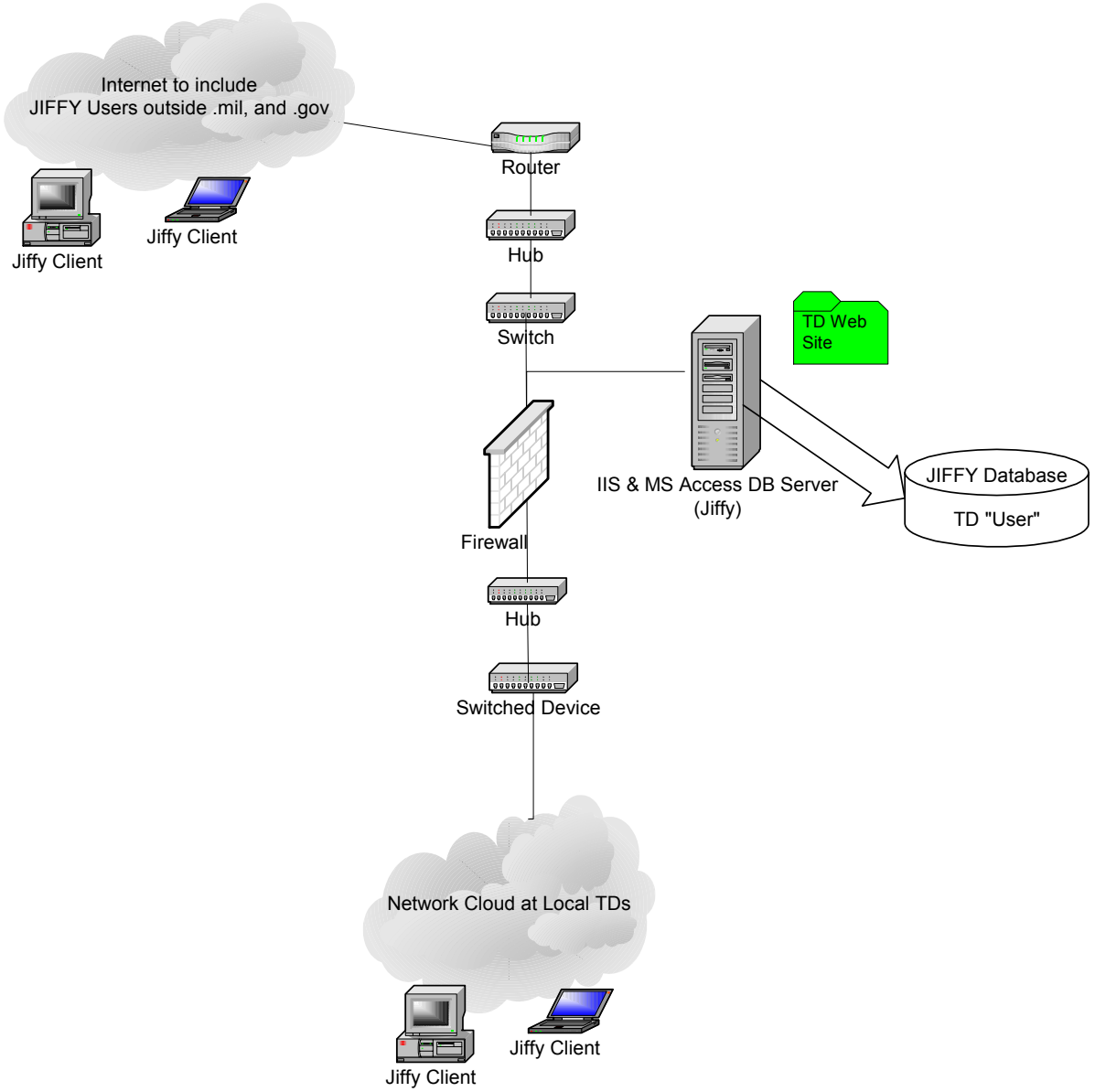
- Web Server
  - Handles user input and Graphical Display of information
- Database Server
  - Houses information gathered from AF Standard Systems and e-documents related to research programs



Workgroup Architecture

- Served a Handful of Users
- Entirely Windows-Based
  - IIS Web Server
  - MS Access Database
  - Same Physical Computer
  - e-documents stored in Windows File System
- Outside Base Firewall to Facilitate .com/.edu Access
- Development Staff: Two Part-Time Engineer/Programmers, One Full-Time Programmer

Workgroup Architecture (cont'd)



Workgroup Architecture (cont'd)

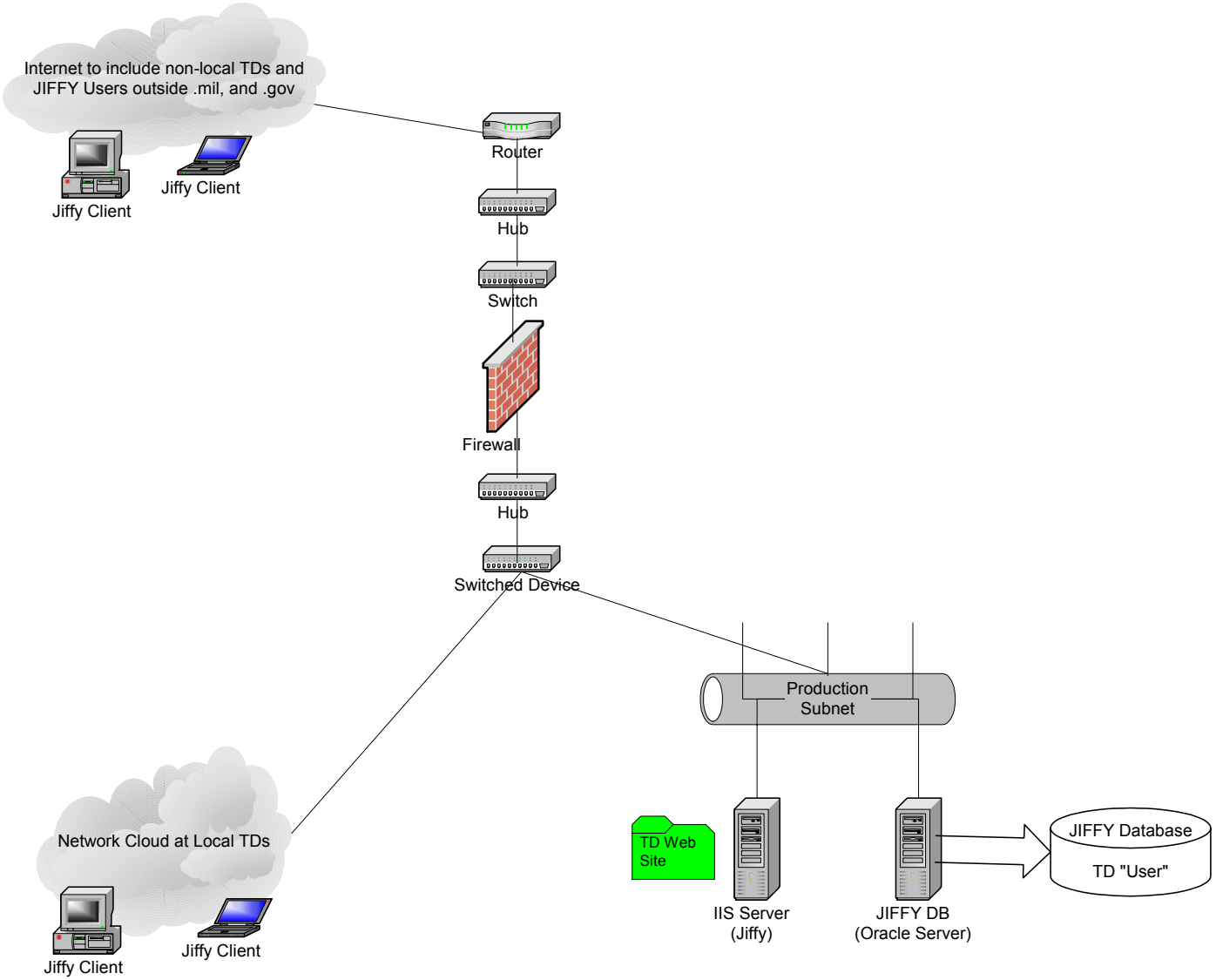
- **Advantages**
  - Good Performance
  - Low Maintenance Costs
  - Quick Development Cycle

- **Disadvantages**
  - Security Concerns
    - Computer Not Protected by Base Firewall
    - IIS, and MS Access vulnerabilities
    - Windows File System storage of e-documents
  - Not Scalable

Directorate-Wide Architecture

- Few Hundred Users
- Windows-Based IIS Web Server
  - Also used for storage of e-documents
- Sun Solaris Oracle Database Server
- Inside Base Firewall to Enhance Security
  - Firewall Rules Used to Facilitate .com/.edu Access
- Development Staff:
  - Six Programmers
  - Two Part-Time Program Managers
  - One Application Support Person
- Nine Month Development Timeframe

Directorate-Wide Architecture (cont'd)



Directorate-Wide Architecture (cont'd)

- **Advantages**
  - Higher Level of Security
  - More Robust and Scalable
  - Quick Development Cycle

- **Disadvantages**
  - Security Concerns
    - Windows File System storage of e-documents
  - Slight Performance Degradation
    - Initially Was Large (will be discussed later)

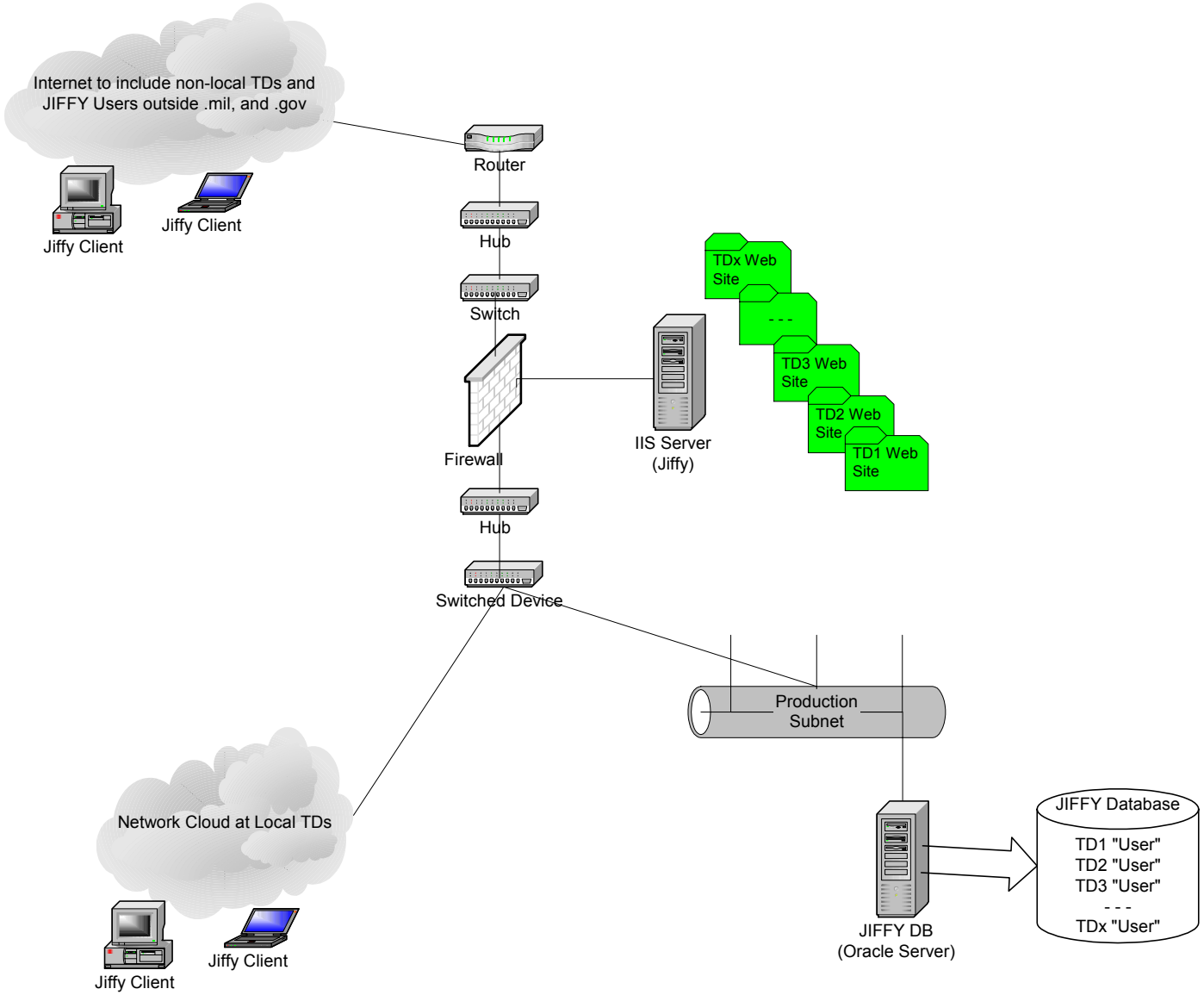
Enterprise-Wide Architecture

- Few Thousand Users Geographically-Dispersed across CONUS
- Windows-Based IIS Web Servers
  - On separate Firewall “leg” (Extranet)
  - Three Physical Servers to Share Load
- Sun Solaris Oracle Database Server
  - e-documents stored in the database
- Firewall Rules Used to Facilitate .com/.edu Access

Enterprise-Wide Architecture (cont'd)

- **Development Staff:**
  - Nine Programmers
  - Two S/W Testers
  - QA Person
  - Part-Time S/W Security Person
  - Program and Deputy Program Managers
  - Two Application Support People
  - Short-Term Paid Consultant
- **11 Month Development Timeframe**

Enterprise-Wide Architecture (cont'd)



Enterprise-Wide Architecture (cont'd)

- **Advantages**
  - Higher Level of Security
  - More Robust and Scalable
  - Performance Improved

- **Disadvantages**
  - Longer Development Cycles
  - Higher Maintenance Requirements

Presentation Outline

- Introduction
- Evolution of the Web-Based Architecture
- **Security Considerations**
- Application Scalability
- Conclusion

Security Considerations

- Server Access
- Application Access
- User Roles and Permissions
- Traceability

Security Considerations (cont'd)

- **Server Access**
  - **Contractor Access to Their Program Info is Crucial Feature**
    - **Requires .com/.edu Access to Server**
  - **DMZ (Extranet) Established to Facilitate Secure non-.mil Domain Access**
  - **Anti-Hacking Measures Incorporated Against; SQL Injection, Anonymous File System Access, Undesired Execute Privileges, URL Hijacking**
  - **e-documents Moved to Database Diminishes Exposure**

Security Considerations (cont'd)

- **Application Access**
  - Trusted-Agent Account Nomination Process
  - Must Be US Citizen or I-551 "Green Card" Holder
- **User Permissions**
  - Role-Based Permissions
  - Row-Level Data Security

Security Considerations (cont'd)

- **Traceability**
  - Track User Activity in Critical Application Areas
  - Track Data Changes in Critical Application Areas
  - Web Server Logs Track User Activity Related to File Access
  - e-documents Moved to Database Diminishes Exposure
  - Allows Post-Mortem Analysis on Hacks
  - Assists in Debugging and Help-Desk Problem Resolution

Presentation Outline

- Introduction
- Evolution of the Web-Based Architecture
- Security Considerations
- **Application Scalability**
- Conclusion

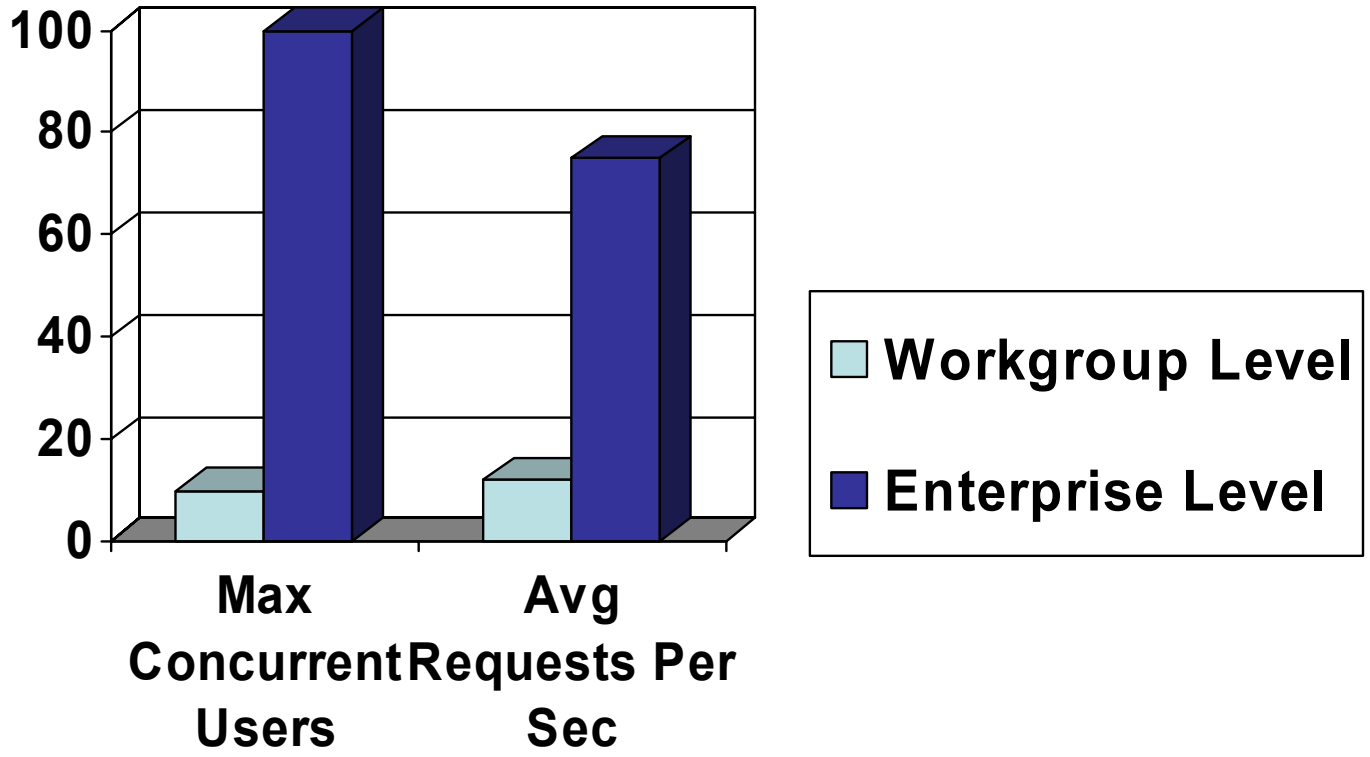
Application Scalability

- Performance Issues
- Database Conversion
- Design Testing
- Continuing Improvements

Application Scalability (cont'd)

- **Application Performance**
  - **Response Time**
    - Time for Web Server to Return Request
    - Average Number of Requests Served per Second
  - **Concurrent Users**
    - Number of Simultaneous Users that can Access a System
    - Normal Use Testing
    - Load Testing
  - **Problem Areas**
    - Early Development not Geared Toward Enterprise Scalability
    - Migration Time Constraints Led to Trade-offs

Application Scalability (cont'd)



Application Scalability (cont'd)

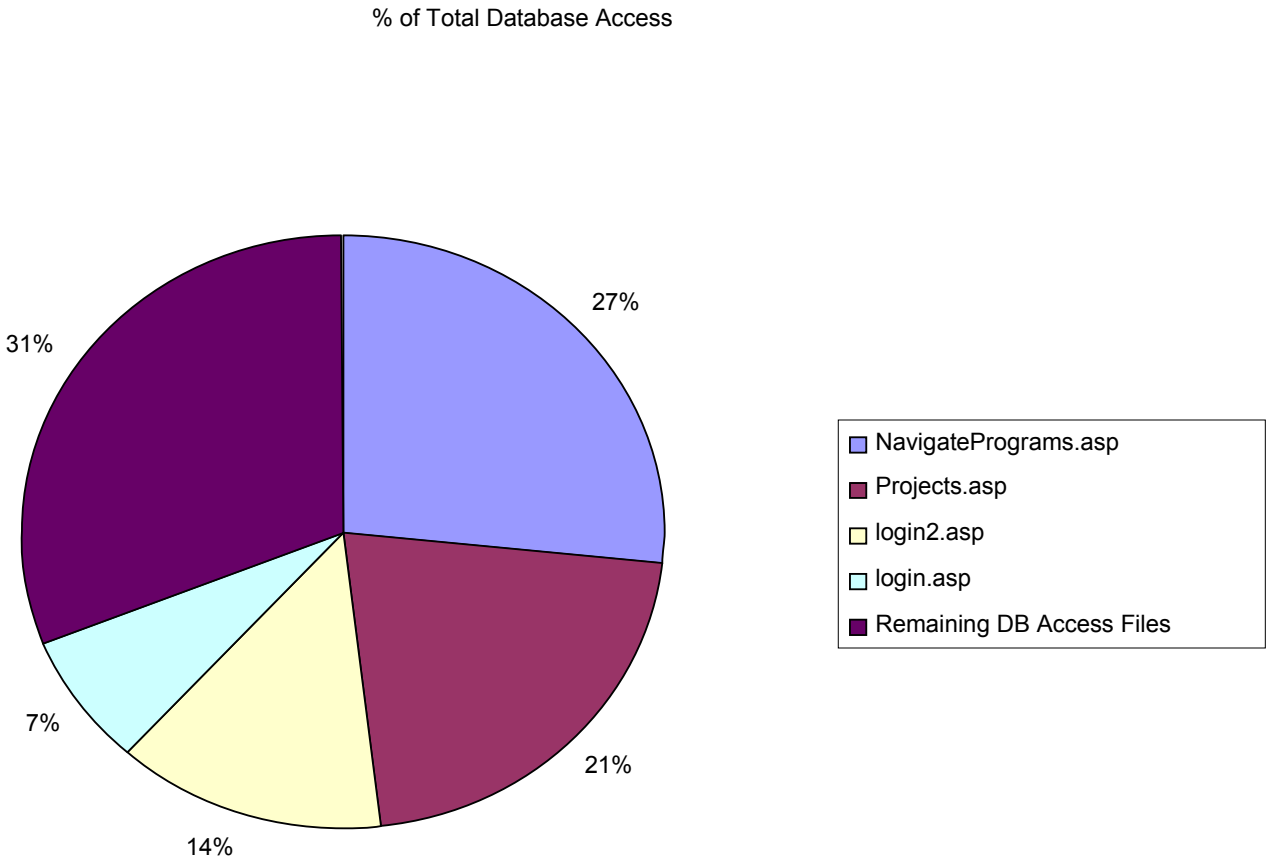
- Database Conversion
  - Interface Decisions
    - First Choice – MS Generic ODBC – poor performance
    - Moved to Oracle OO4O – significant performance gains
  - Stored Procedures
    - Moved Database Access Logic Into Stored Procedures
    - Consolidate Related Activities into APIs
      - Helps Developers
      - Allows Data Feeds to Properly Interact with System Logic
    - Use Native Database Routines for Speed and Functionality

Application Scalability (cont'd)

- Design Testing
  - VBScript Classes for Encapsulation
    - Early Development Decision to Encapsulate DB Access Logic in VBScript Classes
    - Poor Performance but Easily Maintainable
  - Profiling Components
    - Ported VBScript Classes to COM Classes
      - Compiled Executables
      - Superior Logging Capabilities
      - Helps Determine Data Access Bottlenecks
  - Automated Testing
    - Reliability and Regression Tests Developed
    - Load Tests Conducted for Performance Measurement

Application Scalability (cont'd)

Continuing Improvements



Application Scalability (cont'd)

- **Continuing Improvements**
  - **Make Improvements Based on Application Profiling**
  - **Eye on Performance and Security**

Conclusions

- Took Application from Workgroup to Enterprise in 24 Months
- Meets the Needs of the Diverse User Community
- Providing Help Desk and Hands-On Training is Crucial to Acceptance
- Well-Positioned for Long Life in the Enterprise
- Lessons Learned Applicable to any Web-based Application Development Effort

Comments/Questions?