AJAX & GWT

Trey Roby

The Web is Changing

- Things we never imagined
- Central to people's lives
- Great Opportunity

A Very Brief History of Computing

How did Microsoft make it big?

- Entered the Main Frame world
- Everything batch
- Introduced interactive programming to business

Batch processing front end

1. *Fill out form*
2. *Server processes*
3. *Get results*

AJAX Revolution

Changed the way we think about web-development

- Google Maps - Feb 8, 2005
- Started it all - 4 1/2 years ago
- Millions of people wanted to use it

Everywhere

- Google
  - *Calendar*
  - *Word processor*
  - *Gmail - fastest growing*
- Yahoo has revamped website
  - *Finance, Mail*
- News websites
- YouTube, NetFlix, Hulu, Facebook
- Slashdot.org
- Specialty sites
  - *doodle.com, rememberthemilk.com*
- Adobe Photoshop coming to web
- Apple announced new web-development environment
  - *http://280slides.com/*
- Microsoft will rewrite website

Technologies Behind The Revolution

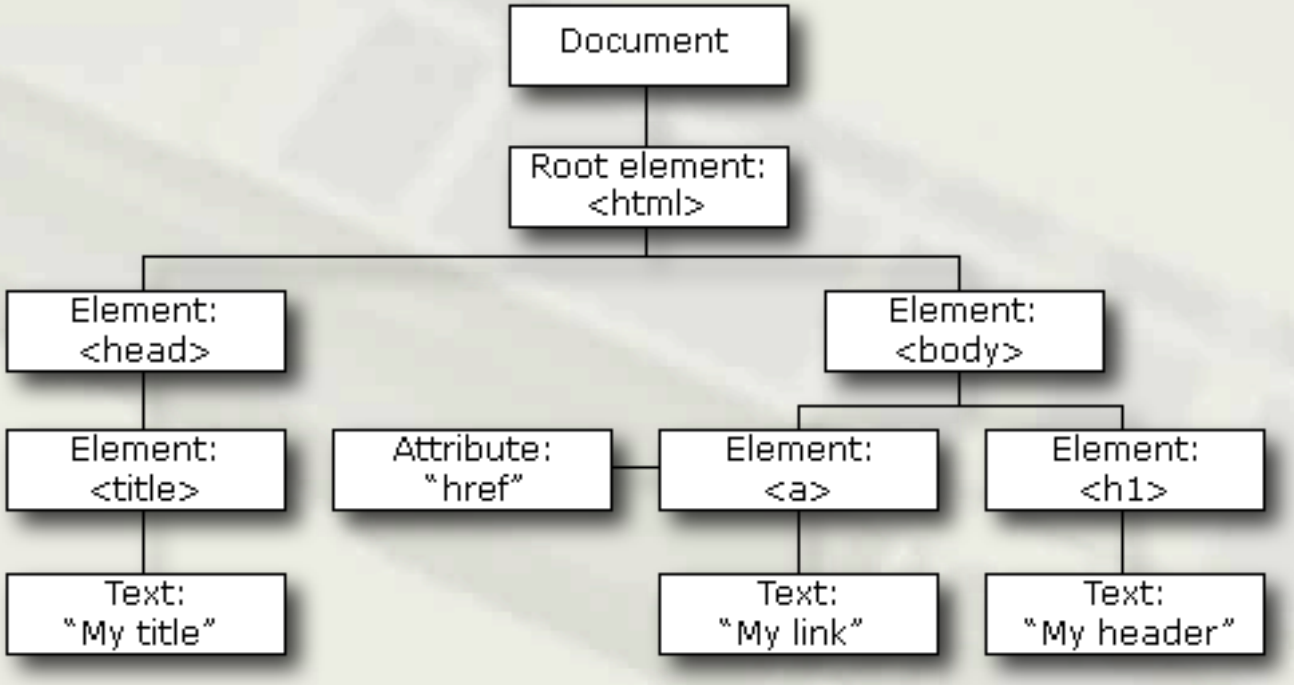
AJAX

- Asynchronous JavaScript and XML
- Nothing new
- Technologies
  - *JavaScript*
  - *Dynamic HTML*
  - *CSS*
  - *DOM*
  - *XML*
  - *JASON*
- Emphasis on JavaScript speed

Dynamic HTML

- Web Page can change without reload
- HTML document
  - *viewed as a tree*
  - *Each tag is a node*
- Scripts can change any part of tree
- Any change immediately shows up on page
- DOM - Document Object Model

DOM


Document
Root element:
<html>
Element:
<head>
Element:
<body>
Element:
<title>
Attribute:
"href"
Element:
<a>
Element:
<h1>
Text:
"My title"
Text:
"My link"
Text:
"My header"

JavaScript

- Powerful scripting language
- Very flexible
- Runs in browser
- C like
- Easy to get started
- Event driven
  - *onclick*
  - *onfocus*
  - *onresize*
  - *onkeydown*
  - *onmousemove*
- DOM
  - *access DOM*
  - *change DOM*

Asynchronous Calls

- JavaScript can make Asynchronous HTTP calls

- Synchronous Call?
  - *call made*
  - *results return immediately*
  - *typical program function call*

- Asynchronous Call to Server?
  - *function call is made*
  - *results return later*
  - *a method (function) is called upon completion*
  - *results can be used to change DOM*

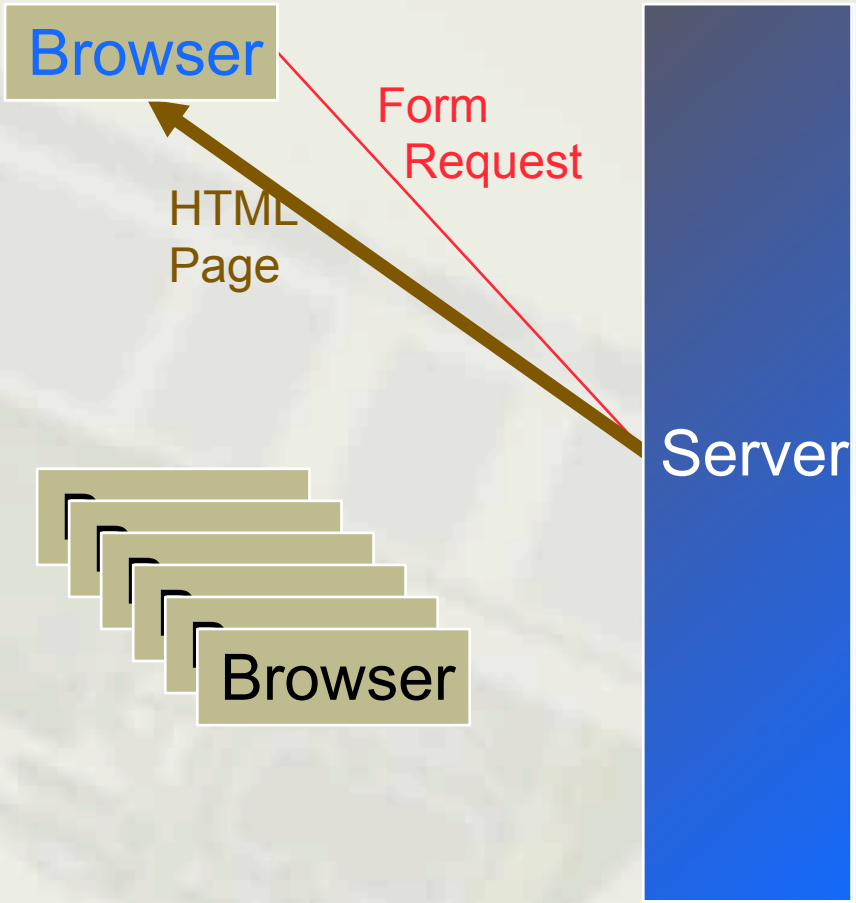
AJAX Web vs. Classic Web

- Classic Web
  - *One server call = Whole Page Load*
  - *Server call = UI Generation*
  - *Server call = Large Overhead*
  - *Server call required for useful results*
- AJAX Web
  - *One Server call = Page update*
  - *One Server call = data only*
  - *One Server call = small overhead*
  - *Page can generate results without server call*

Ajax allows for multiple, lighter, server calls and more dynamic web pages

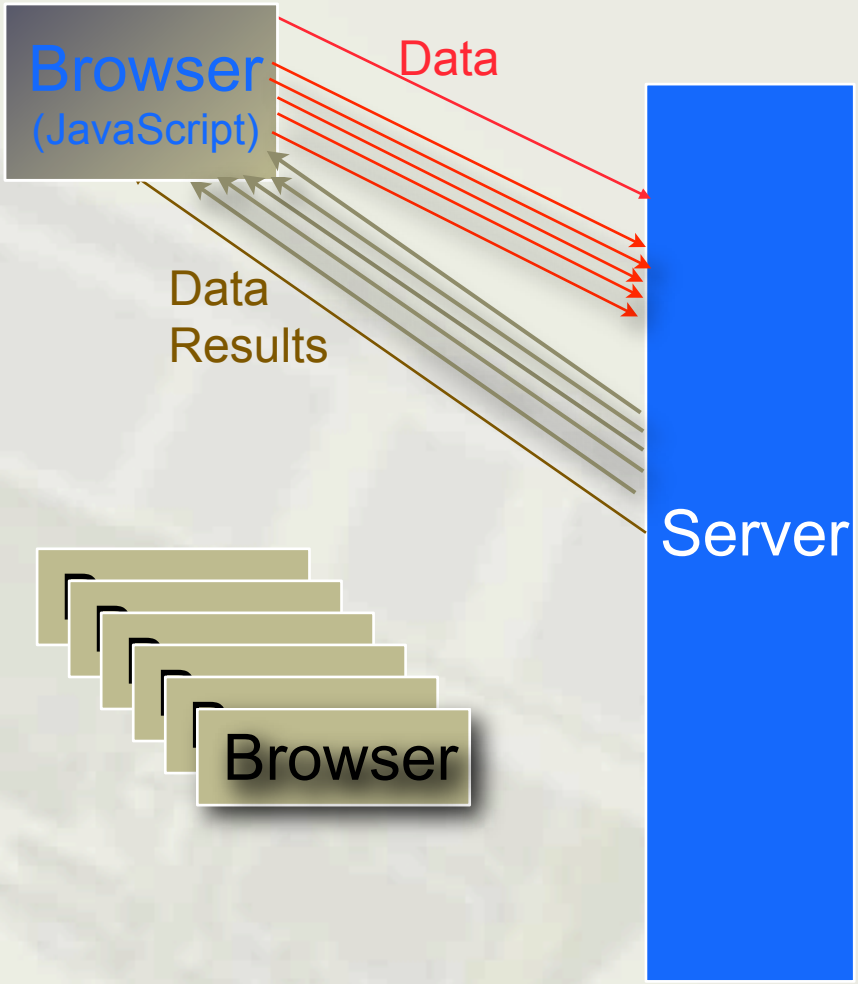
Classic Web Application

1. Send Form Request
2. Server
   1) retrieves data
   2) builds UI
   3) generates HTML
3. Server returns HTML Page
4. Browser shows HTML page



JavaScript/Ajax Based Web Application

1. Compiled JavaScript client runs on browser
2. Sends data request
3. Server returns data
4. Client manages & presents data
5. Happens again & again on same page



Browser becomes Application Environment

- Application code runs on browser
- Return of interactive programming
- Heavy client

Heavy Client

- Better use browser computing power
- Takes advantage full browser potential
- Faster
- Interactive
- Better user experience

Challenges in using AJAX

AJAX Challenges

- Browsers work differently
- Browser API can be slightly different
- Requires constant checking of browser type
- Many, many browser subtleties

JavaScript Weaknesses

- Not strongly typed
  - *Scripting language*
- Weak debugger support
- No compilation
  - *syntax errors caught at runtime*
  - *Can't optimize*
- Does not work well with large applications

Great Opportunity / Lots of problems

- Opportunity
  - *Stunning web applications*
  - *Great benefit to user*
  - *Easy to use*

- Problems
  - *Hard to debug*
  - *Lots of testing on Browsers*
  - *Have to be AJAX guru*
  - *Easy to write bad Java Script*

What is the Solution?

Google Web Toolkit

GWT

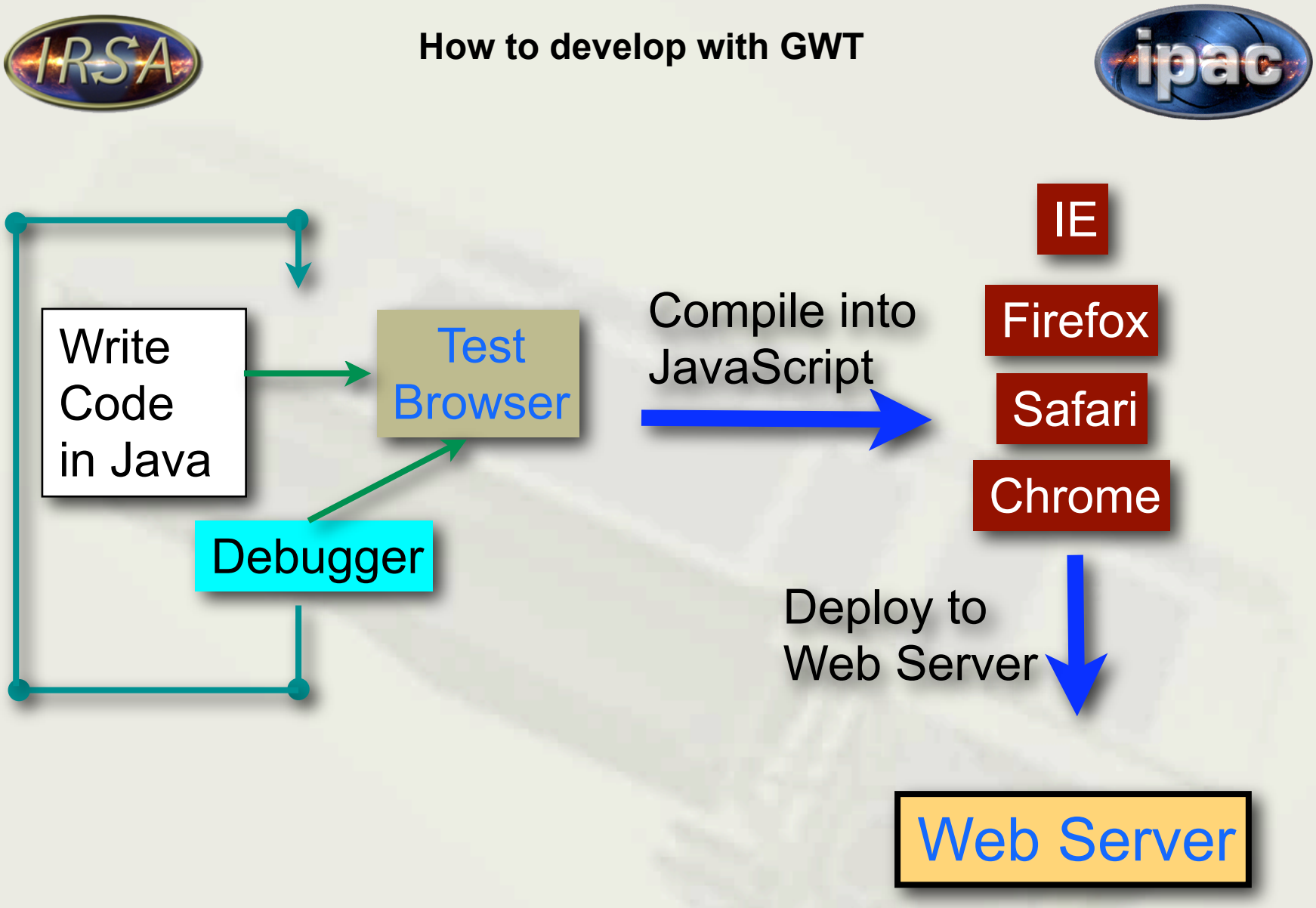
- Google Web Toolkit
- One approach to AJAX
- Well Supported
- Java instead of JavaScript
- Attempts to fix the AJAX challenges
- Free

Benefits of GWT

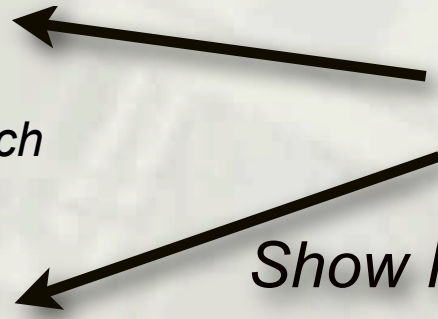
- Handles cross browser issues
- Benefits of compilation
- More optimal
- Java better for large applications
- Great debugging support
- Java code reuse
- Natively call JavaScript
- No plugins

How it works

- Compiles Java into JavaScript
- Creates one JavaScript file per browser type
- Only builds what you use
- Provides UI support
- Comes with a development browser for debugging
- Provides an RPC environment for server calls
- Allows for anything JavaScript can do


How to develop with GWT
Write Code in Java
Test Browser
Debugger
Compile into JavaScript
IE
Firefox
Safari
Chrome
Deploy to Web Server
Web Server

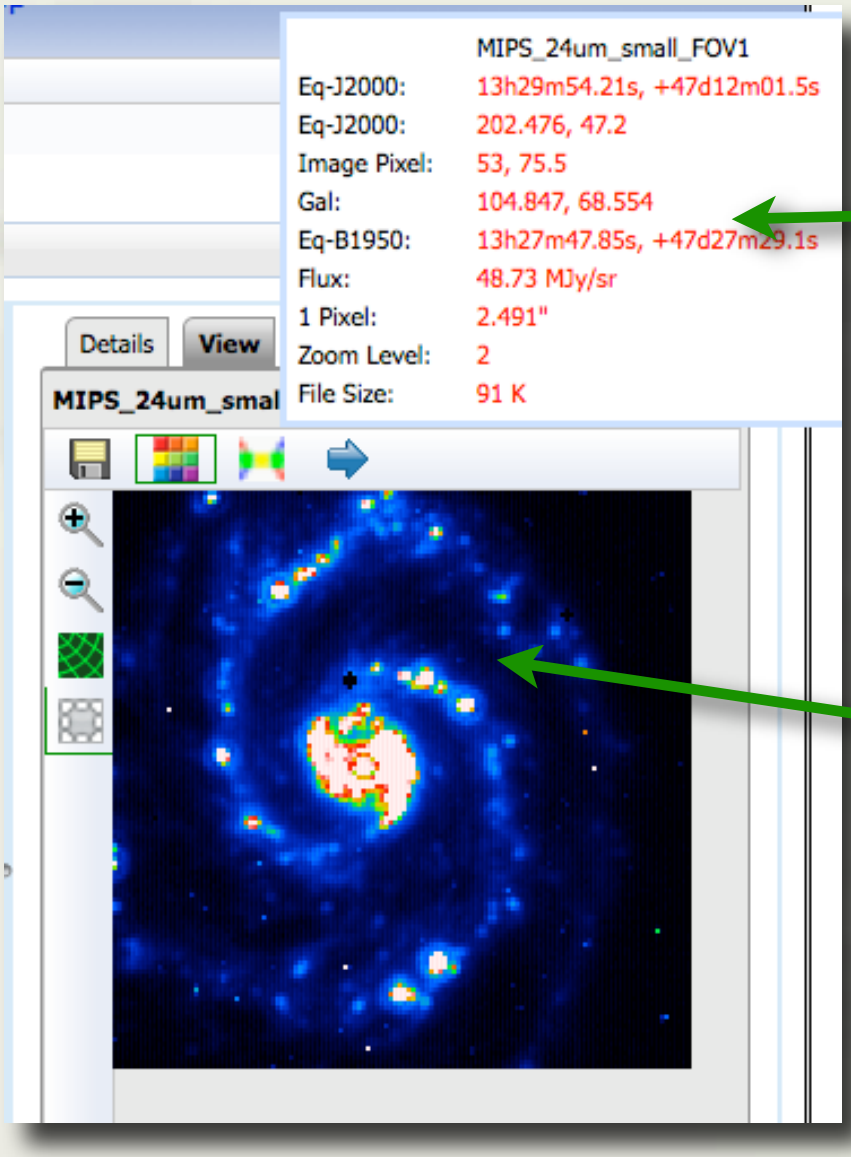
- Name resolution on the browser
- Input field validation
- Coordinate conversion
- Tabular displays without page reload
  - *Sort*
  - *Page*
  - *Update*
  - *Row clicking*
- True FITS Visualization
  - *Zooming*
  - *Changing color / Stretch*
  - *Magnifier / Thumbnail*
  - *Crop*
  - *Ra, Dec, Flux readout*



Example

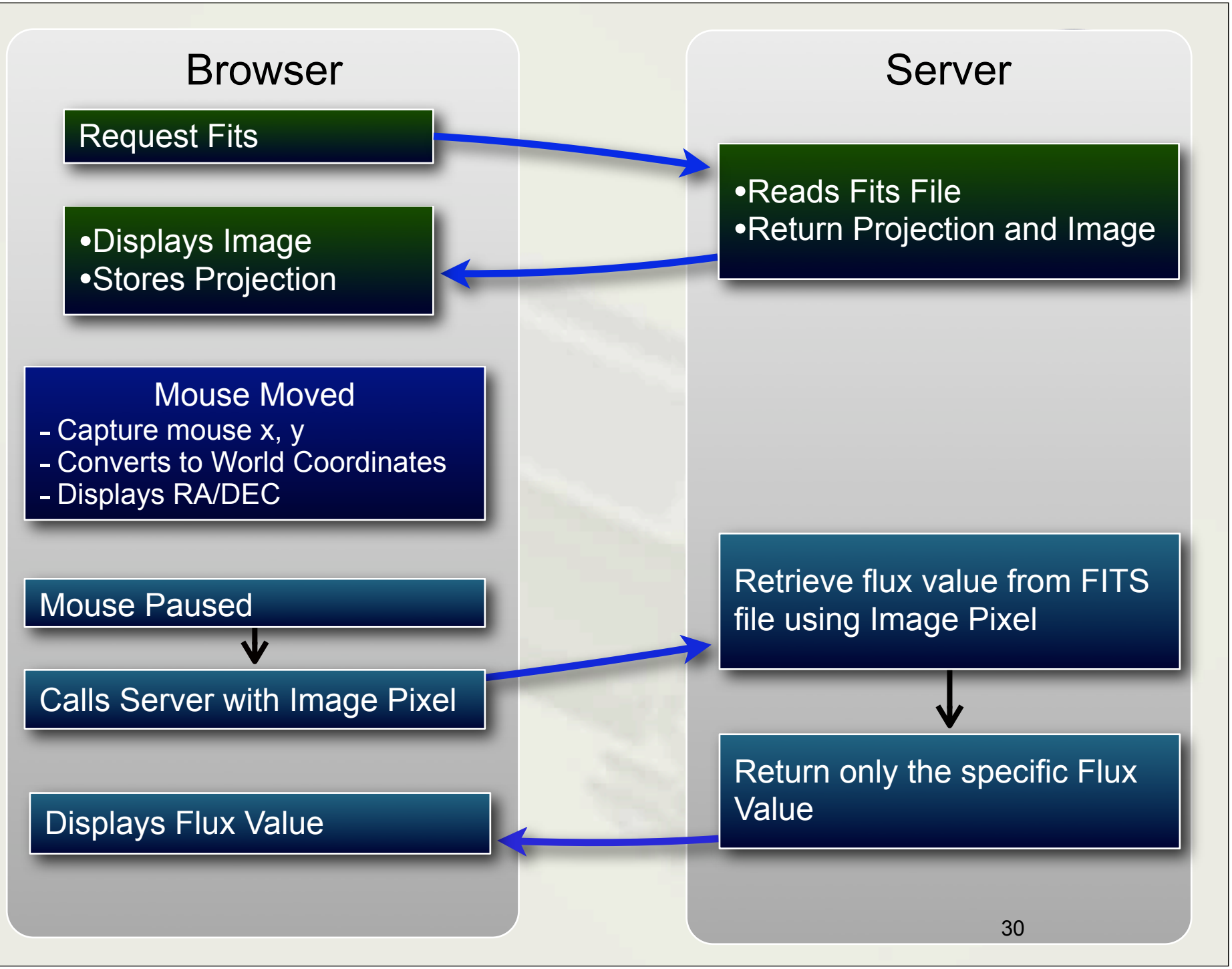
Show FITS data and mouse readout

Example: Web Fits Viewer



See Mouse Readout

Move Mouse Over Image


Browser
Server
Request Fits
•Reads Fits File
•Return Projection and Image
•Displays Image
•Stores Projection
Mouse Moved
- Capture mouse x, y
- Converts to World Coordinates
- Displays RA/DEC
Mouse Paused
Calls Server with Image Pixel
Retrieve flux value from FITS file using Image Pixel
Return only the specific Flux Value
Displays Flux Value


ipac