# Application and System Architecture

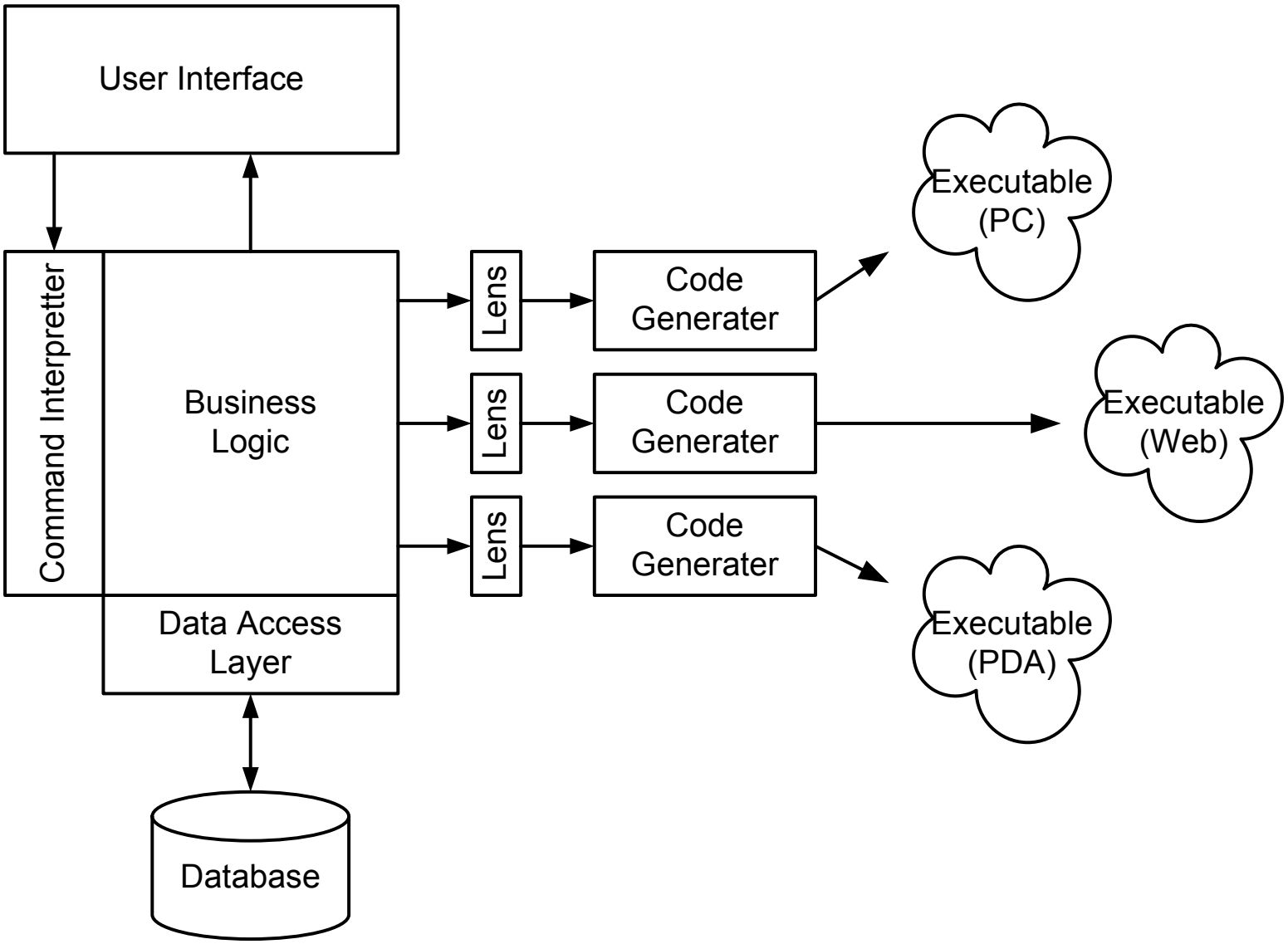
Experience and Lessons Learned

# Coupling & Cohesion

- ***Coupling*** is measure of how much the internals of one component depend on the internals of other components
  - Low Coupling means more reuse
  - Plug-N-Play
- ***Cohesion*** is measure of how much the functions/services of a component belong together
  - High Cohesion means more robust

# Responsibility Driven Design

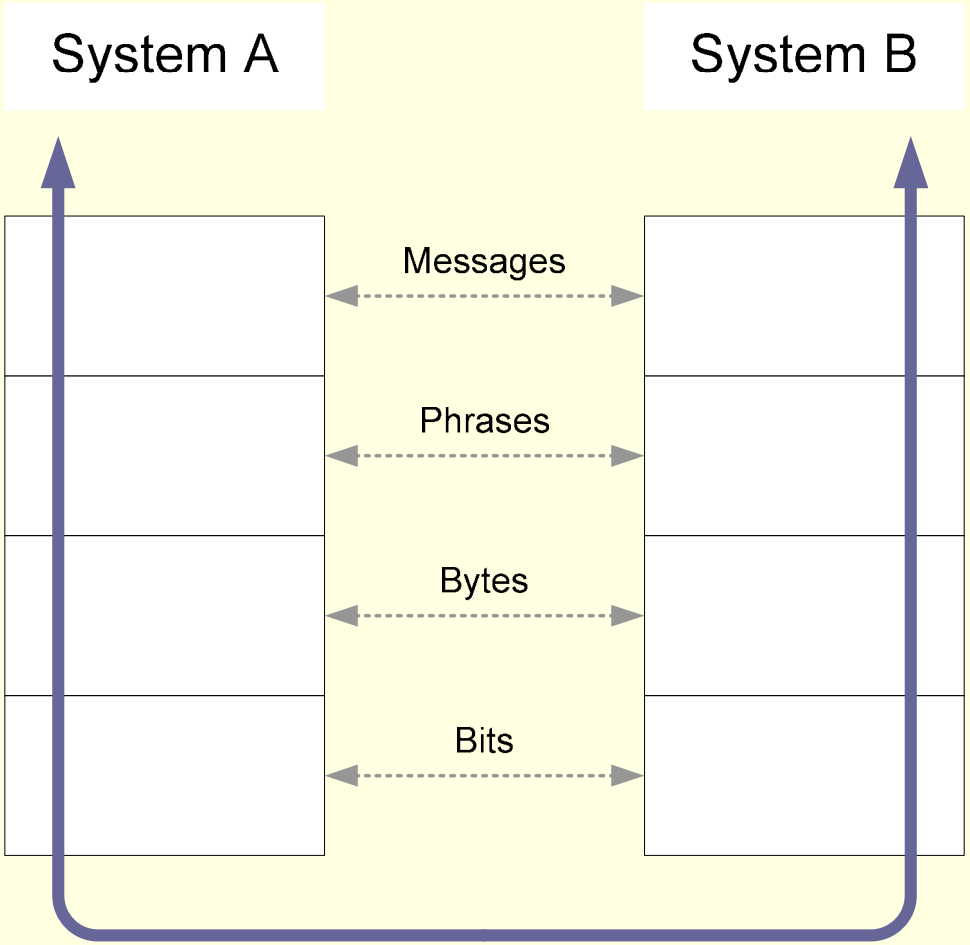
- System Decomposition
  - Divide system into component parts
  - Each with a core focus
  - Well defined interfaces
- Who is Responsible for What?
- Low Coupling, High Cohesion


User Interface
Command Interpretter
Business Logic
Data Access Layer
Database
Lens
Lens
Lens
Code Generater
Code Generater
Code Generater
Executable (PC)
Executable (Web)
Executable (PDA)

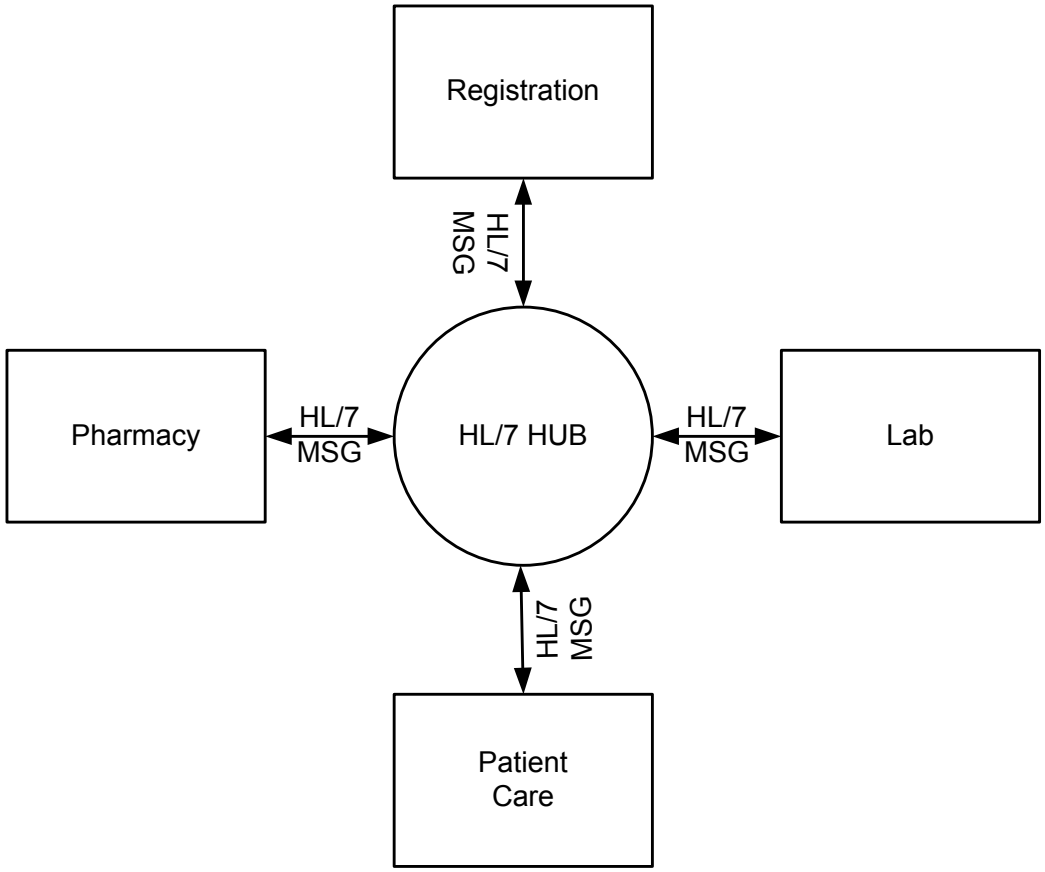
# Abstraction Layers

- Focus on the High Level ***What***
  - Don’t worry about the details.
    - That is someone else’s problem
- Recursive System Decomposition

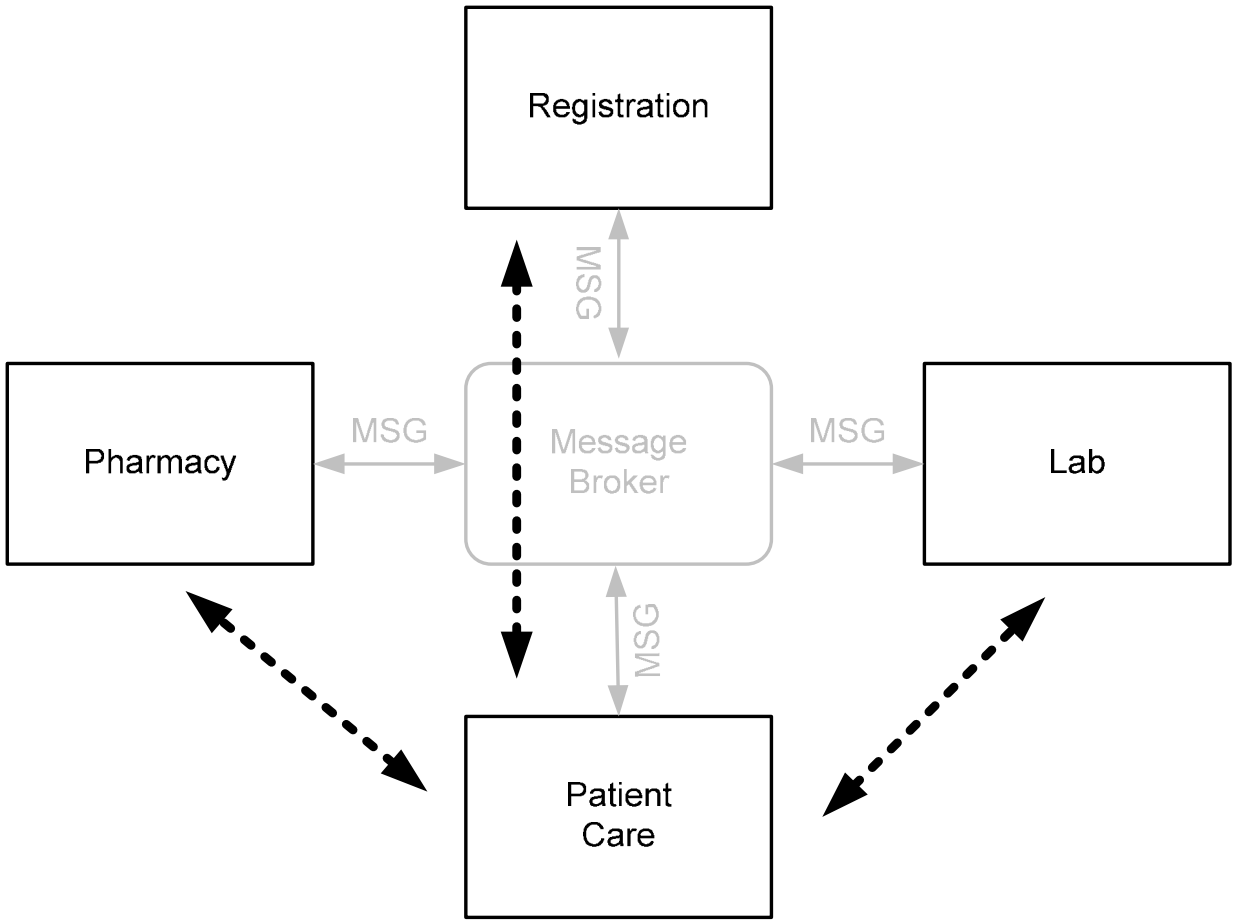
# Communications Stack (sort of)



# Physical Architecture



# Logical Architecture



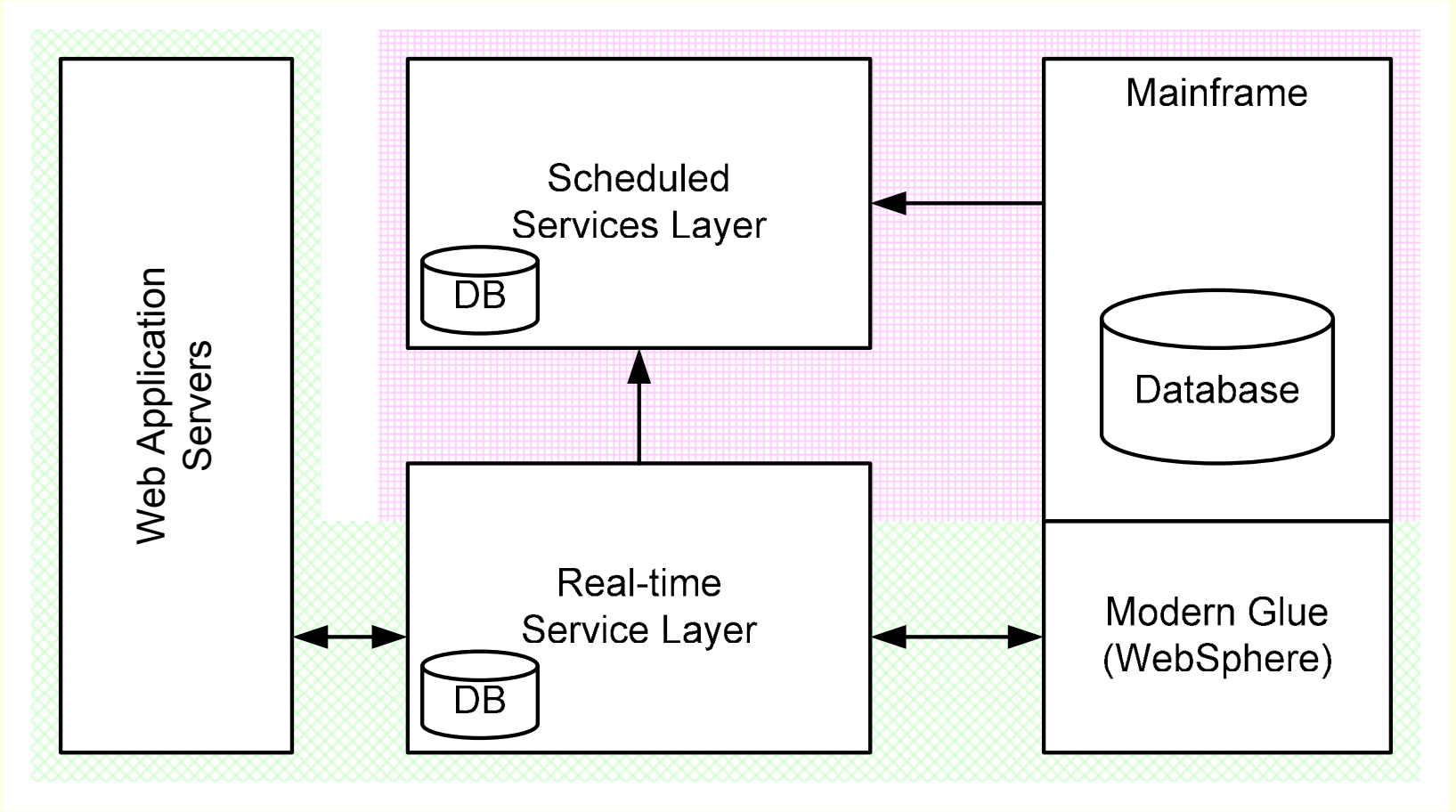
# Communication Patterns

- Tightly Bound Business Process
  - Response to User
  - Process Execution
  - Synchronous Communication (RPC)
    - Wait for answer
- Loosely Bound Business Process
  - Asynchronous Communication
    - Message-Oriented Middleware
    - Database as Middleman

# Batch Versus Real-time

- Amazon.com
  - Placing order is real-time
    - Customer needs response
  - Order fulfillment can be batch
    - Physical (offline) processes involved
- The Modern Dilemma
  - Legacy Systems often Batch
  - The World is Moving to Real-Time

# Batch Versus Real-Time



# Service-Oriented Architecture

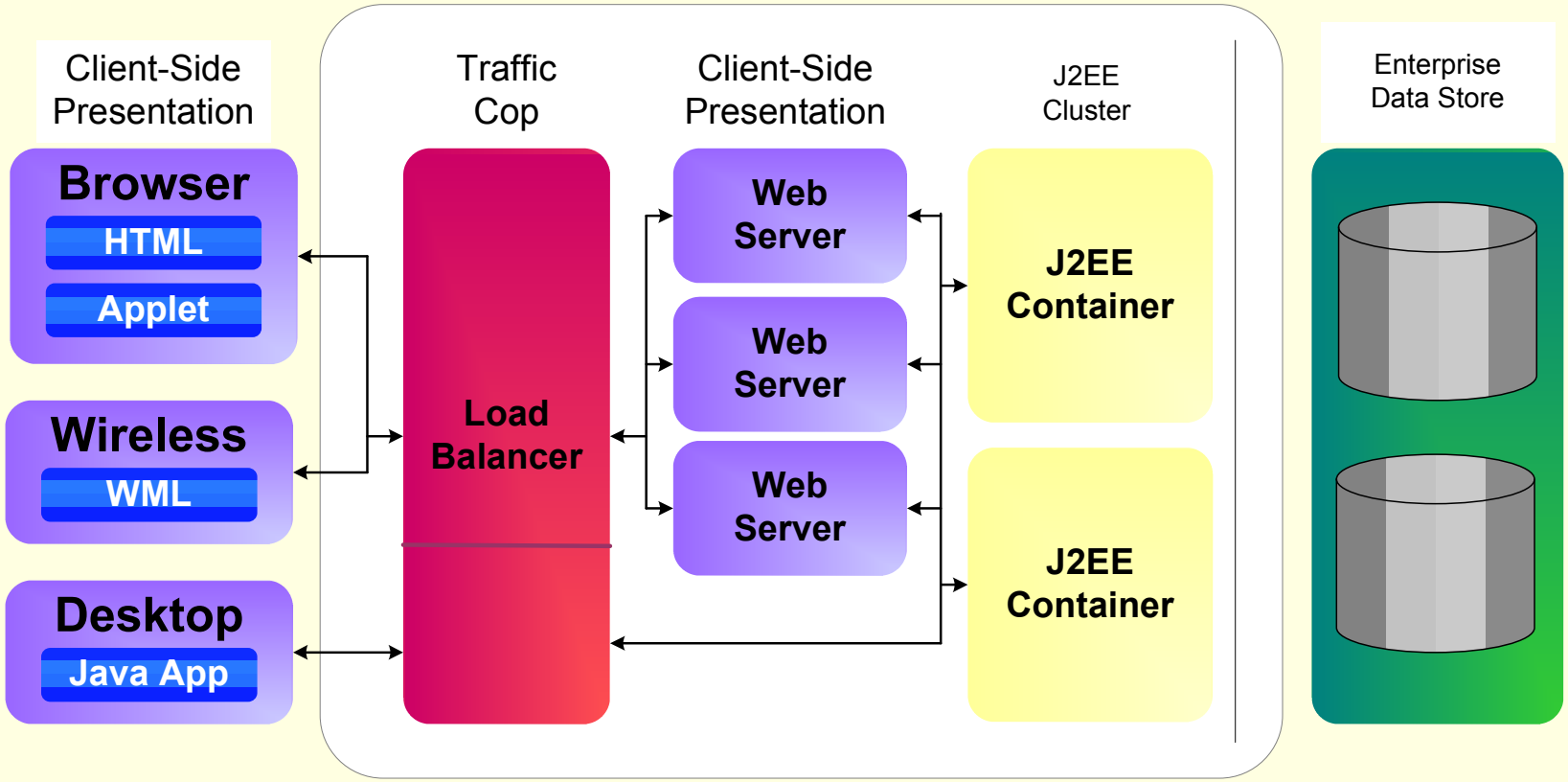
- Good as Abstraction
  - Component Architecture
  - Responsibility Driven Design
- Good as Implementation Mechanism
  - Separation of UI from Business Logic
  - Separation of What from How

# SOAP

- One Technology for Service Layer Implementation
- XML as Communication Mechanism
  - Character-Based Neutral Format
    - Business-to-Business
    - Servers with Different Technologies
  - Self-Describing Data Structures
    - VOLUMOUS!
    - Can really suck down bandwidth

# Enterprise Concerns

- Stability
- Robustness
- Redundancy
- Fail-over
- Recovery

- Organizational Capabilities
- Organizational Biases

# Enterprise Concerns