

Component Based Approach

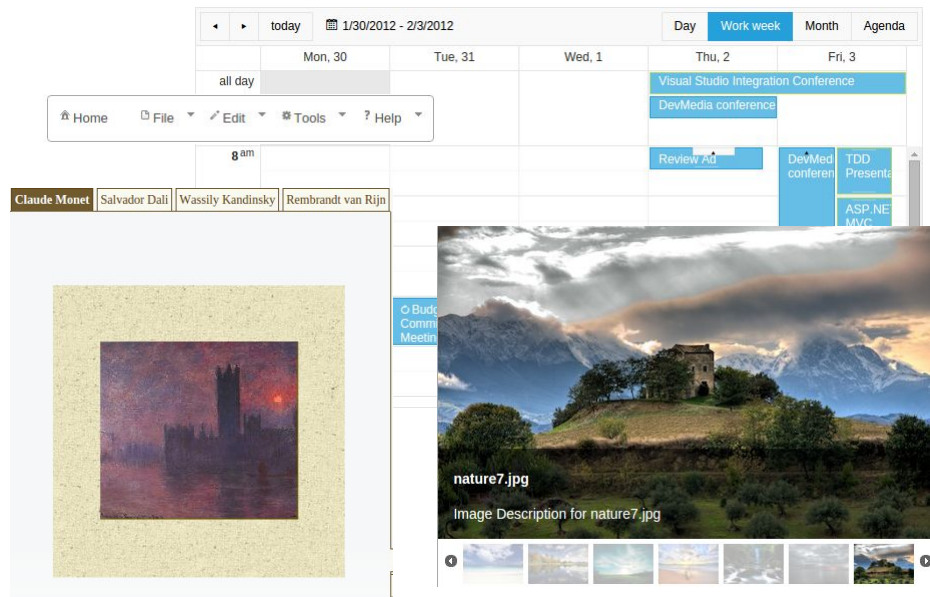
Intro

JSF Slides #1

Content

- Basic Idea
- Architecture
- Java Server Faces (JSF)
- NetBeans and JSF

Basic Ideas



Basic ideas is to emulate desktop programming (Swing)

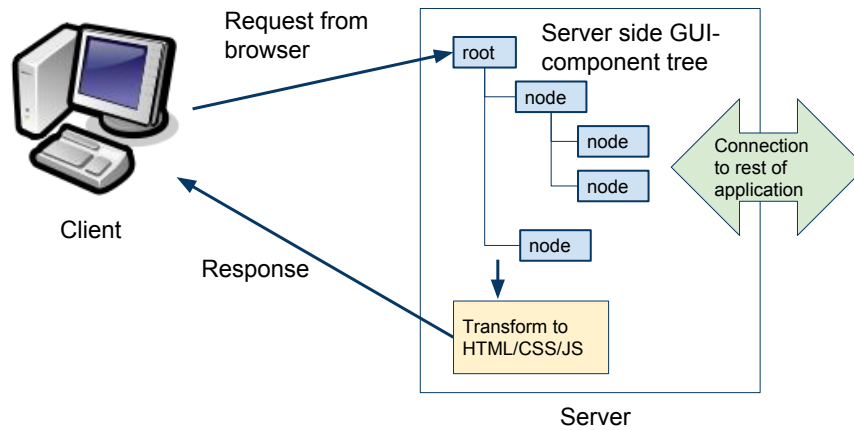
- Abstract away the network
- Skip all CSS, JS
 - Use components hiding it all

Everything in slide is a component (from component suites by [ICEFaces](#), [PrimeFaces](#), [ASP.NET](#), [OpenFaces](#))

Demo:

- [Mock OSX](#) (PrimeFaces)
- [Scheduling](#) (ASP.NET)

Basic Architecture



Let GUI live as a component tree (tree of objects) on server side

- One for each unique client, so tree must have unique id
- Incoming requests routed to clients tree for updates of values,
 - Calls to listeners, application processing
- Translate to HTML (CSS, JS) and send to client for rendering

Java Server Faces



[Java Server Faces \(JSF\)](#)

- Swing for the Web ...

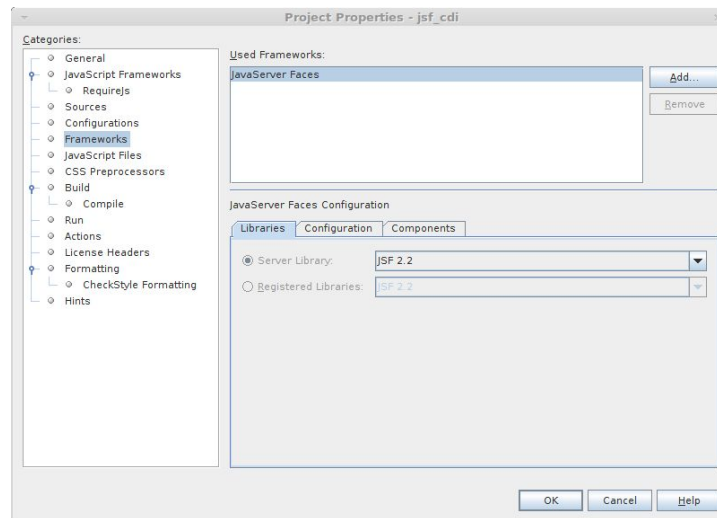
Huge "framework" for web apps (server side) component based approach

- Very many pieces ...
 - Hard to find good documentation
- Often many ways to do things, .. confusing
- Evolving, ... thing previously mandatory, relaxed ... also confusing ...
- Default a "full page load" technique
 - AJAX possible, more to come ...

Reference implementation Mojarra

[Examples](#) on Github

NetBeans



Possible to specify JSF (and component suites) in Netbeans