Applying Patterns & Frameworks to Concurrent & Networked Software

Douglas C. Schmidt
d.schmidt@vanderbilt.edu
www.dre.vanderbilt.edu/~schmidt
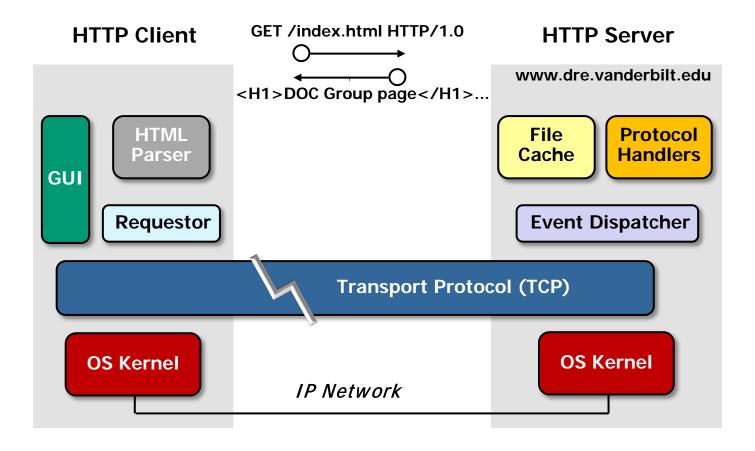
Professor of Computer Science

**Institute for Software
Integrated Systems**

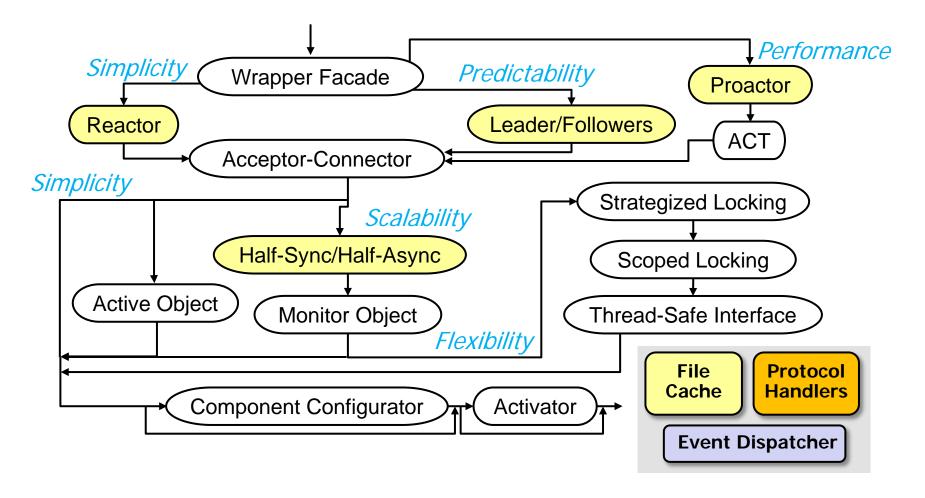**Vanderbilt University
Nashville, Tennessee, USA**
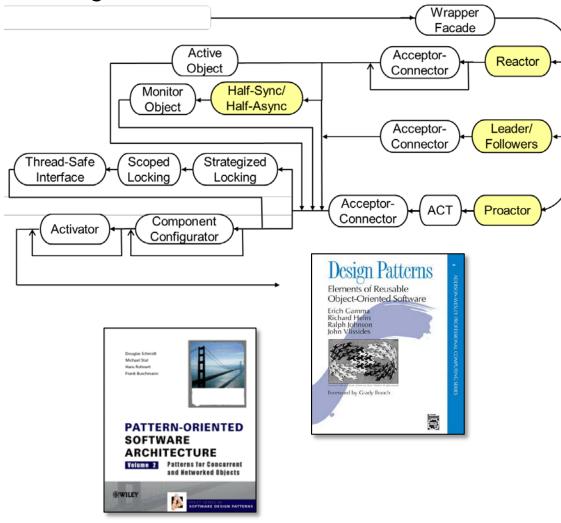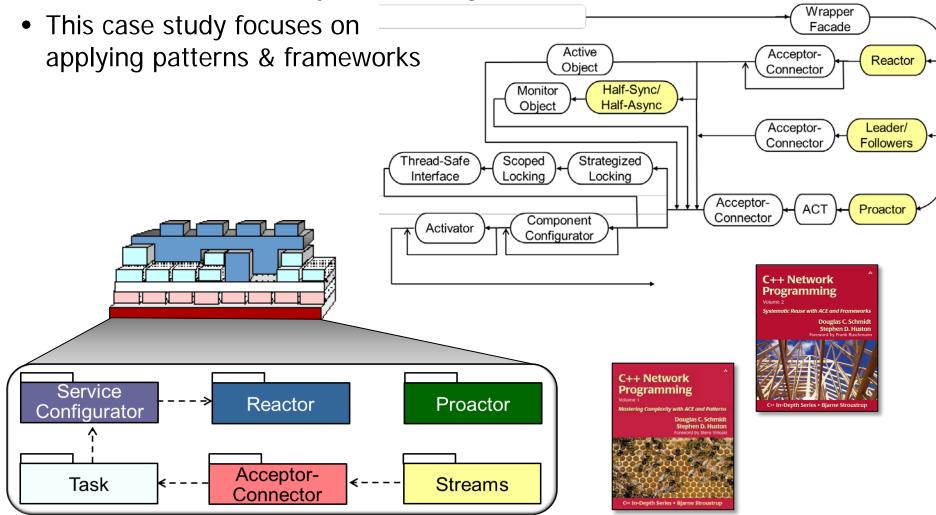
Topics Covered in this Section of Modules

- Summarize the case study used throughout this module

HTTP Client

GET /index.html HTTP/1.0

<H1>DOC Group page</H1>...

HTTP Server

www.dre.vanderbilt.edu

GUI

HTML Parser

Requestor

File Cache

Protocol Handlers

Event Dispatcher

Transport Protocol (TCP)

OS Kernel

OS Kernel

IP Network

Topics Covered in this Section of Modules

Topics Covered in this Section of Modules
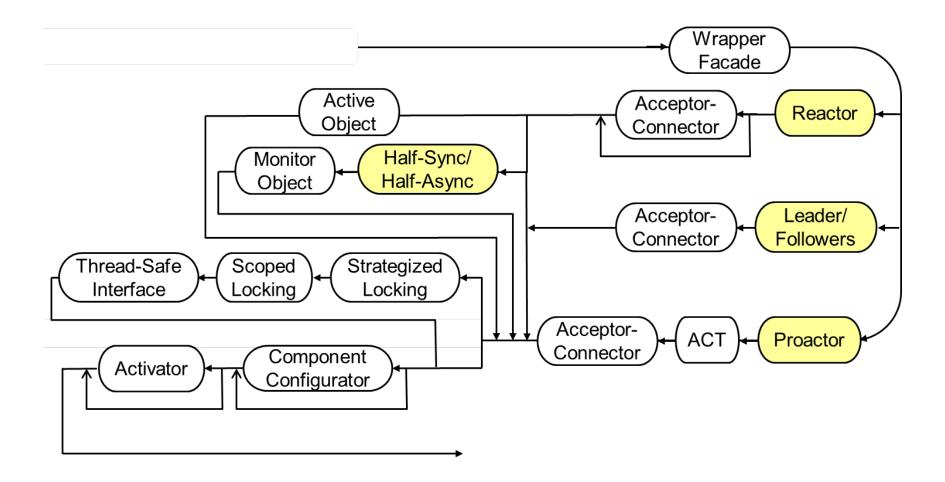
Topics Covered in this Section of Modules

- Summarize the case study used throughout this module

  - This case study focuses on applying patterns

Topics Covered in this Section of Modules

- Summarize the case study used throughout this module
  - This case study focuses on applying patterns & frameworks

Topics Covered in this Section of Modules

- Summarize the case study used throughout this module
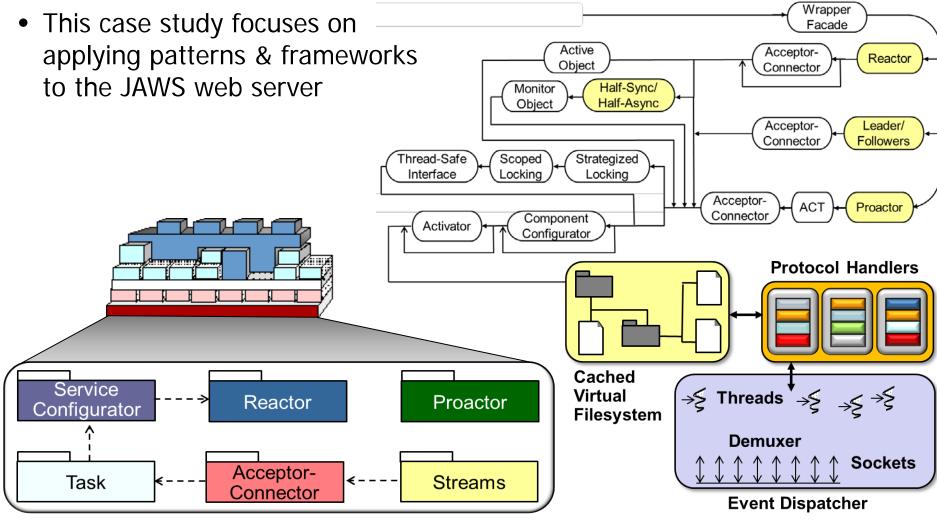 - This case study focuses on applying patterns & frameworks to the JAWS web server
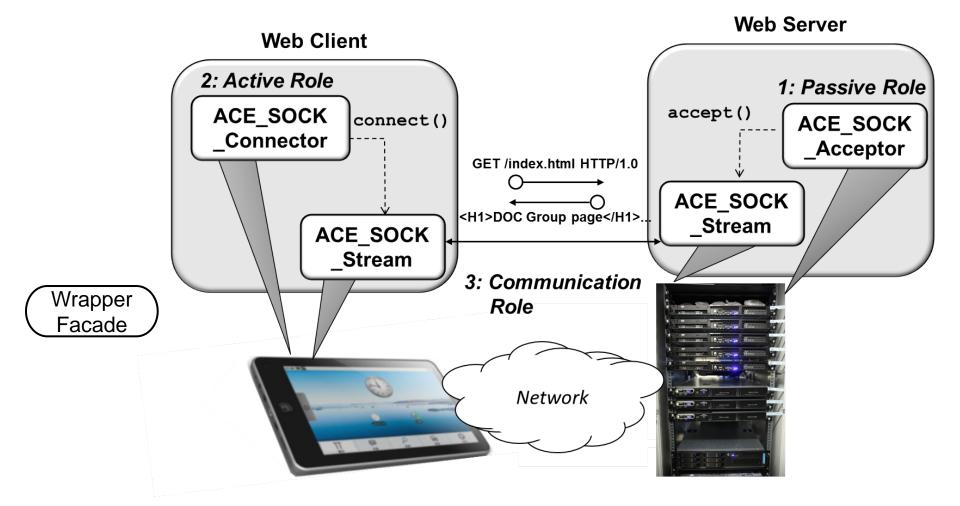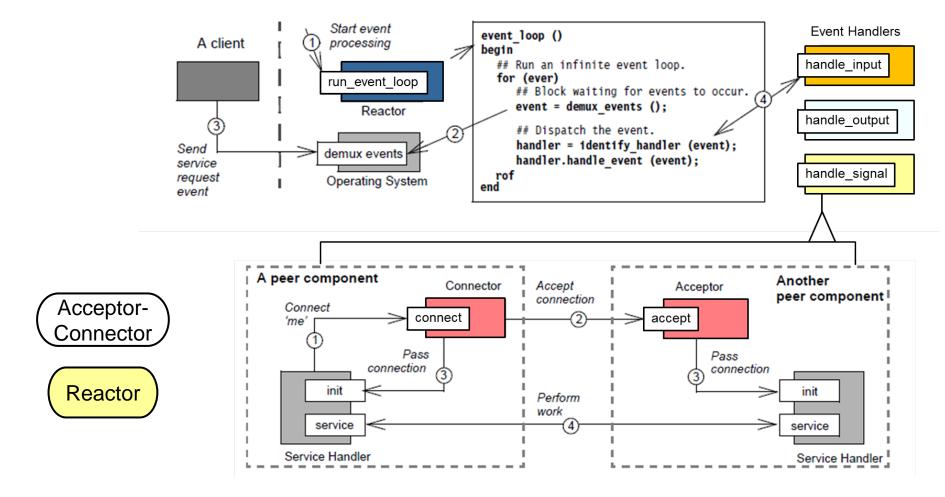
JAWS is available in open-source ACE release at download.dre.vanderbilt.edu

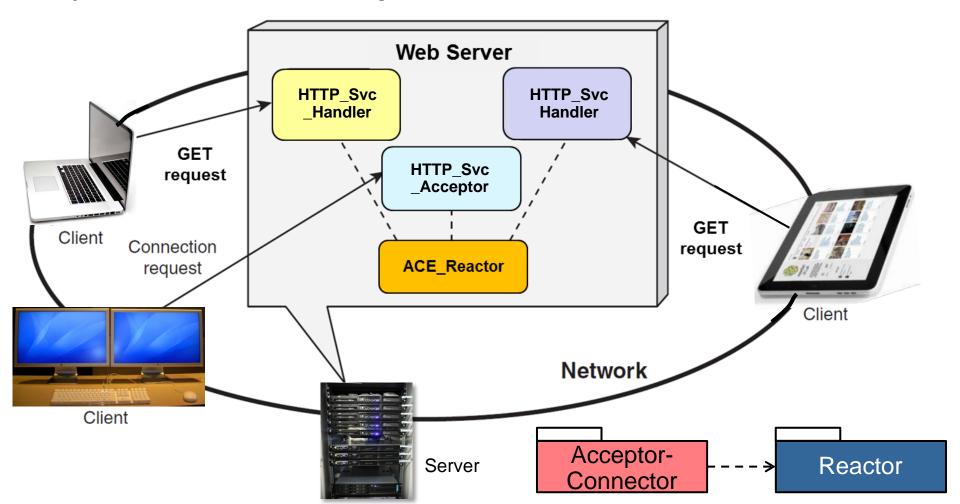# Topics Covered in this Section of Modules

- Summarize the case study used throughout the module
- Apply patterns & frameworks to JAWS in the following areas
  - Service access & communication

Topics Covered in this Section of Modules

- Summarize the case study used throughout the module

- Apply patterns & frameworks to JAWS in the following areas
  - Synchronous event handling, connections, & service initialization
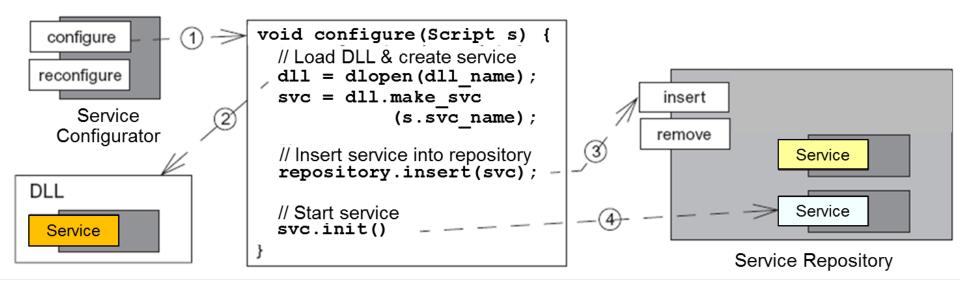
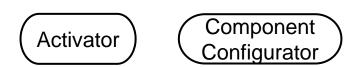# Topics Covered in this Section of Modules

- Summarize the case study used throughout the module
- Apply patterns & frameworks to JAWS in the following areas
  - Synchronous event handling, connections, & service initialization

Topics Covered in this Section of Modules

- Summarize the case study used throughout the module
- Apply patterns & frameworks to JAWS in the following areas
  - Service configuration & activation

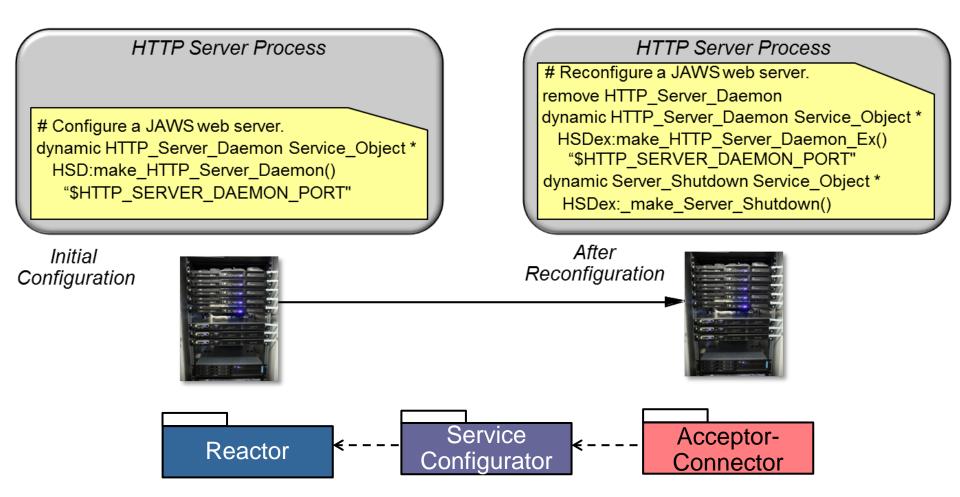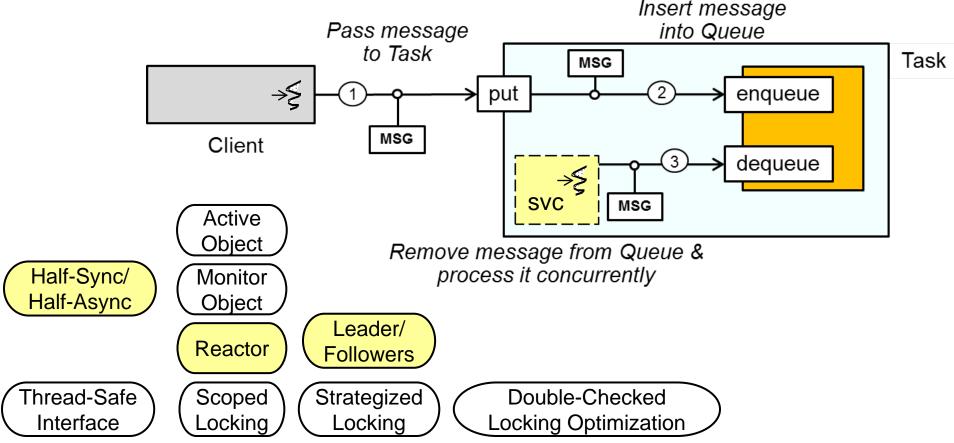# Topics Covered in this Section of Modules

- Summarize the case study used throughout the module
- Apply patterns & frameworks to JAWS in the following areas
  - Service configuration & activation

HTTP Server Process

```
# Configure a JAWS web server.
dynamic HTTP_Server_Daemon Service_Object *
  HSD:make_HTTP_Server_Daemon()
    "$HTTP_SERVER_DAEMON_PORT"
```

HTTP Server Process

```
# Reconfigure a JAWS web server.
remove HTTP_Server_Daemon
dynamic HTTP_Server_Daemon Service_Object *
  HSDex:make_HTTP_Server_Daemon_Ex()
    "$HTTP_SERVER_DAEMON_PORT"
dynamic Server_Shutdown Service_Object *
  HSDex:_make_Server_Shutdown()
```

Initial Configuration

After Reconfiguration

Reactor ← Service Configurator ← Acceptor-Connector
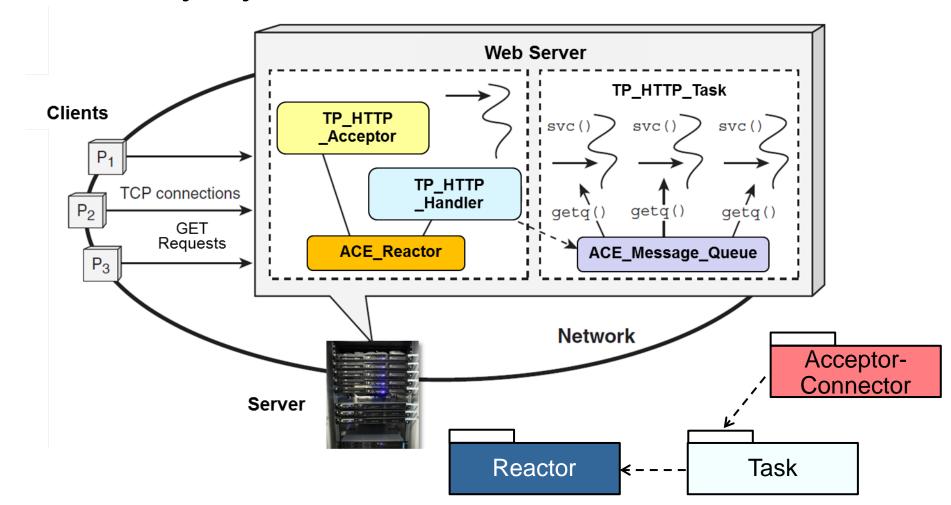
Topics Covered in this Section of Modules

- Summarize the case study used throughout the module
- Apply patterns & frameworks to JAWS in the following areas
  - Concurrency & synchronization

Topics Covered in this Section of Modules
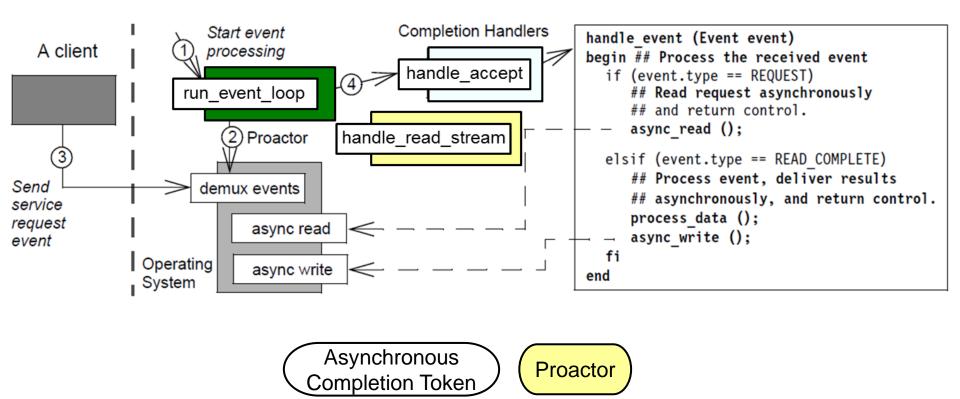
- Summarize the case study used throughout the module

- Apply patterns & frameworks to JAWS in the following areas
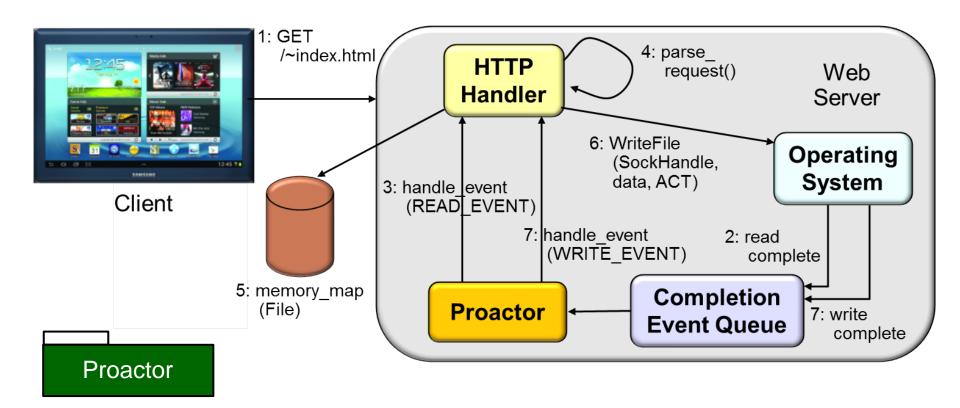
  - Concurrency & synchronization

Topics Covered in this Section of Modules

- Summarize the case study used throughout the module
- Apply patterns & frameworks to JAWS in the following areas
  - Asynchronous event handling

Topics Covered in this Section of Modules

- Summarize the case study used throughout the module
- Apply patterns & frameworks to JAWS in the following areas
  - Asynchronous event handling

Topics Covered in this Section of Modules

- Summarize the case study used throughout the module

- Apply patterns & frameworks to JAWS in the following areas

  - Service access & communication

  - Synchronous event handling, connections, & service initialization

  - Service configuration & activation

  - Concurrency & synchronization

  - Asynchronous event handling

- Evaluate benefits & limitations of applying patterns & frameworks to concurrent & networked software

  - Complexity vs. productivity

  - Extensibility vs. performance

  - Buy vs. build

  - Debugging & testing

We'll cover many code examples written in C++ & Java