New England Electronic Commerce Users' Group

Building an eCommerce Solution Architecture

Vin D'Amico
President
vin@damicon.com
January 2002

Goals

- Introduce Damicon, LLC
- Define “Architecture”
- Show Sample Architectures
- Share Best Practices
- Compare Web Services

DAMICON

Who is DAMICON, LLC?

- Information Technology Advisors
- IT Help Desk Enhancers
- Adjunct CIOs
- IT Change Agents

It’s not about technology, it’s about business!

DAMICON

Core Competencies

- IT Process Reengineering
- Technical Consulting
- Program Management
- Planning and Execution Methodologies
- Joint Requirements Planning
- IT Training and Mentoring
- IT Staffing

DAMICON

What is Architecture and Why is it Important?

“A common mistake that people make when trying to design something completely foolproof is to underestimate the ingenuity of complete fools.”

- Douglas Adams, Author

DAMICON

Architecture Definition

Formal description of a system at the component level.

The structure of components, their interrelationships, and the principles governing their design and evolution over time.

Why Architecture?

- Lower software development, support, and maintenance costs
- Improved interoperability and easier system and network management
- Simpler upgrade and exchange of system components
- Reduced complexity in IT infrastructure
- Flexibility to make, buy, or outsource IT solutions

DAMICON

Architecture is Critical In…

- Delivering an enterprise-computing system
- Providing control points to manage complexity
- Maintaining system integrity
- Unifying component structure
- Organizing people and processes
- Generating rules for growing the system
- Protecting an enterprise system

DAMICON

Architecture Goals

1. Accommodate Change
2. Adhere to Standards
3. Scale as Business Grows
4. Provide Full Functionality
5. Deliver Low Response Times
6. Be Reliable
7. Interoperate with Other Systems
8. Provide Robust Security
9. Be Simple to Manage
10. Service International Users

DAMICON

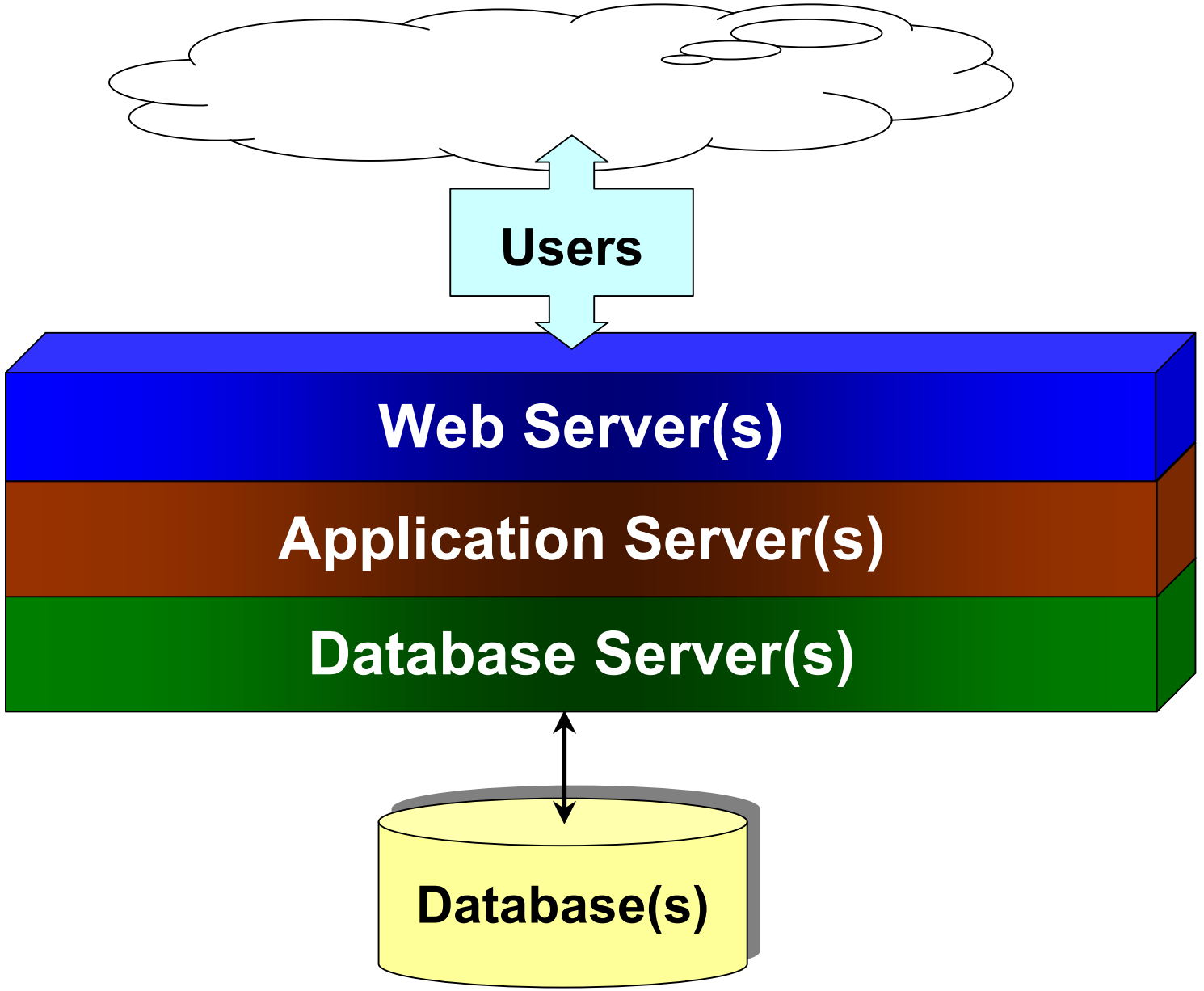
Architecture Elements

- Personal Systems
  - Desktops, PDAs, Phones, Pagers, etc.
- Network Components
  - Routers, Load Balancers, Switches, etc.
- Security Elements
  - Firewalls, Encryption, VPNs, etc.
- Servers
  - Web, App, DB, Directory, etc.
- Application Components / Web Services
  - Packaged and/or Custom
- Data
  - Local, Remote, Internal, External

DAMICON

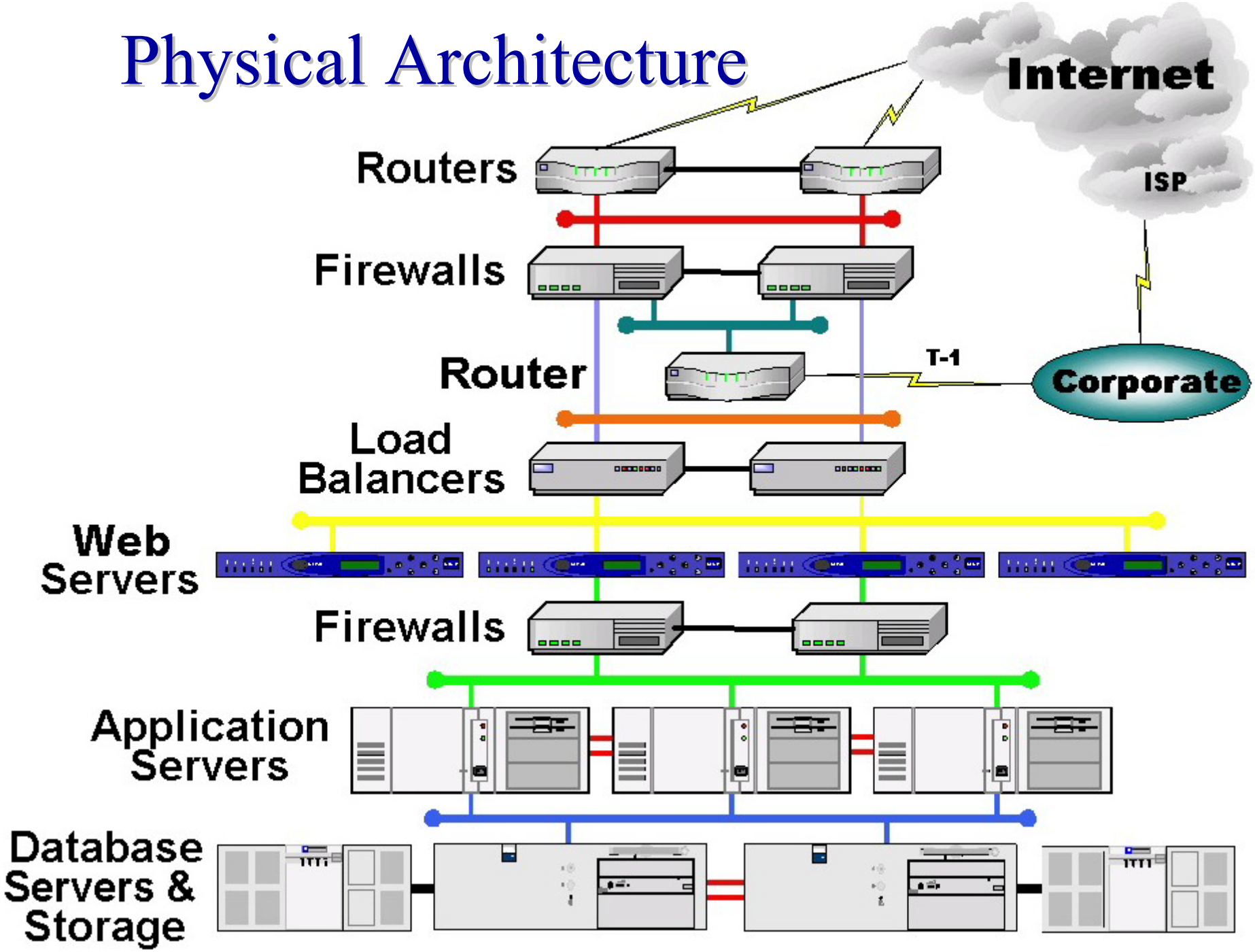
Examples of Architecture

Basic 3-Tier Architecture

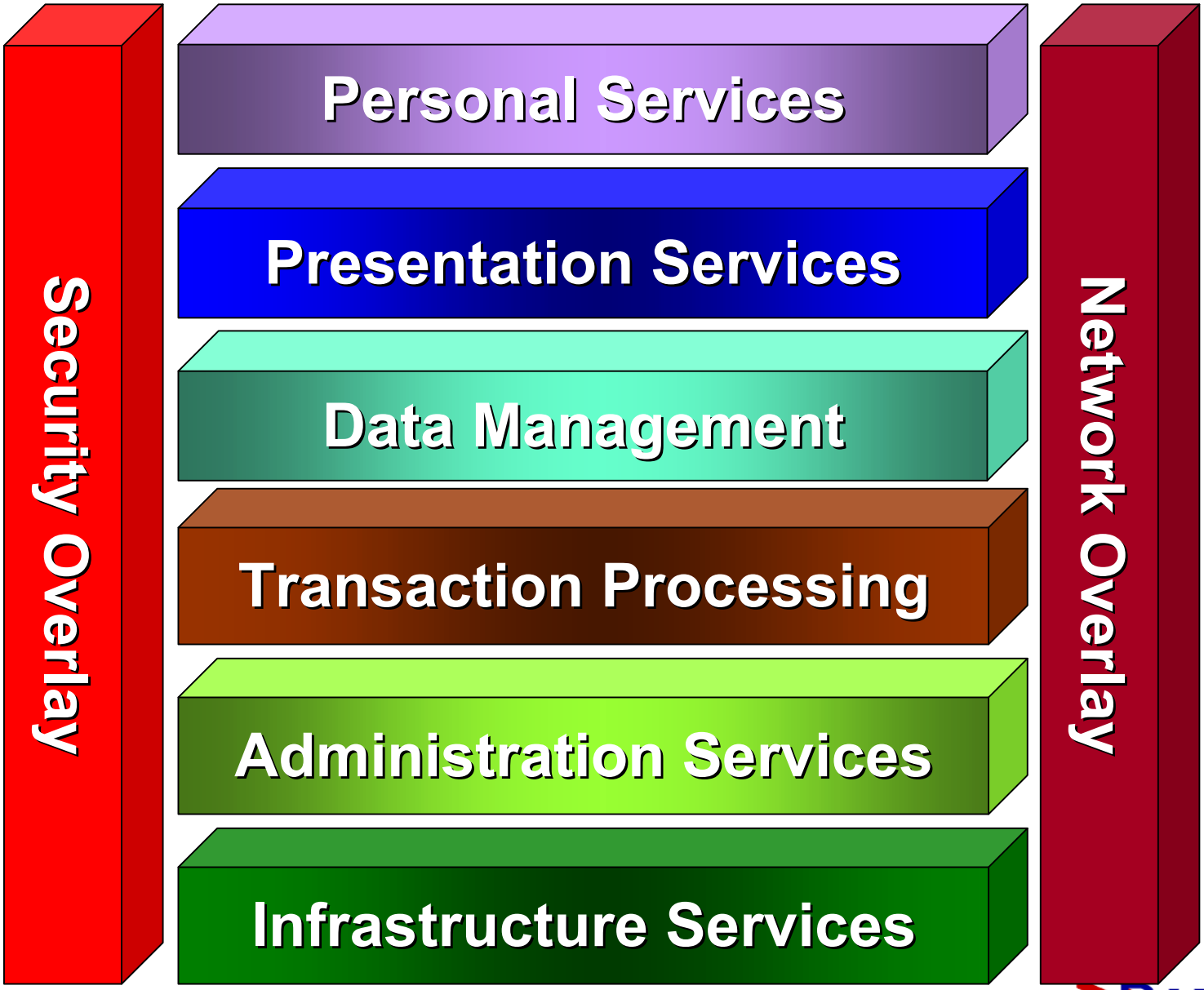


DAMICON

Physical Architecture



Complex Distributed Architecture



DAMICON

Personal Services

- Any Browser-based Device:
  - Desktop
  - Laptop
  - Handheld
  - Phone
  - Appliance
  - etc.

- Best Practices:
  - Adhere to Standards
  - Avoid Browser-Specific Features
  - Minimize Decision Logic
  - Use Java or C# for Complex Functions

Presentation Services

- Formatting Logic
- Dynamic Content Delivery
- Portlets
- Reporting
- Internationalization

- Best Practices:
  - Separate Data Retrieval from Formatting
  - Don't Mix Business Rules and Display Logic
  - See Model-View-Control and Layer Patterns

Data Management Services

- Searching
- Categorization
- Content Aggregation
- Group Collaboration
- Personalization
- Distribution

- Best Practices:
  - Identify User Types
  - Focus on User Goals
  - Consider Performance
  - See Presentation-Abstraction-Control and Chain of Responsibility Patterns

Transaction Processing Services

- Transaction Management
- Metadata Control
- Application Interfaces
- Business Rules
- Data Interchange

- Best Practices:
  - Focus on Interfaces
  - Beware Incomplete User Activities
  - Think Services
  - Do Not Hard Code Business Rules
  - See Adapter, Façade, Proxy, Observer and, Broker Patterns

Administration Services

- Directory Services (LDAP)
- System Administration
- State Management
- Session Management
- User Controls
- Rules Definition

- Best Practices:
  - Define Policies
  - Control System States
  - Anticipate Growth
  - See Command and Microkernel Patterns

Infrastructure Services

- Data Access
- Communications
- Process and Thread Management
- Sun One and MS .Net
- Content Repositories

- Best Practices:
  - Adhere to Standards
  - Understand Data
  - Model Data
  - Manage Data
  - See Abstract Factory and Mediator Patterns

Security Overlay

- Hardware Firewalls
- Software Firewalls
- SSL and WTLS
- VPN's
- Encryption

- Best Practices:
  - Establish Policies
  - Secure the Perimeters
  - Monitor for Intrusions
  - Stay Aware
  - Patch, Patch, Patch

DAMICON

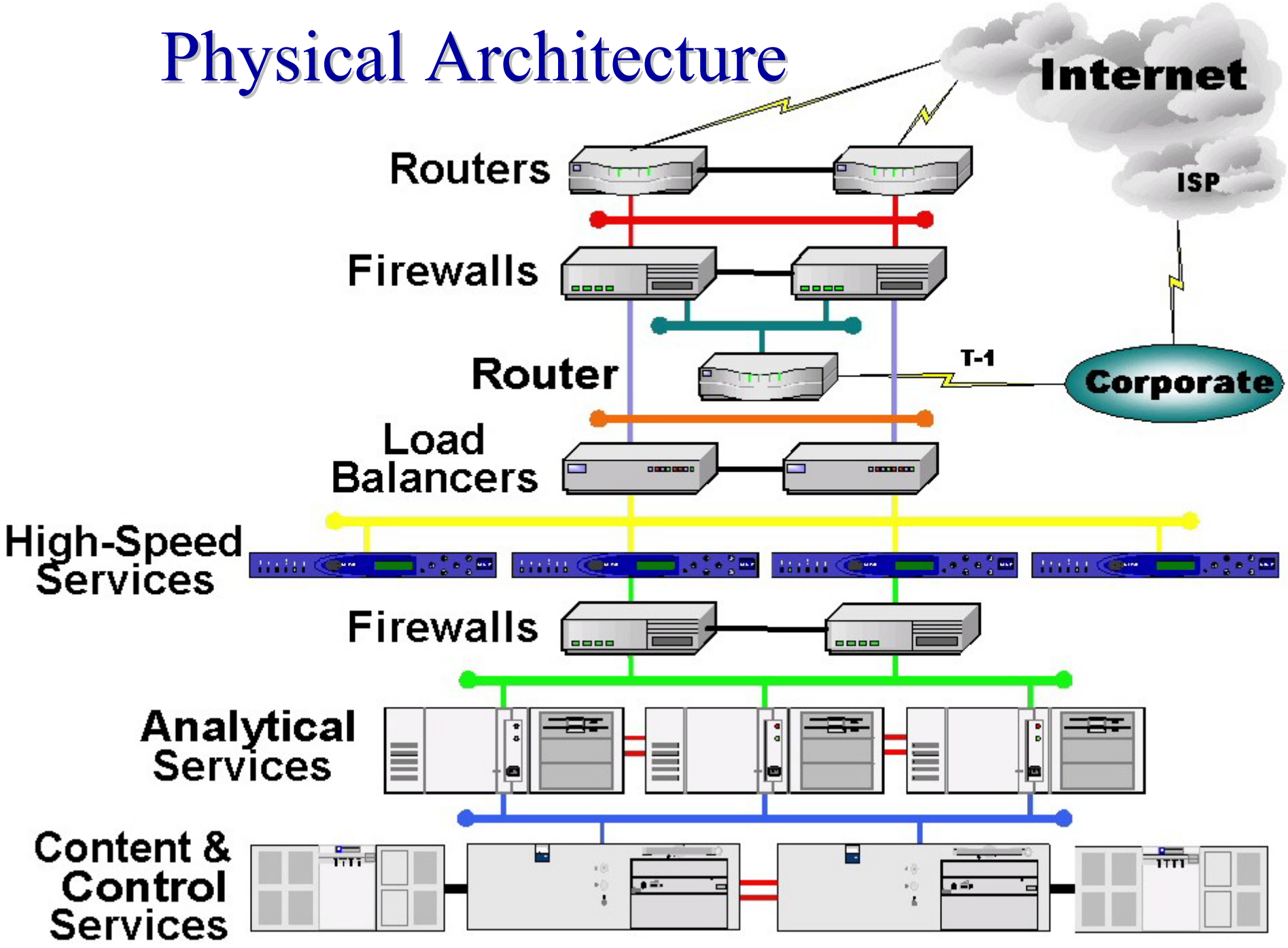
Network Overlay

- Routers
- Load Balancers
- Switches
- Gateways
- Modems
- Hubs
- Cabling

- Best Practices:
  - Separate Major Workgroups
  - Separate Major Applications
  - Constantly Review Structure
  - Document, Label, Diagram

DAMICON

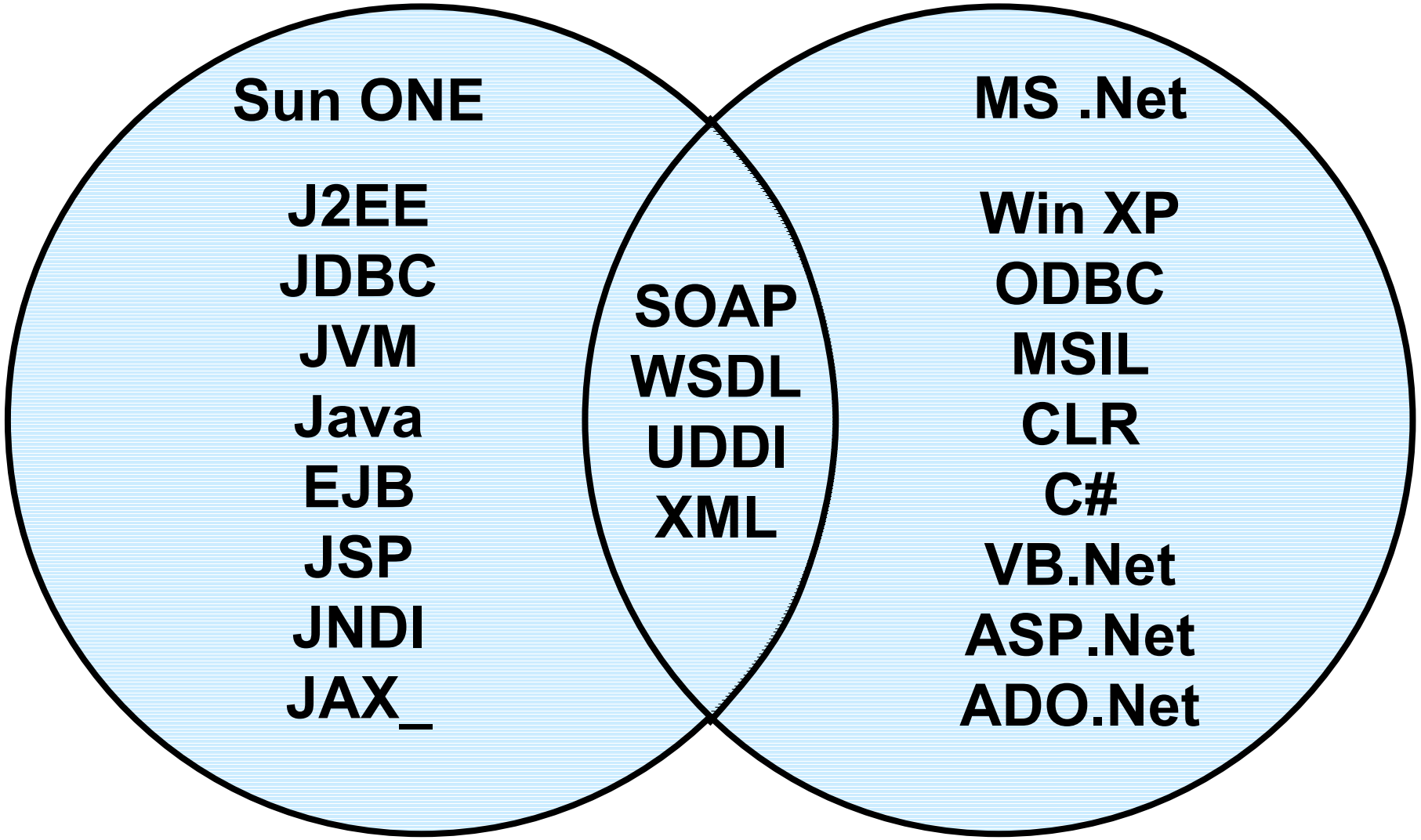
Physical Architecture



Web Services

- Major Players
  - IBM (WebSphere)
  - Microsoft (“.NET”)
  - Sun Microsystems (Sun ONE)
- Problem Space
  - Distributed Applications
  - Interoperability
- Solution Space
  - XML
  - Interfaces

Web Services
Think Interfaces!



Resources

- The Open Group (TOGAF)
  - http://www.opengroup.org
- Portland Pattern Repository
  - http://c2.com/cgi-bin/wiki?PatternIndex
  - http://c2.com/cgi-bin/wiki?AntiPatterns
- Microsoft (“.NET” initiative)
  - http://www.microsoft.com/net
- Sun (“Sun ONE” initiative)
  - http://www.sun.com/sunone
- Acronym Finder
  - http://acronymfinder.com

New England Electronic Commerce Users' Group