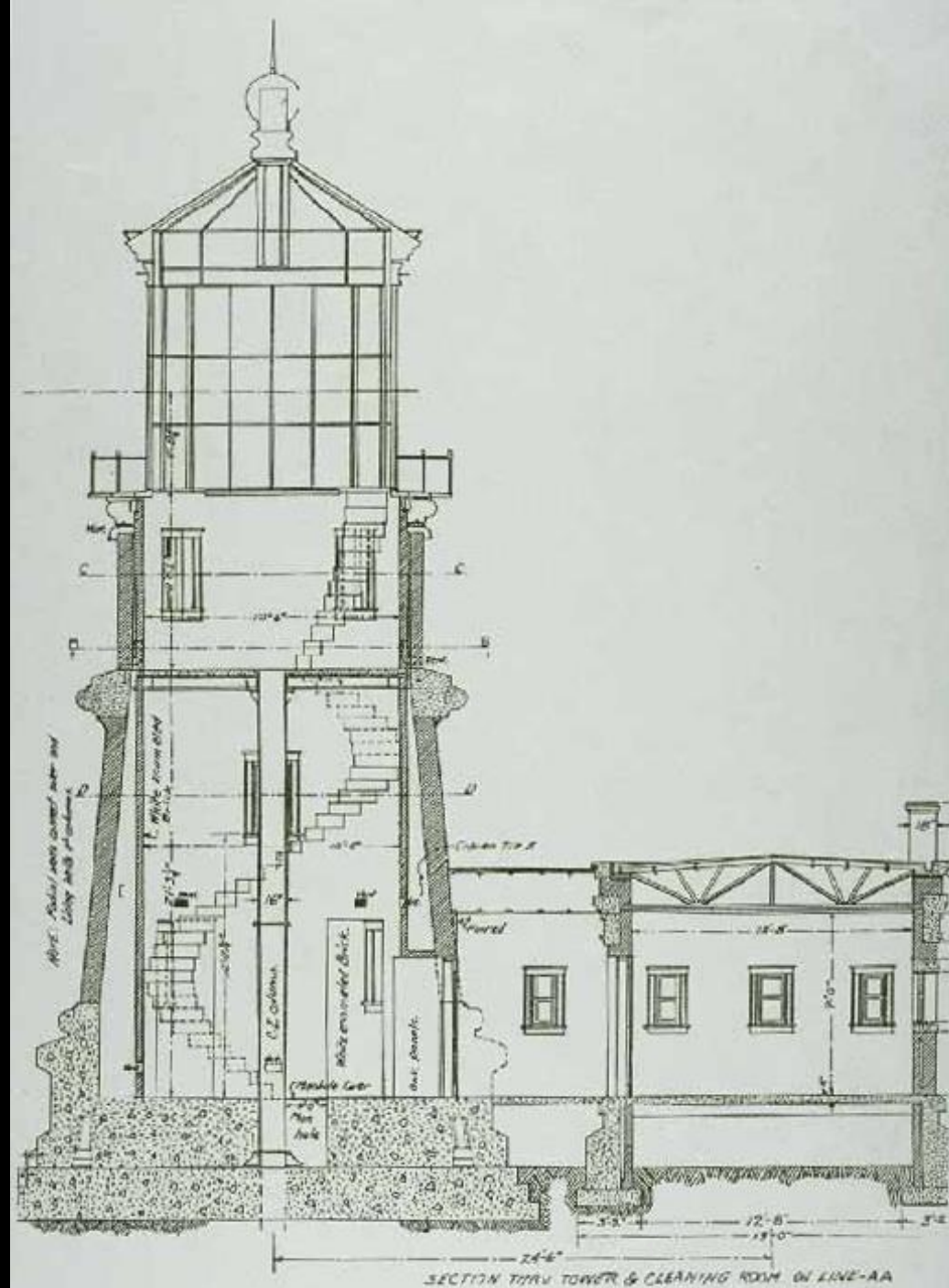


Software Architecture & Design





UI Architecture

The Presentation Layer

Contents

- Introduction
- History & Context: The Emergence of UI Infrastructure
- Guest Speaker: UI Architect
- Discussion & Q&A

History & Context

The emergence of UI Infrastructure

WIMP & WYSIWG

- Late '80s/Early '90s ushered in richer Windowing systems
 - X/Windows, Gnome
 - Microsoft Windows
 - Apple MAC UI
- Led to greater complexity in UI work

Web & Web Frameworks

- Early-Mid 2000's - Browser based UI 'become the trend'
 - Easier installation* and distribution
 - Greater Control of Upgrade & Migration
- One step forward...
 - More immature technology than predecessor
 - Lack of development & support tools
 - e.g. logging, debugging
- Rapid succession of the 'latest Web Framework'
 - e.g. JavaWeb, Adobe Flex, JavaScript & AngularJS, TypeScript
- Enterprises have had to 'redevelop' the UI
- Application Developers need to concentrate on Functional Requirements
- UI Infrastructure handles non-functional requirements
 - Insulate Application Developers

User Experience - UX

- UX emerges as a new discipline
 - Good UX can be 'the' determining factor
 - The UI is the shop-front window
 - UI 'trend' is a factor
 - Application Look & Feel will date
- Web introduced a different paradigm
 - Single-Click
 - Breadcrumbs
 - URL-like links
- Traditional UI (Fat Client) incorporates these

Client/Server

- Client/Server does not refer to concrete hardware
- Describes roles, not location

Traditional 3-Tier Client/Server



Client

(Presentation Logic)



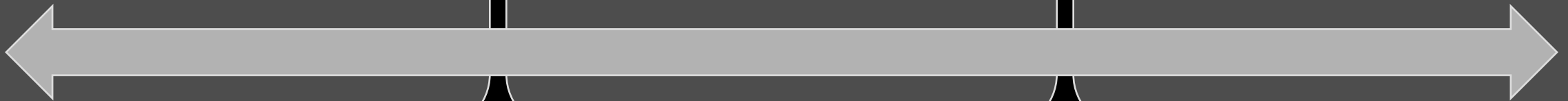
Server

(Business Processing)



Server

(Data Processing)



Client/Server Roles



Client

(Presentation Logic)



Server

(Business Processing)



Thin Client



UI
Rendering



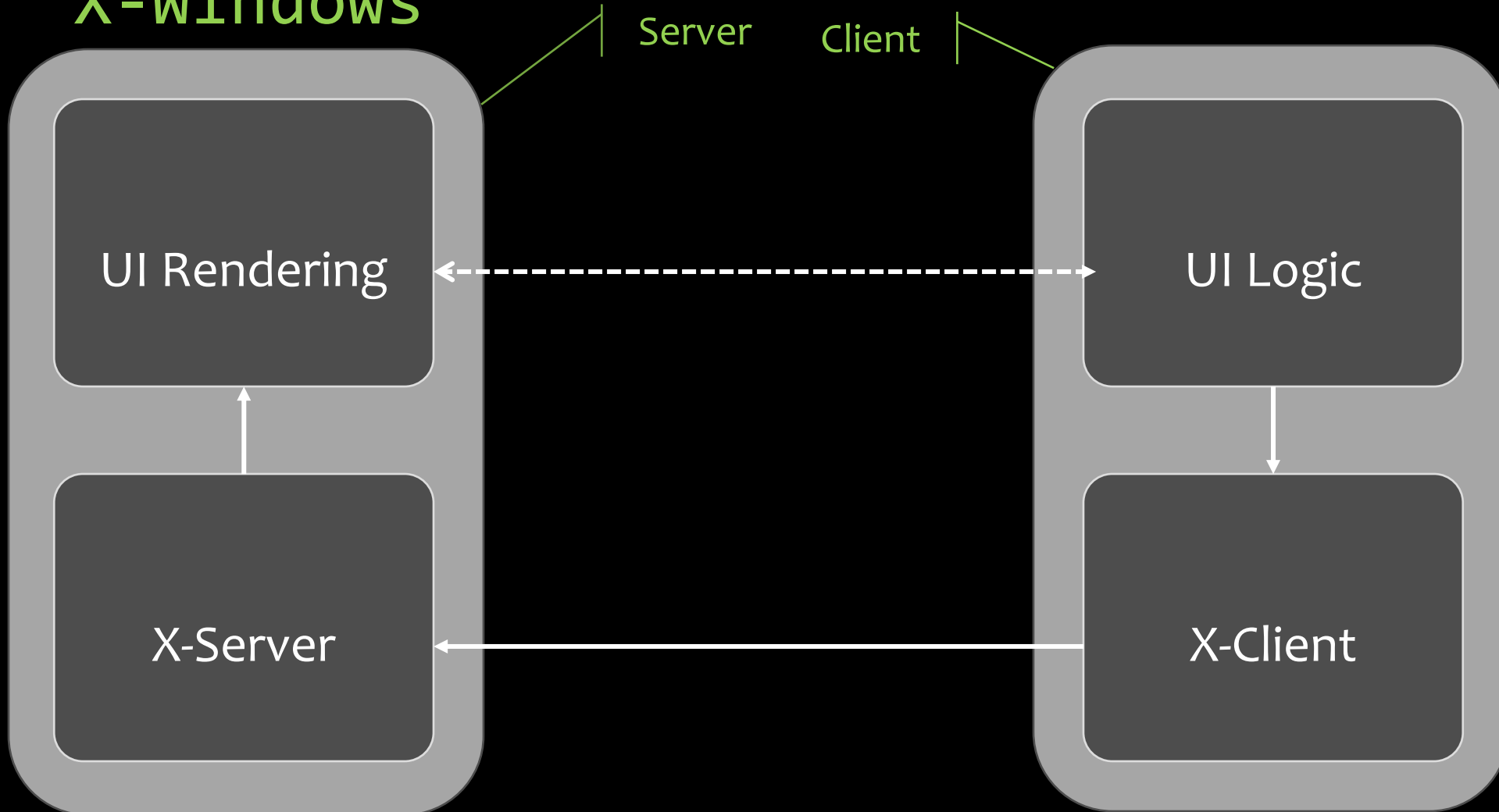
(Presentation Logic)



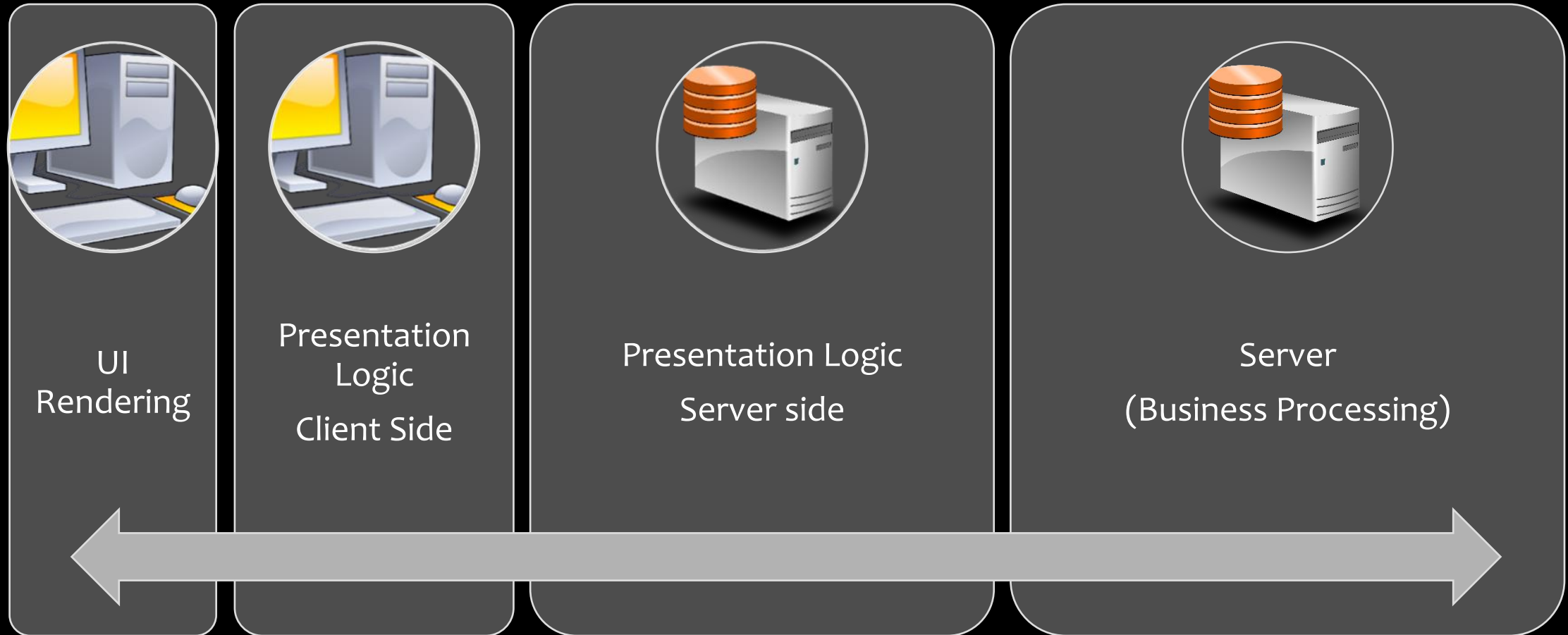
Server
(Business Processing)

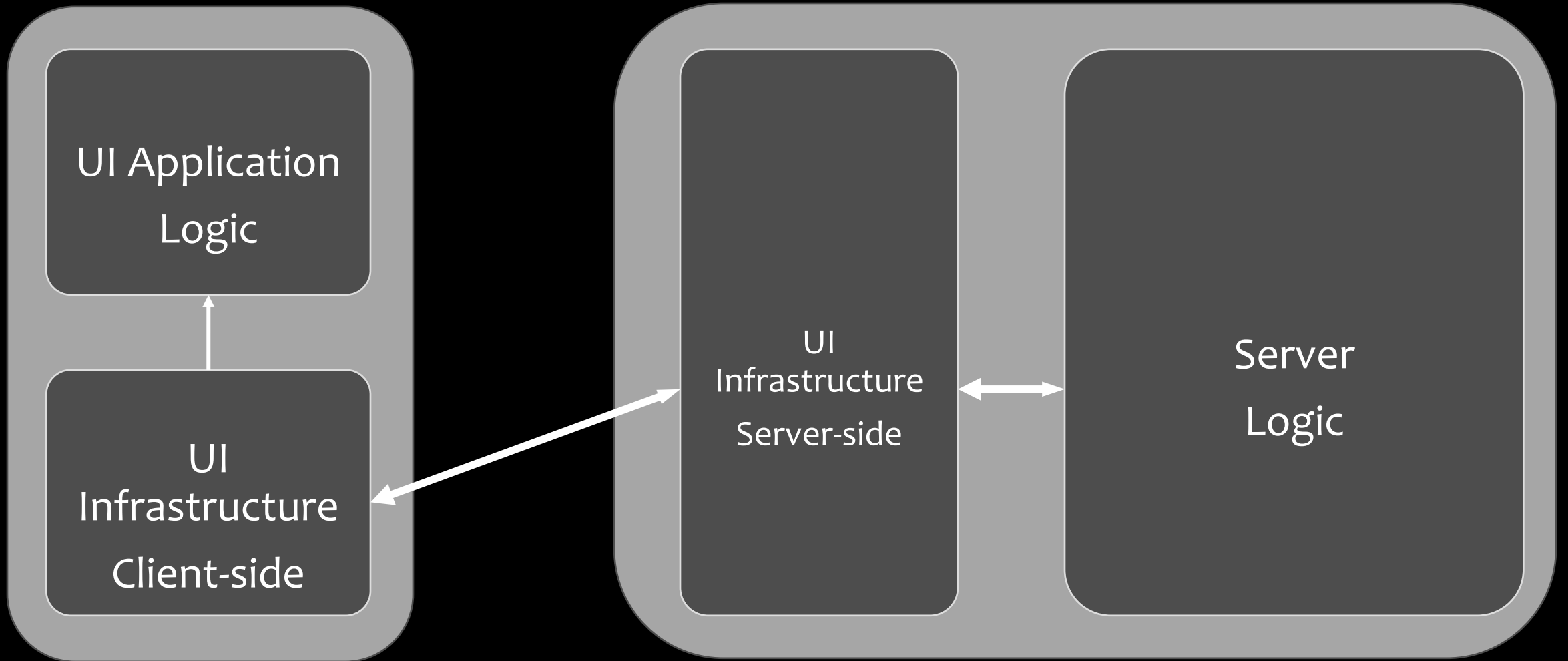


X-Windows



Full/Fat Client





Division of Labor

UI Agnostic

UI Application Logic

- Landing Page/Home Page
- List Views
- Detailed Views
- Wizards

UI Infrastructure (Client)

- Command Dispatching & Result Handling
- Session Management
- Authentication
 - Relaying Credentials
- Remote Event Forwarding

UI Infrastructure (Server)

- Command Fulfillment
- Session Management
- Remote Messaging Service
- Query Service
- Paging Service

Functional
Requirements

Non-Functional Requirements

Q&A

Discussion Time

Thank You

Recommended Reading

- Patterns of Enterprise Application Architecture *by Martin Fowler*

