SOAized SHIELD

Team Pic Portal

Introduction

- Team Members:
  - Shaun Netherby (Leader)
  - Karan Shah
  - Ryan Bohner (Webmaster)
- Technical Advisor:
  - Dr. Wolf-Dieter Otte
- Sponsor:
  - Michael Inman
- Company:
  - Lockheed Martin

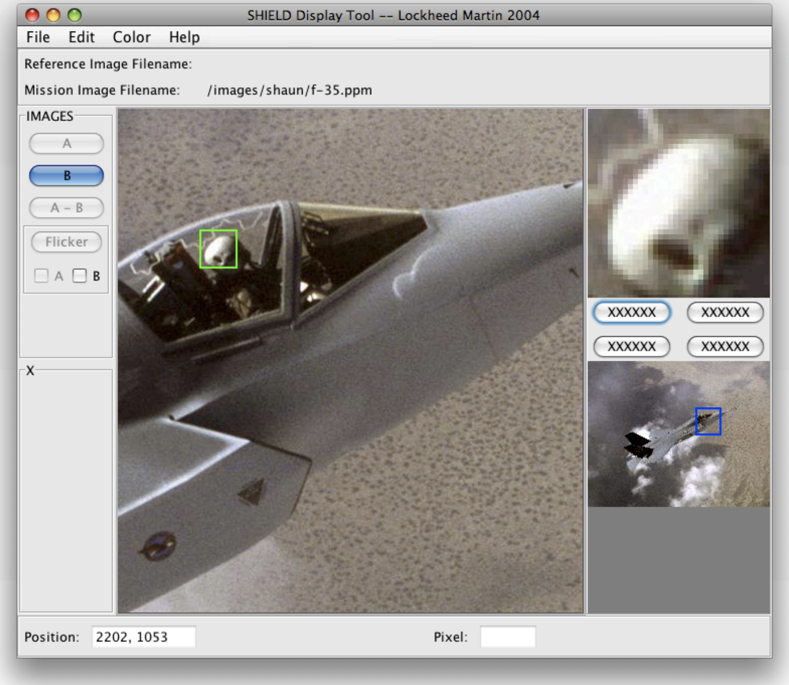
Overview

- Overview of original SHIELD
- Problems with original SHIELD
- Solution : SOAized SHIELD
- Architecture
- Results
- Conclusion

Current Tool - SHIELD

- Lockheed Martin uses SHIELD to display and manipulate SAR images
- Installed as a standalone application
- Requires the client to download the images to their local workstation

SHIELD



SHIELD Display Tool -- Lockheed Martin 2004

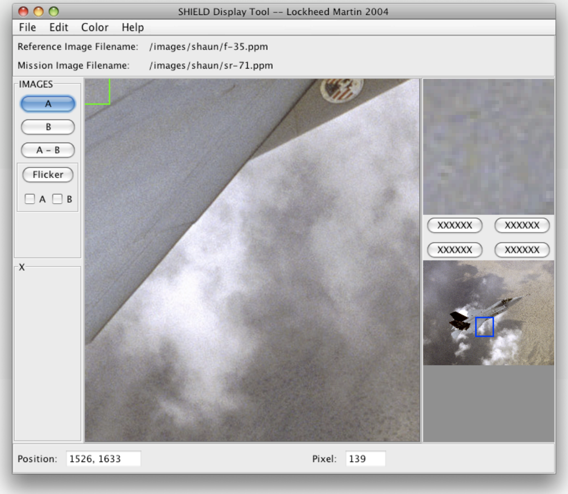
Reference Image Filename:

Mission Image Filename: /images/shaun/f-35.ppm

Position: 2202, 1053

Pixel:

SHIELD

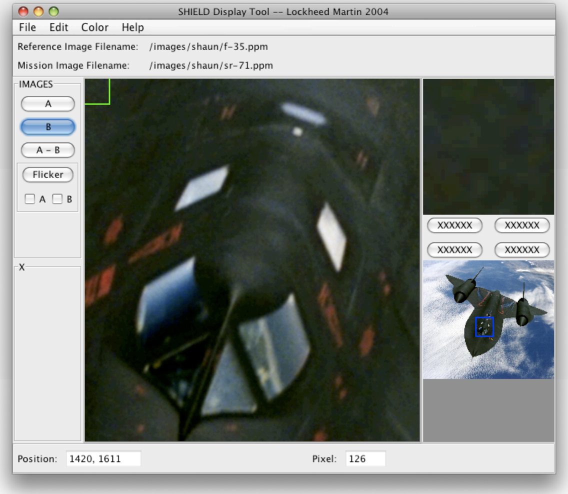


Reference Image Filename: /images/shaun/f-35.ppm

Mission Image Filename: /images/shaun/sr-71.ppm

Position: 1526, 1633

Pixel: 139

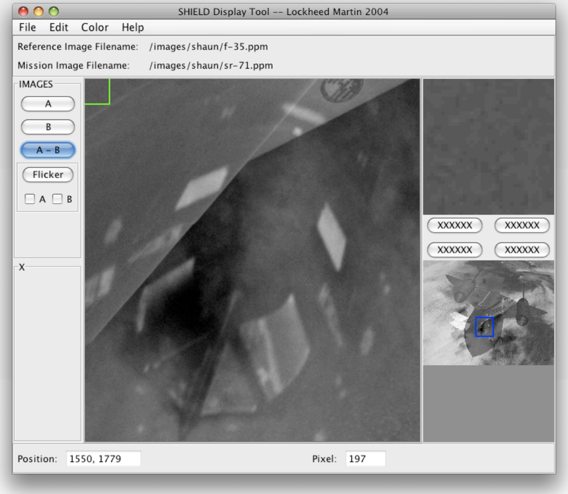


Reference Image Filename: /images/shaun/f-35.ppm

Mission Image Filename: /images/shaun/sr-71.ppm

Position: 1420, 1611

Pixel: 126



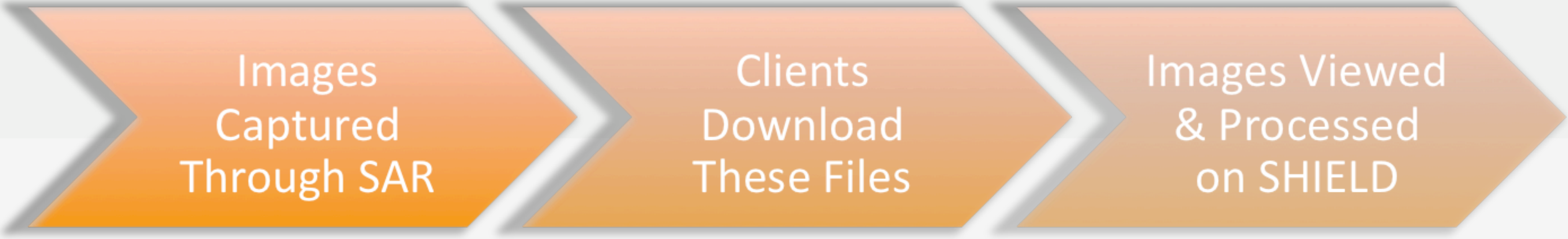
Reference Image Filename: /images/shaun/f-35.ppm

Mission Image Filename: /images/shaun/sr-71.ppm

Position: 1550, 1779

Pixel: 197

Current Tool - SHIELD



Images Captured Through SAR

Clients Download These Files

Images Viewed & Processed on SHIELD

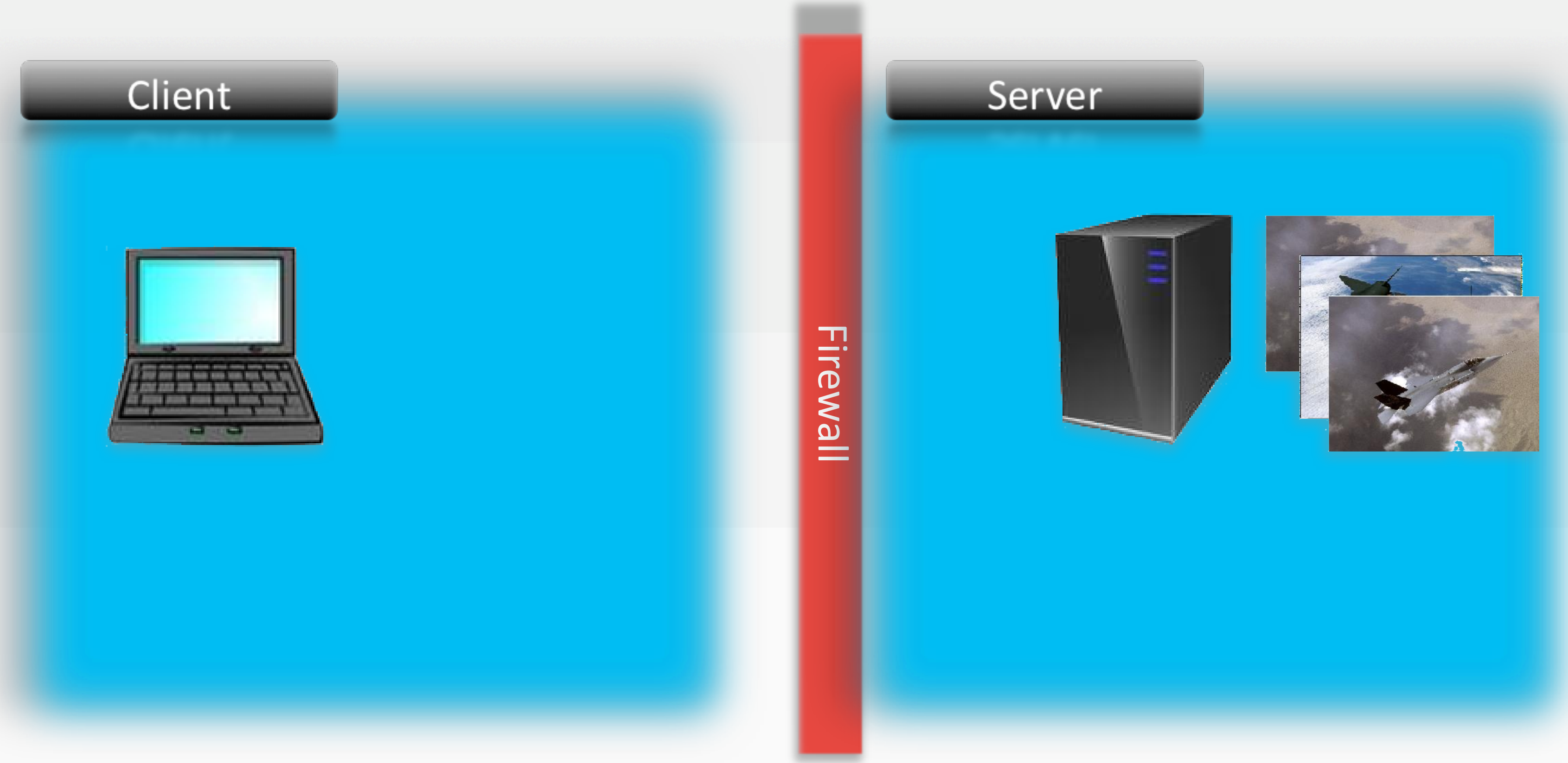
Problems with SHIELD

- SHIELD uses massive amounts of resources
  - Images were huge (upwards of 1 GB)
  - Images were too large to efficiently process on a standard computer
  - It could take more time to download the images than to create them
- These images are sensitive and LM had no control over them once distributed

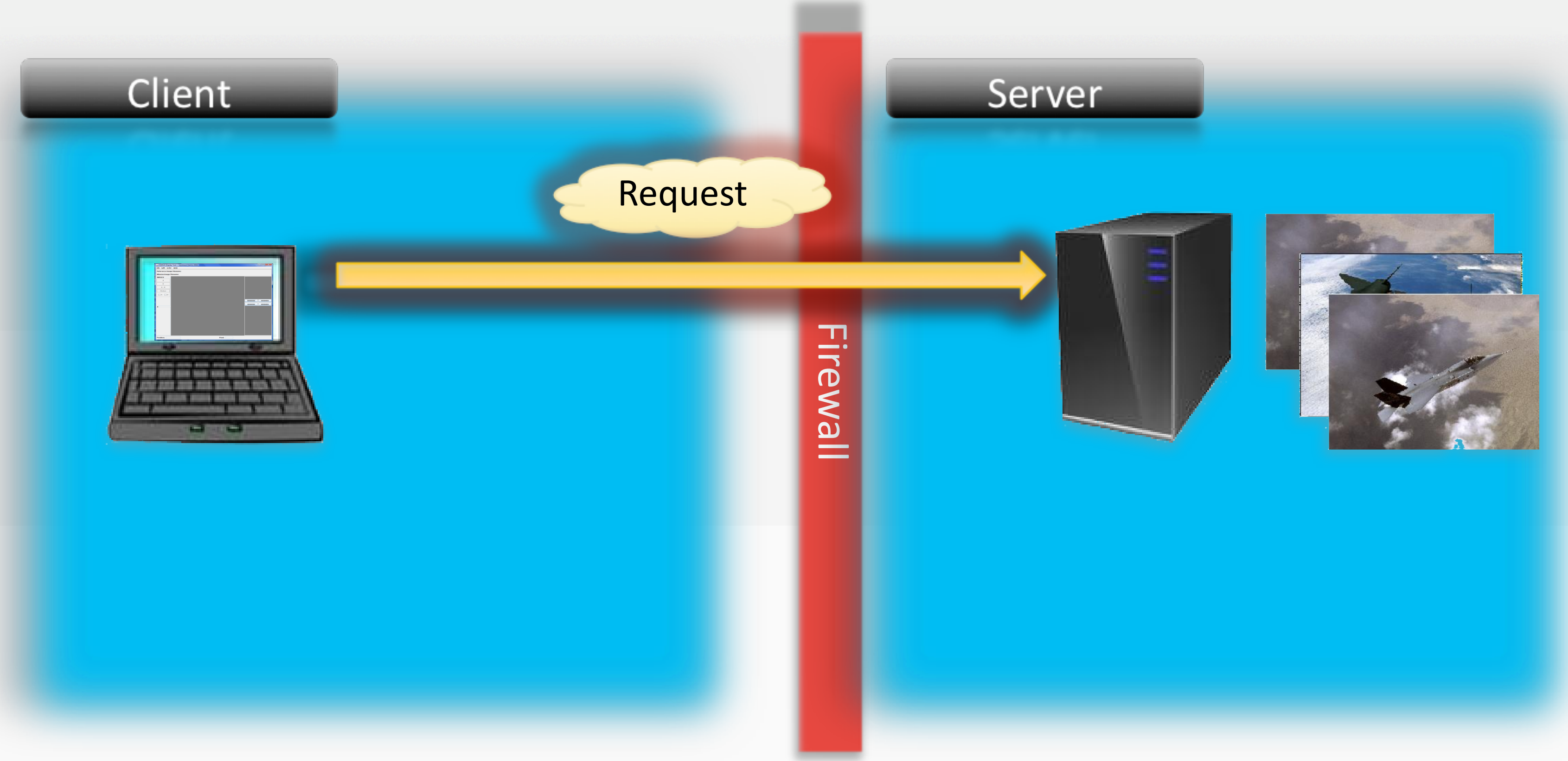
Solution: SOAized SHIELD

- Break SHIELD apart into server-client system
  - Server holds and processes the images
  - Client provides the interface
- This architecture is known as a Service Oriented Architecture (SOA)
- The SOA is implemented using Web Services

SOAized SHIELD
Data Flow Diagram



SOAized SHIELD
Data Flow Diagram



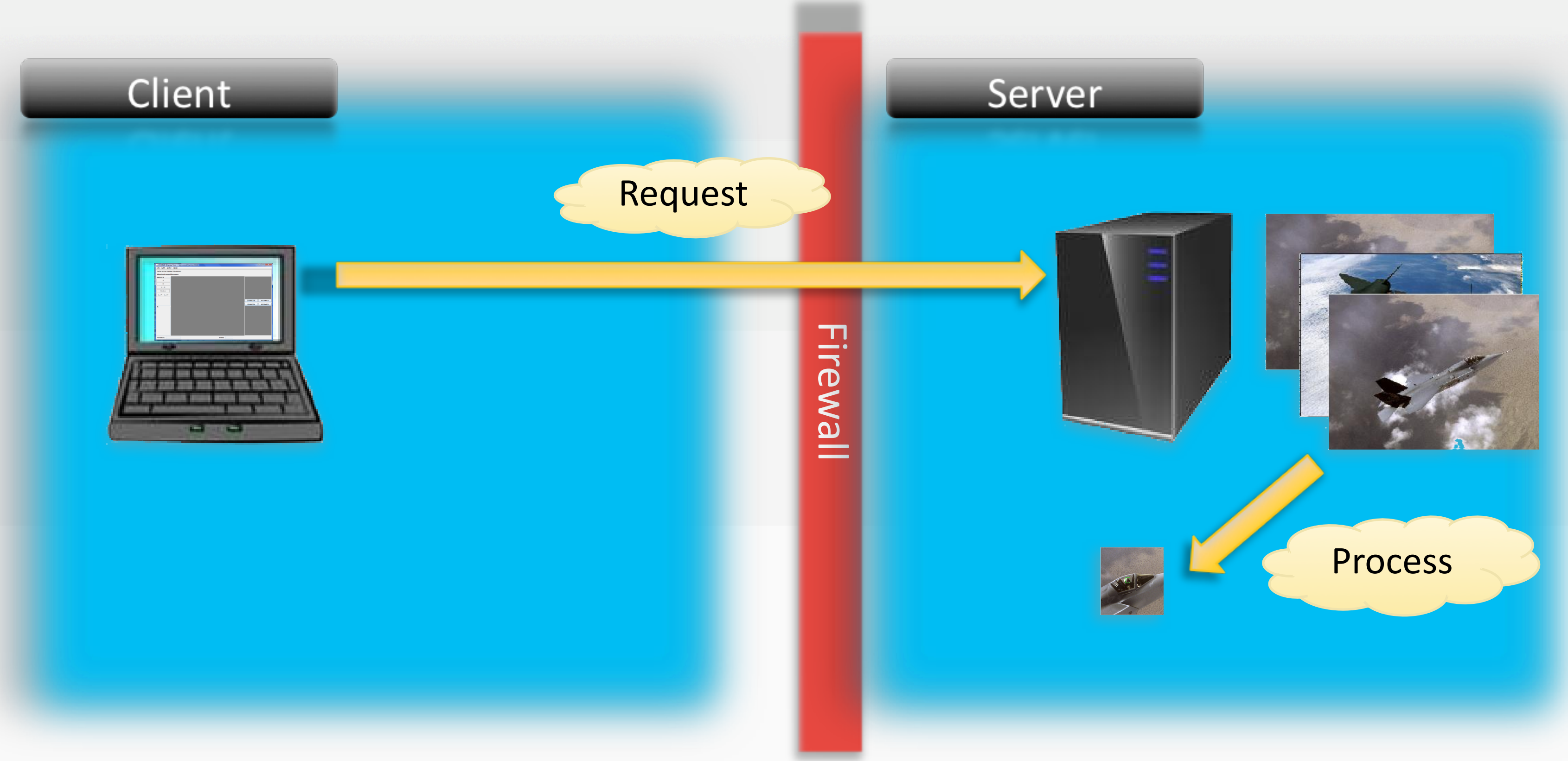
Client

Server

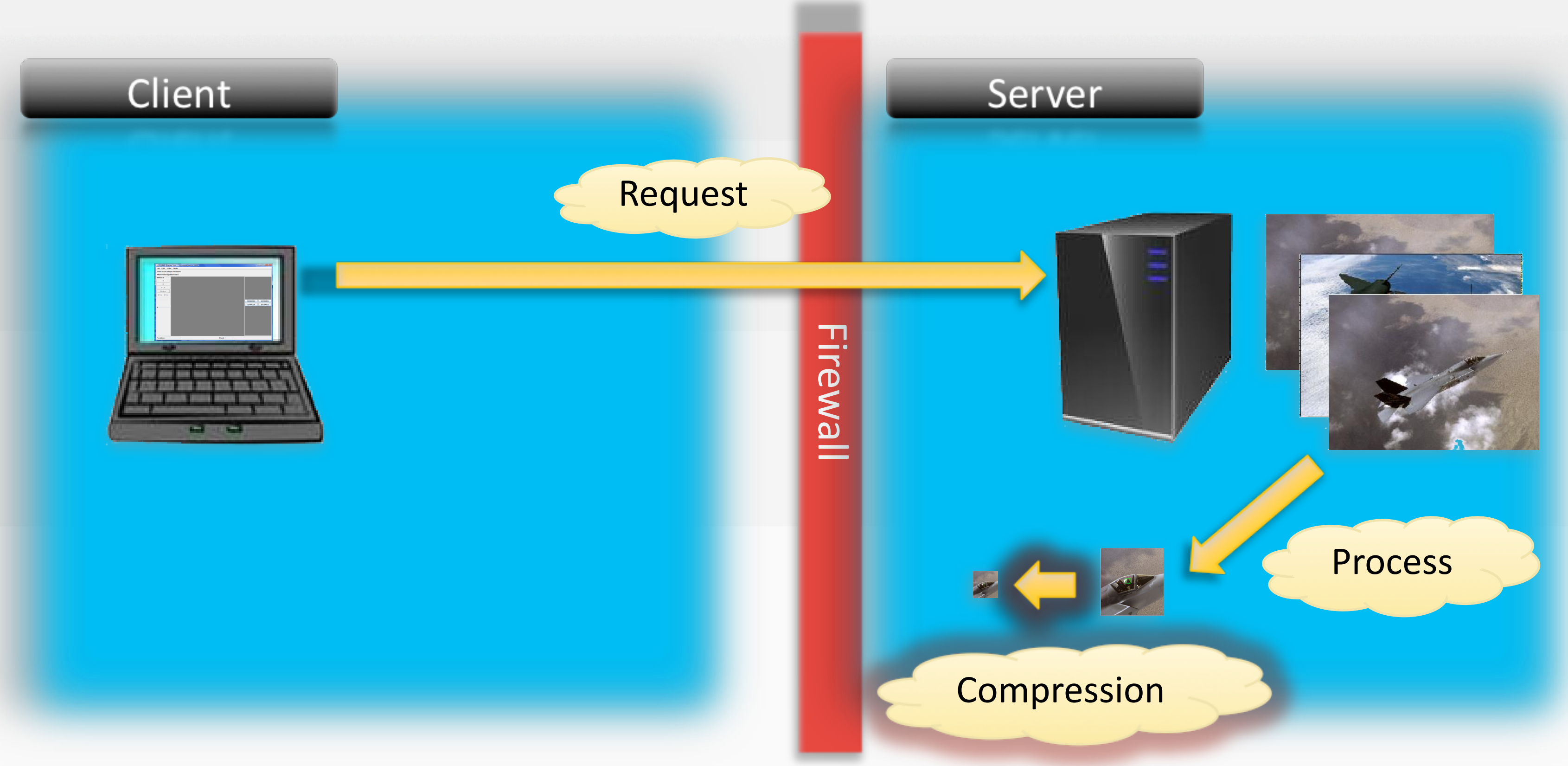
Request

Firewall

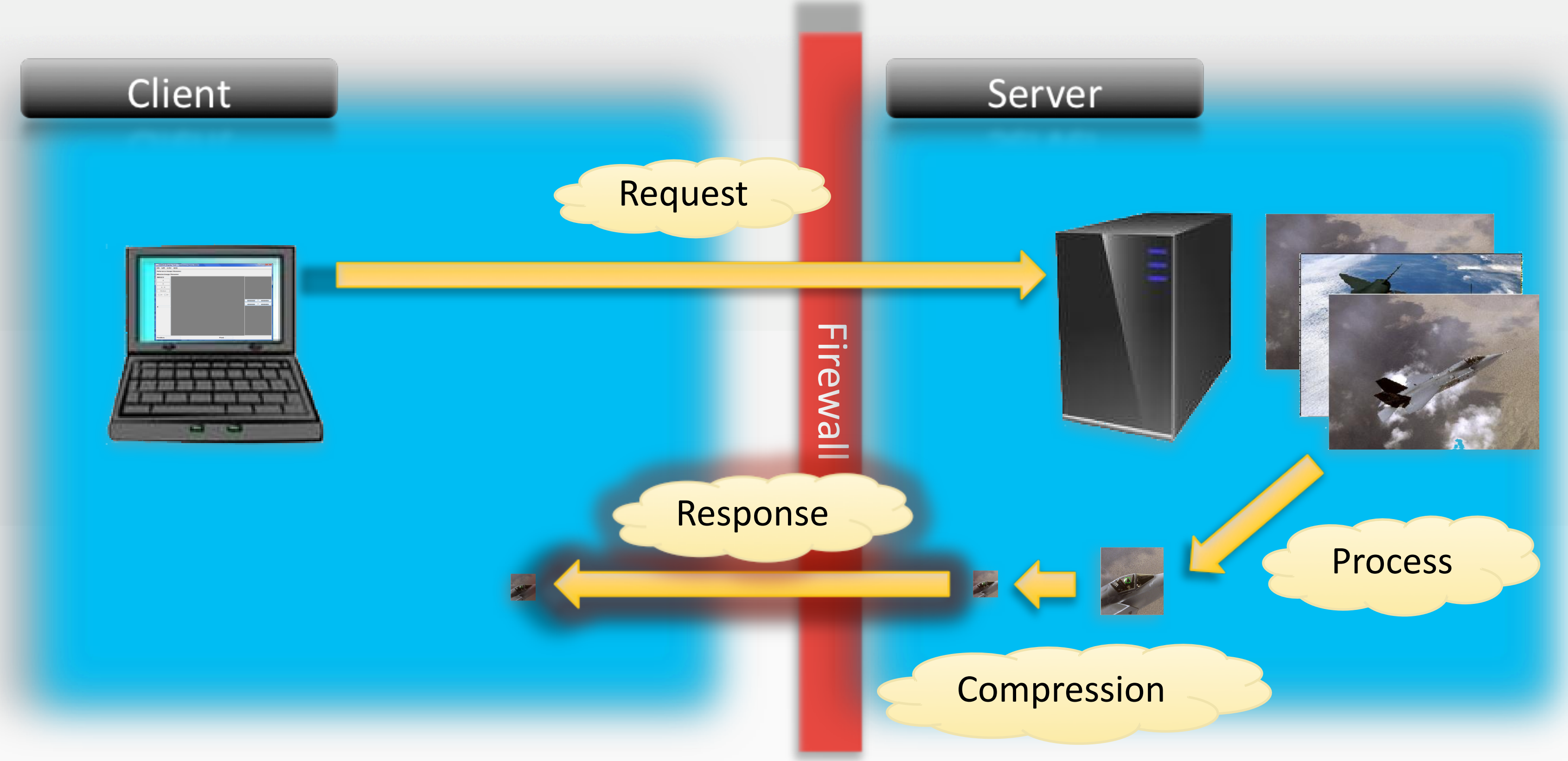
SOAized SHIELD
Data Flow Diagram



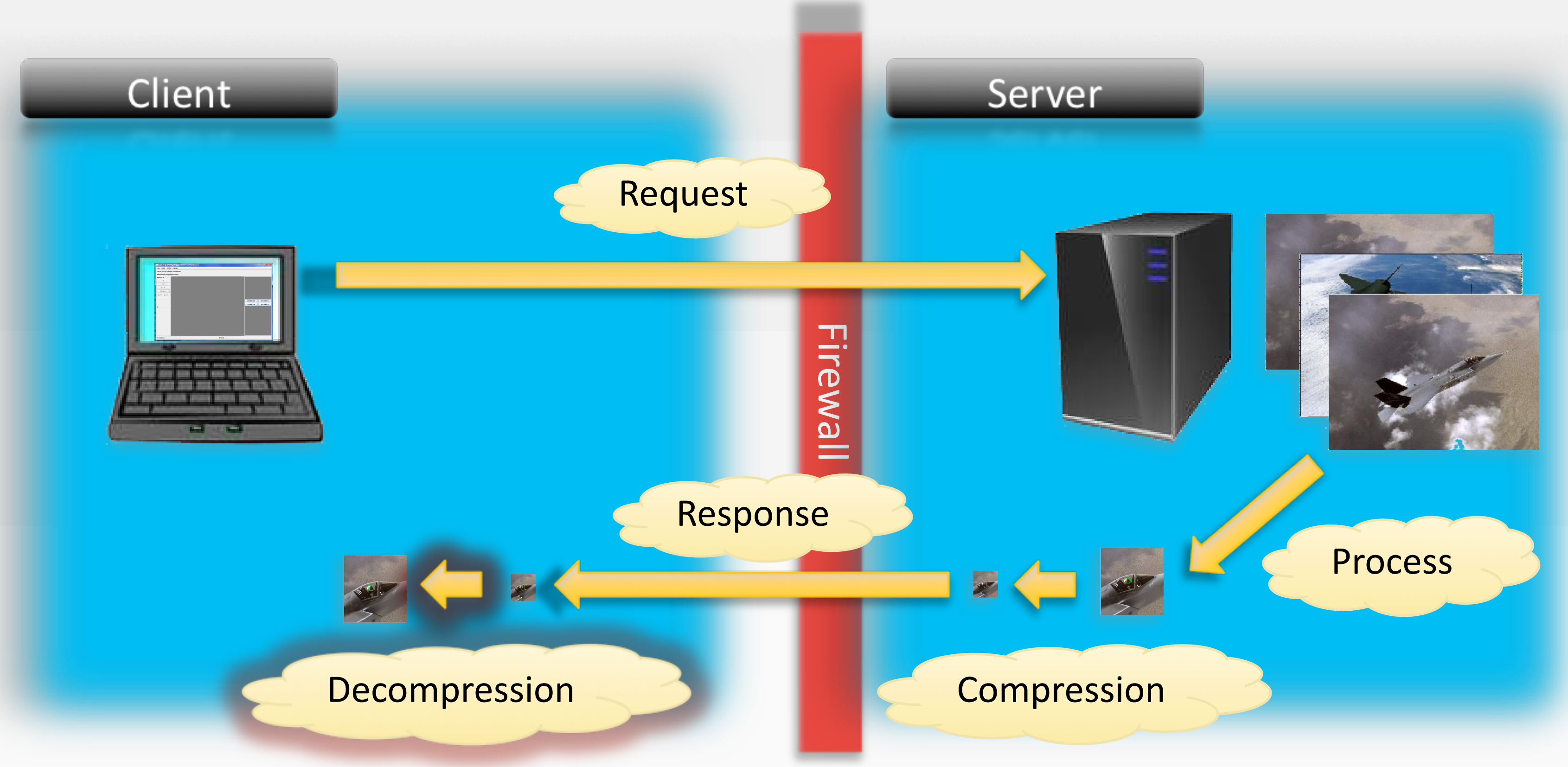
SOAized SHIELD
Data Flow Diagram



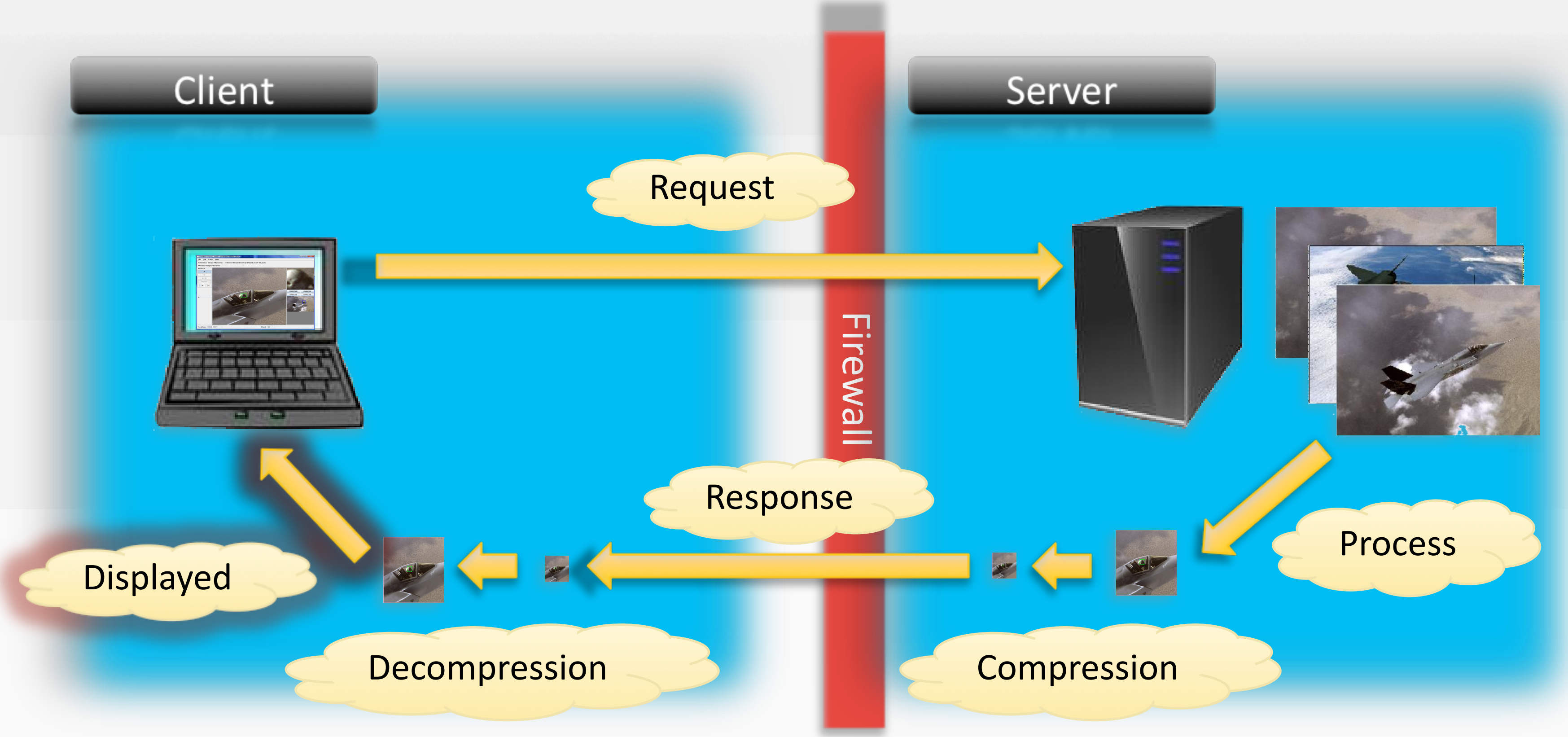
SOAized SHIELD
Data Flow Diagram



SOAized SHIELD
Data Flow Diagram

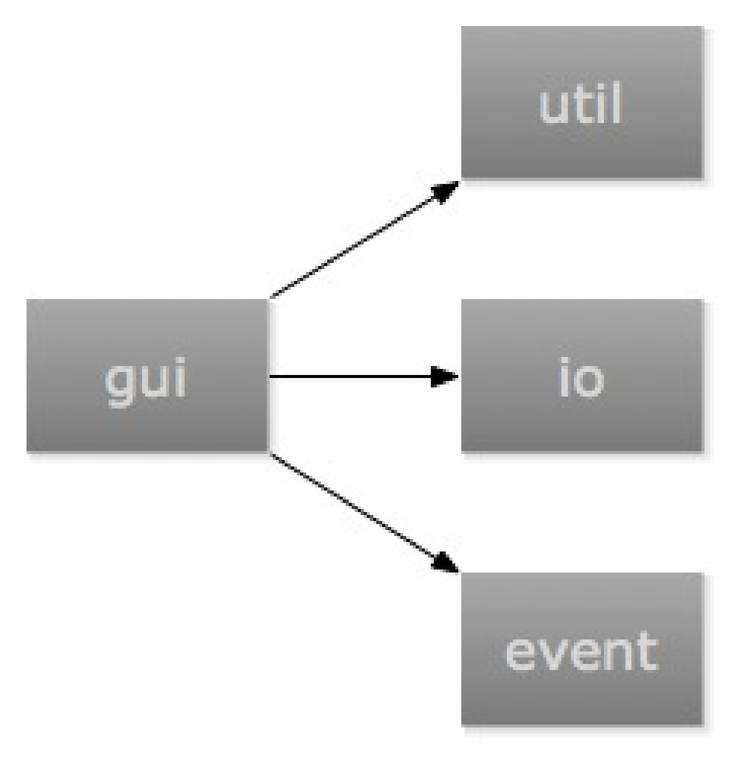


SOAized SHIELD
Data Flow Diagram



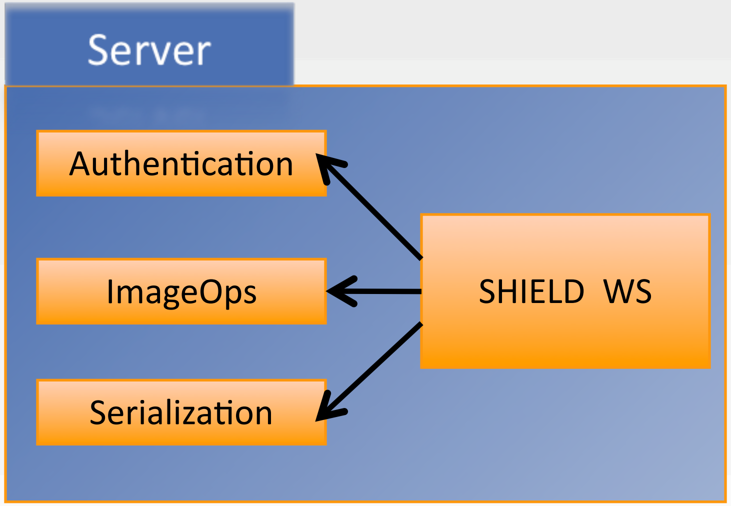
Original SHIELD Structure

- *gui* –User Interface
  - Maintains panels
  - Maintains images, viewing data
  - Performs image processing
- *event*
  - Controls gui operation
- *io* – Input/Output
  - Saving, file viewing
- *util* - Utility
  - User preferences, view settings



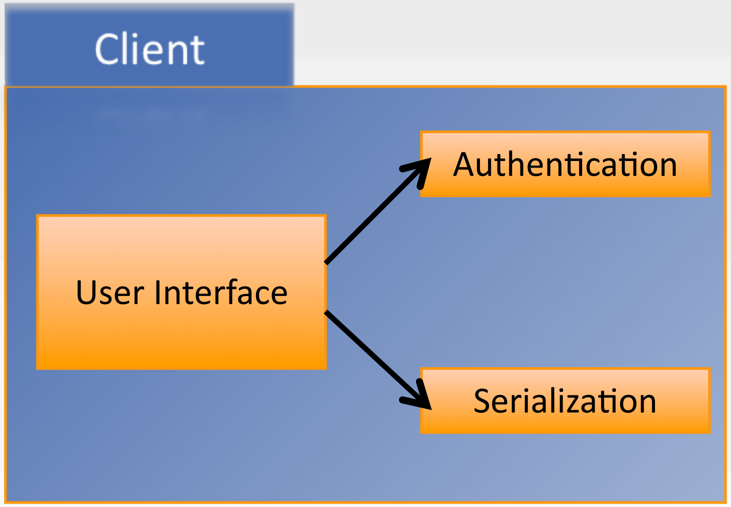
SOAized SHIELD
Architecture Overview - Server

- SHIELD – Web Service
  - Provides image retrieval and processing to the client
- Authentication
  - Authenticates clients attempting to use the service
- ImageOps
  - Performs image processing
- Serialization
  - Enables image passing over the network
  - Compresses the image data into JPEG format

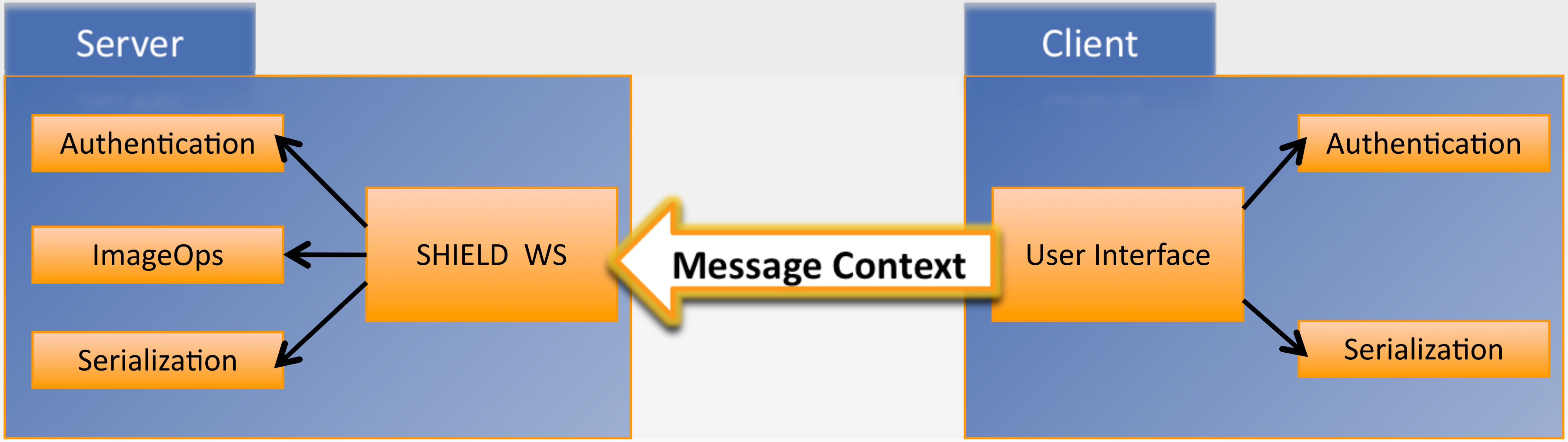


SOAized SHIELD
Architecture Overview - Client

- User Interface
  - Maintains viewing data, displays images
- Authentication
  - Contains login information
- Serialization
  - Reads the serialized data
  - Serializes messages



Architecture Overview



Server

Authentication

ImageOps

Serialization

SHIELD WS

Message Context

Client

User Interface

Authentication

Serialization

Highlight Features

- Client only needs to know the image data that it is going to render
  - It uses significantly fewer resources
- Images are compressed for transferring over the network
- Lockheed Martin's servers maintain control over the images
- Client does not need to store images

Major Differences

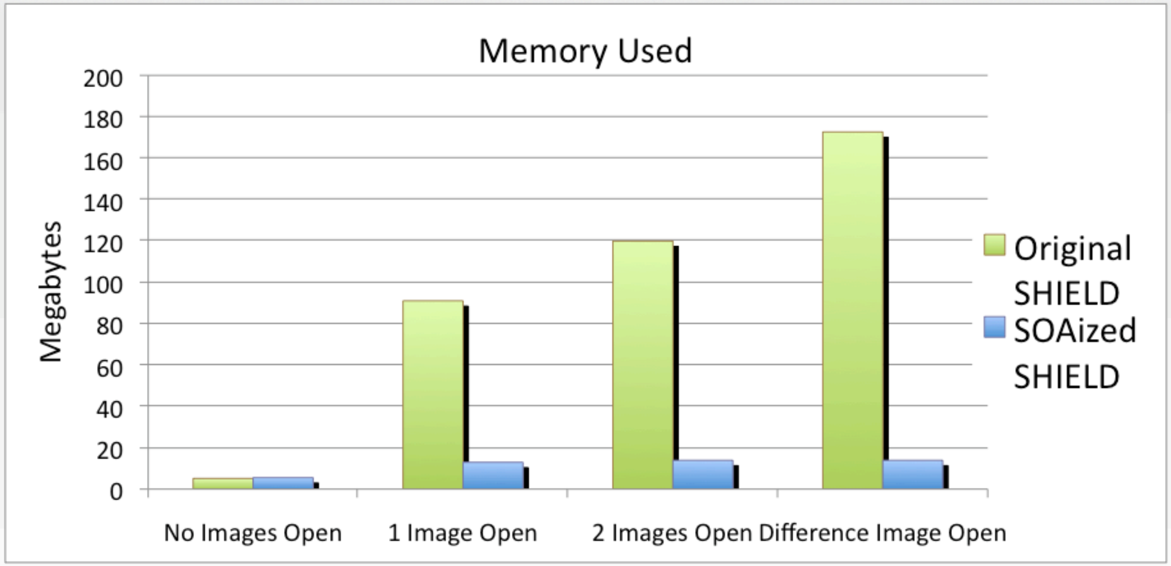
Original SHIELD	SOAized SHIELD
Application holds original images	Client holds image references Server holds images
Panels maintain their images	Panels maintain image viewing data
Panels perform all operations on their images	Panels request formatted images based on viewing data Server performs image processing

Design Process

- Rational Unified Process
  - Iterative approach
- Iterative – break up code in waves
  - I.e. move images to server, move image cropping to server, move colorization to server, etc…
- Concurrent
  - Preexisting code base
  - Able to work on multiple issues at once

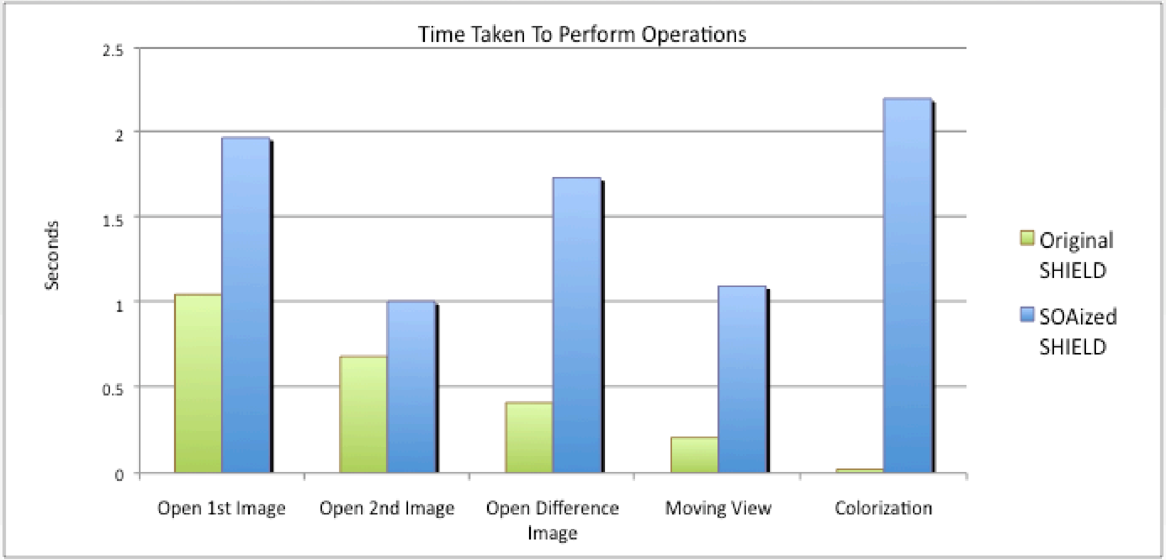
Client-side Memory Performance

- The chart shows the memory used to open 30 MB images
- SHIELD's memory usage scales with the size of the image
- SOAized SHIELD's memory usage is relatively constant



Time Comparison

- The chart shows the time taken to open 30 MB images and process them.
- SHIELD opens images locally
- SOAized SHIELD must pass the image over a network
- This data does not account for the time spent by the SHIELD user to download the image



Problems Encountered and Solved

- Setting up development environment
  - NetBeans, Subversion, Tomcat
- Breaking up SHIELD to provide greatest resource efficiency to the client
  - All image processing on server
- Passing images over the network quickly
  - Only displayed portion is sent to client as a JPEG
- Operating on images without losing quality
  - Operate on the images in their raw form
- Server memory leak

Deliverables

- Design and implement a SOA using Web Services
- Demonstrate the viability of breaking a home grown tool out, using a SOA
- Provide lessons learned about how to break a home grown tool out, to support being SOAized
- Perform timing tests and analysis to demonstrate improved performance over downloading and performing the actions locally

Positive Experiences

- Learning new technologies
- Working in a team on a complex project
- Working with a real world client
- Solving unexpected problems

Future Work

- The version of SHIELD we received had classified functionality stripped out of the program
- This functionality will have to be added to SOAized SHIELD
- Authentication will have to be tailored to Lockheed Martin’s system

Conclusion

- SOAized SHIELD can now be used on almost any computer with Java and an Internet connection
- Lockheed Martin can maintain greater control over their images

Poster Presentation

- Room: Du Bois Ballroom A
- Time: 2:30 pm – 4:00 pm
- Questions?