



Grizzly, Comet

Oleksiy Stashok



Agenda

- What is project Grizzly
 - > Introduction
 - > Community
- GlassFish v3: Powered by Grizzly
 - > Architecture
 - > Extension points
- GlassFish v3 Comet
- Atmosphere

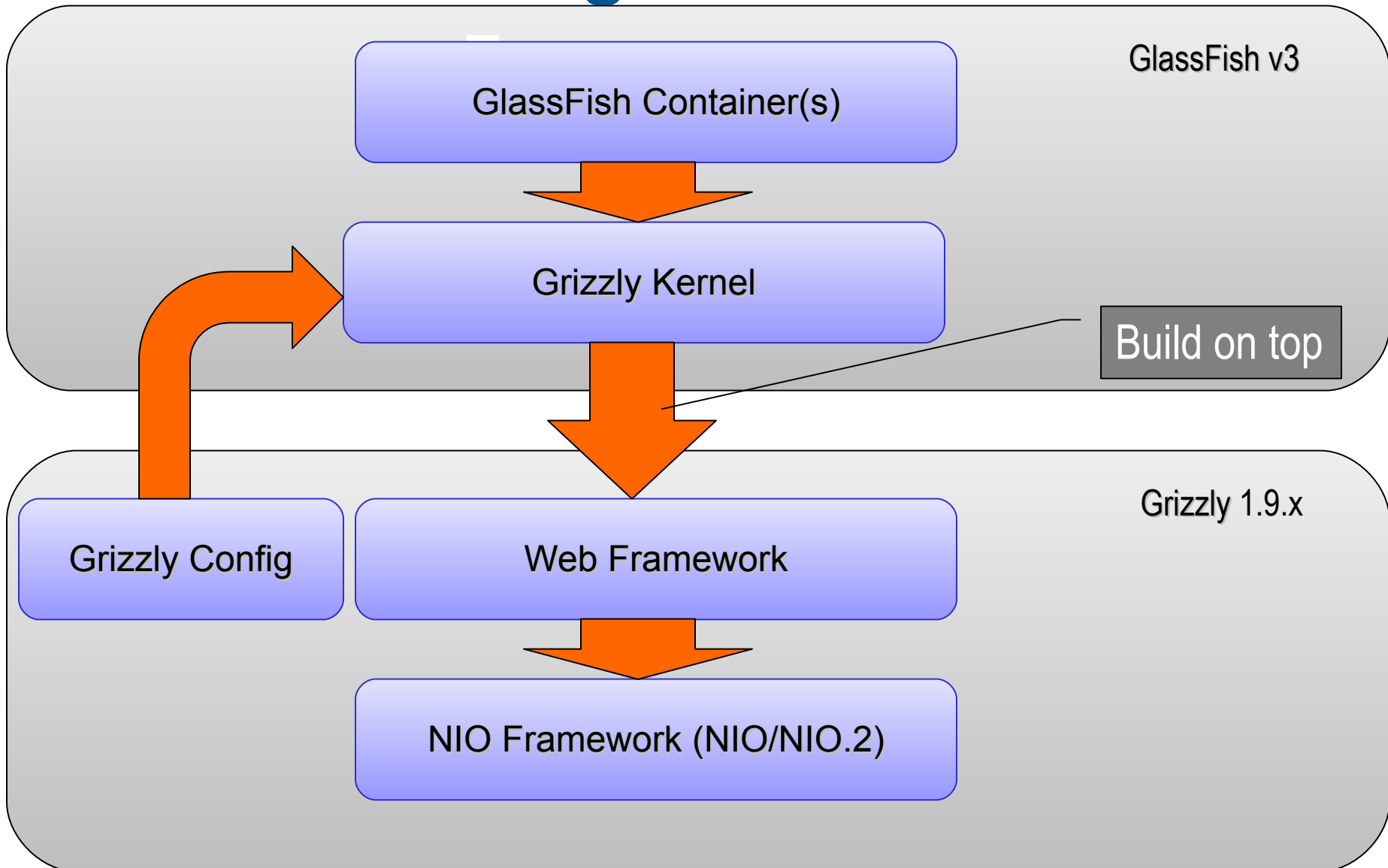
Introduction

- Open Source Project on java.net
 - > <https://grizzly.dev.java.net>
- Composed of 3 main frameworks
 - > NIO framework (Built using JDK NIO classes)
 - TCP/UDP/TLS components to build server/clients
 - Port Unification
 - > Web framework
 - HTTP, Async HTTP, XMPP, OSGi HttpService
 - > Comet framework
 - Comet, Continuations, Bayeux protocol, Message bus

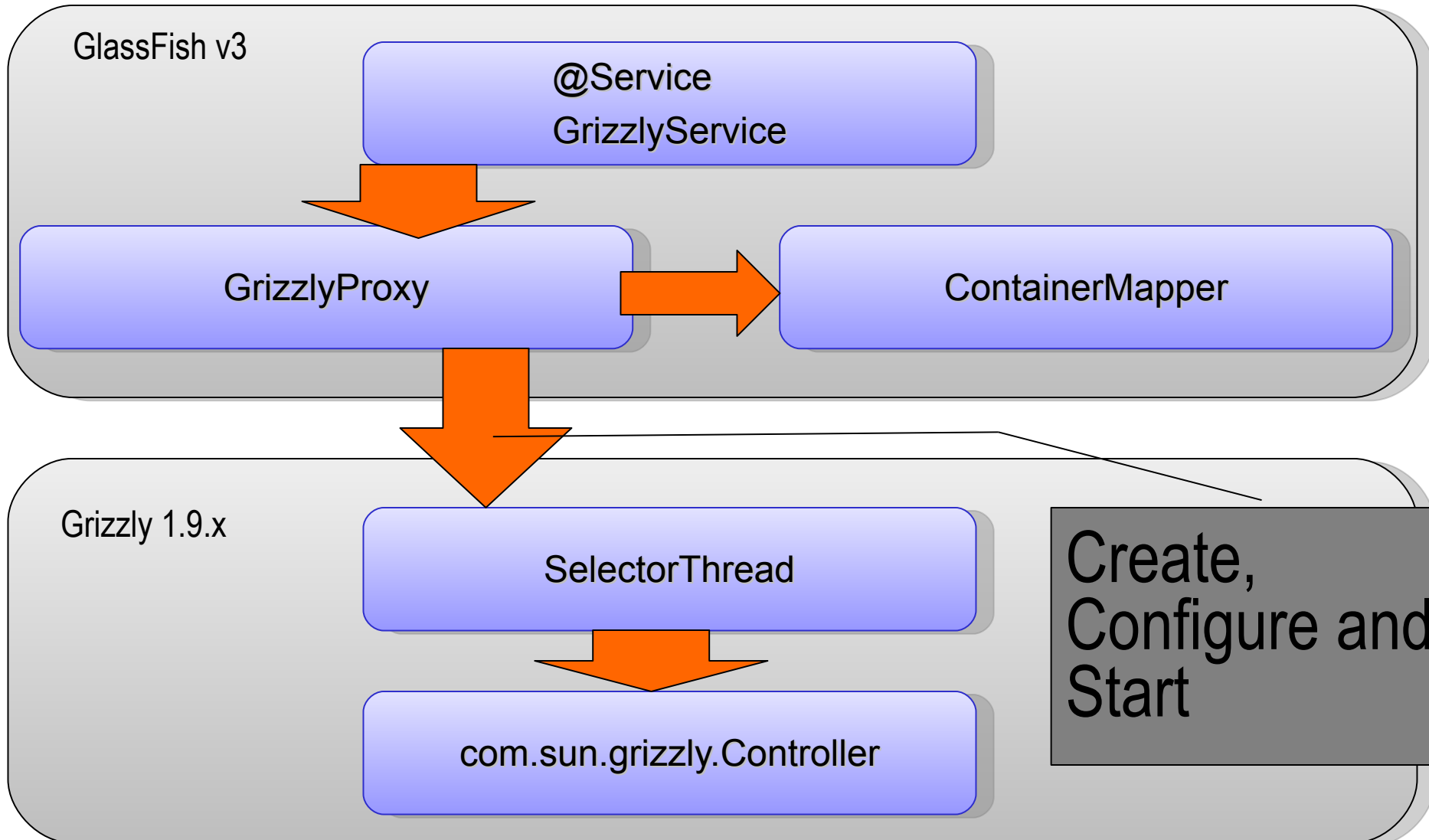
Community

- Very open community policy
 - > All project communications on public Grizzly mailing list. no internal, off line mailing list conversations
 - Everything on users@grizzly.dev.java.net or dev@grizzly.dev.java.net
 - > Project meetings open to anyone
- Project decisions are made by member votes
- Sun and non Sun committers

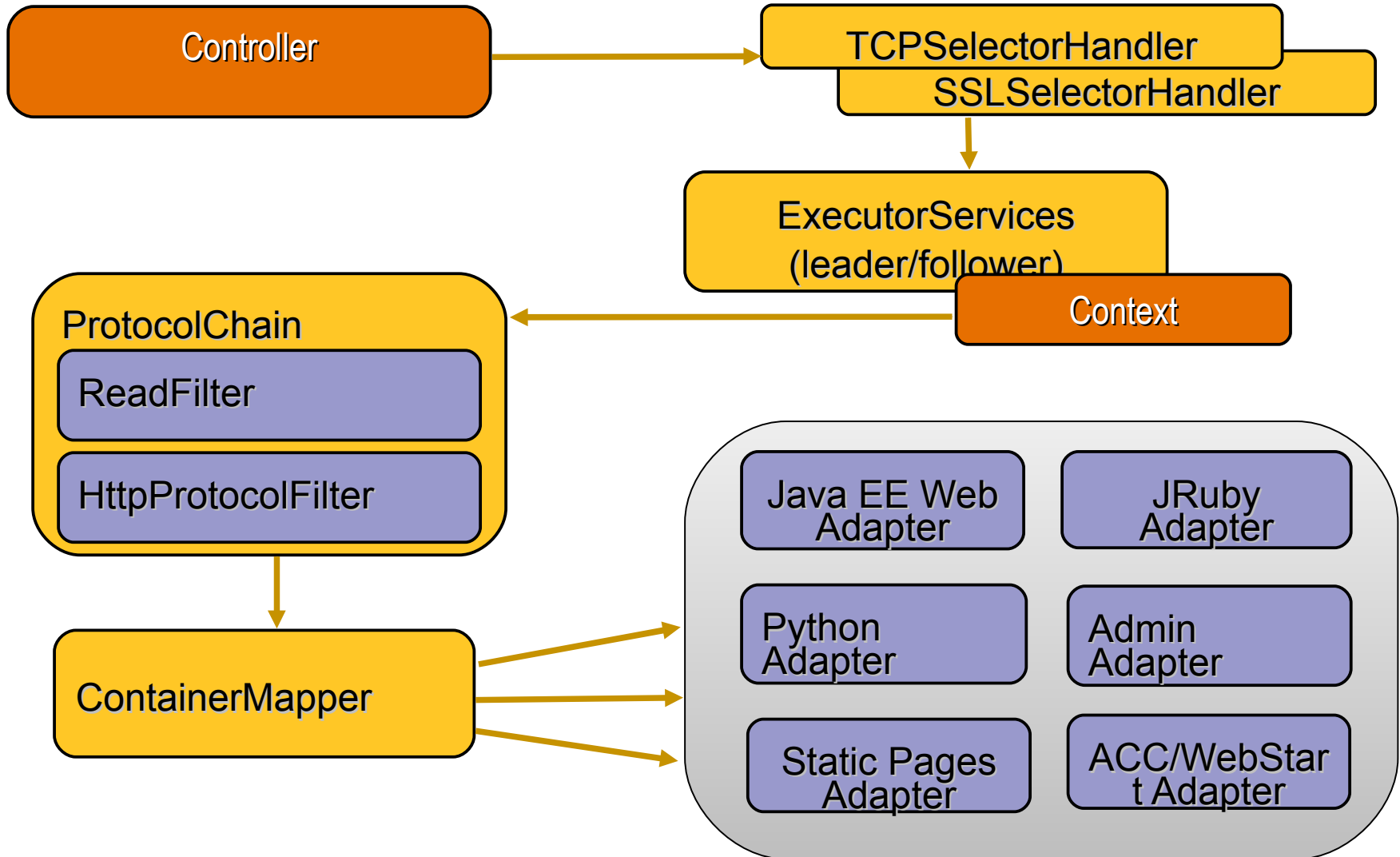
Architecture: High-level



Architecture: Web Framework



Architecture: Grizzly sight



Extending GlassFish v3 with a custom HTTP Container

- GlassFish v3 containers, which work on top of HTTP protocol, are GrizzlyAdapters: CoyoteAdapter, JRubyGrizzlyAdapter, AdminAdapter
- You can write custom GrizzlyAdapter and deploy it to GlassFish v3
 - > Implement GrizzlyAdapter
 - > Write a META-INF/grizzly-glassfish.xml

```

<adapters>
  <adapter
    context-root="/"
    class-name="com.sun.grizzly.http.HelloWorldAdapter"/>
</adapters>

```

asadmin deploy your_adapter_folder or asadmin deploy yourAdapter.jar

Grizzly configuration in GlassFish v3

- Grizzly can be configured via GlassFish v3 instance configuration file domain.xml
- Lets us configure Grizzly very finely
- <network-config>
 - > <transports> : TCP, UDP
 - > <protocols> : HTTP, SOAP/TCP etc...
 - > <listeners> : bind transports and protocols

Port Unification

- Naturally configured using Grizzly configuration
- Currently used in GlassFish v3 for:
 - > HTTP <->HTTPS redirection
 - > SOAP/TCP
 - > Your own usecase?

Port Unification #2 (single protocol)

```

<network-config>
  <transports>
    <transport name="tcp" />
  </transports>
  <protocols>
(1)  <protocol name="http">
      <http default-virtual-server="server"
          max-connections="250" compression="on"
          comet-support-enabled="true">
        <file-cache enabled="false" />
      </http>
    </protocol>
  </protocols>
  <network-listeners>
    <network-listener
      port="8080"
      protocol="http"
      transport="tcp"
      name="http-listener-1"
      thread-pool="thread-pool-1" />
  </network-listeners>
</network-config>

```

Port Unification #3 (multi protocol)

```
<protocols>
  <!-- Multi protocol description (HTTP + SOAP/TCP) -->
  <protocol name="multi-protocol">
    <port-unification>
      <protocol-finder protocol="http" classname="..." />
      <protocol-finder protocol="soap-tcp" classname="..." />
    </port-unification>
  </protocol>

  <!-- HTTP protocol description -->
  <protocol name="http">
    <http max-connections="250" default-virtual-server="server" />
  </protocol>

  <!-- SOAP/TCP protocol description -->
  <protocol name="soap-tcp">
    <protocol-chain>
      <protocol-filter classname="ws.transport.WSTCPProtocolFilter" />
    </protocol-chain>
  </protocol>
</protocols>

<network-listeners>
  <network-listener port="8080" protocol="multi-protocol" transport="tcp" />
</network-listeners>
```

Upgrading Grizzly in GlassFish v3

- GlassFish v3 will be shipped with Grizzly 1.9.18
- You can upgrade Grizzly easily by downloading Artifacts from <http://download.java.net/maven/2/com/sun/grizzly/>
- Maximum compatibility between the components maintained, e.g you can use 1.9.18 with 1.9.19
- Replace `${glassfish.home}/modules/grizzly-XXX` with your new version
- You can always build grizzly from source:
svn checkout <https://www.dev.java.net/svn/grizzly/trunk/code>

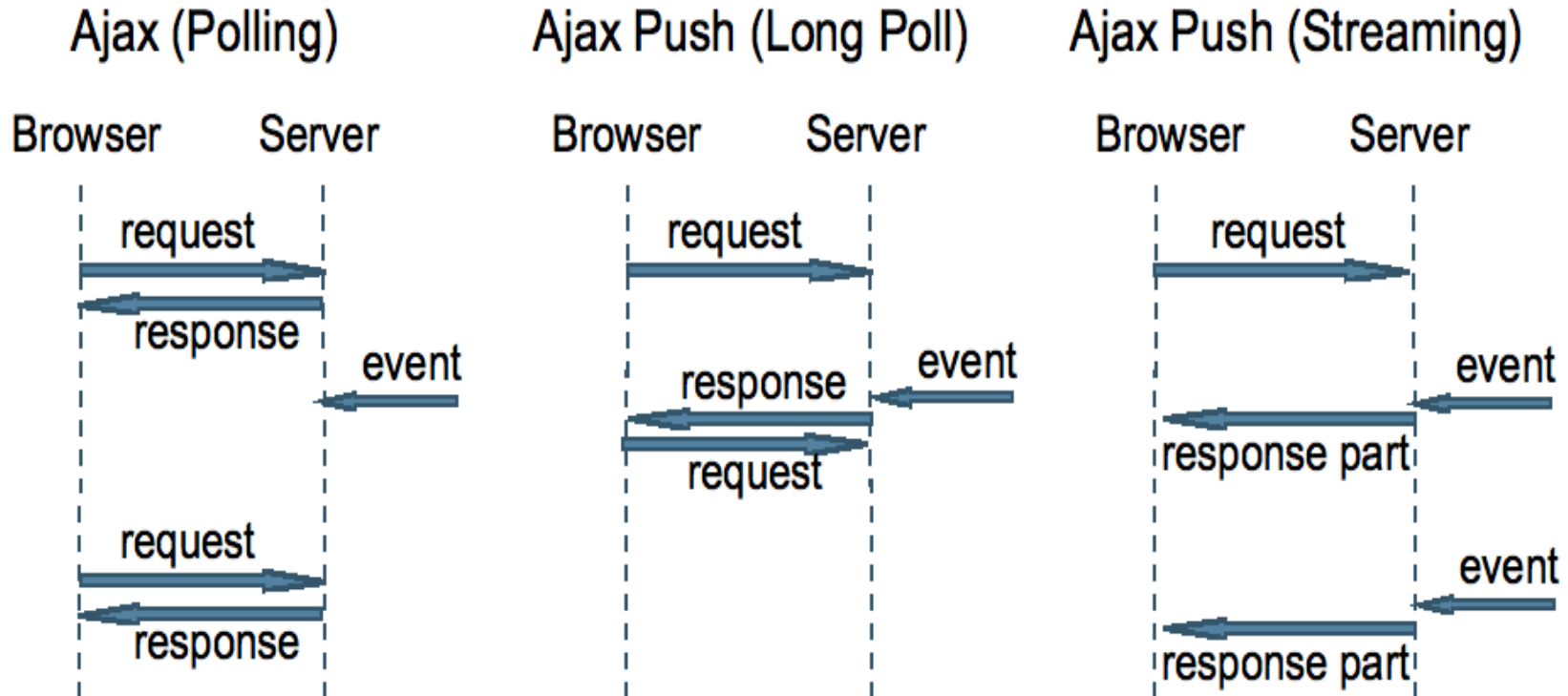
GlassFish Comet

- What is Comet?
 - > Allows webserver to push data to a browser using a long-held HTTP request without a browser explicitly requesting for this data
 - > Also known as Ajax push, HTTP server push, etc...
 - > Known example? - Gmail
- Enabled by default for all GlassFish v3 HTTP listeners

Why Comet?

- Responsive, low latency interaction for Web
- Keep clients (browsers) up-to-date with data arriving or changing on the server, without frequent polling
- Server and network do not have to deal with frequent polling requests, which check for update

HTTP polling vs. Comet



Atmosphere

- Atmosphere is the **first** POJO based framework using Inversion of Control (IoC) to bring Ajax Push/Comet to the masses.
 - > Support REST and Web
- Atmosphere allows writing powerful asynchronous applications without the need to know:
 - > How it works under the hood
 - Tomcat AIO is painful to learn! Jetty Continuation throws a RuntimeException (oh!), WebLogic AsyncServlet is complicated, Grizzly Comet is good, but only runs on top of GlassFish!
 - > Browser behavior differences
 - > Web Server differences
 - > Proxy/Network troubleshooting
 - > Knowledge of Clustering solution

Atmosphere

- Run on top of **any** Servlet 2.3 compliant Web Server
 - > **Natively:** GlassFish, Grizzly, JBoss, WebLogic, Jetty, Tomcat
 - > **Using a Blocking I/O Thread:** Resin, Tomcat 4, Sun WS, Oracle4J, Jetty 5, Winstone, etc.
- Support the upcoming Servlet 3.0 Async API
 - > Auto-Detect Servlet 3.0 API
 - (GlassFish v3, Jetty 8, soon Tomcat 7)
- Easily embeddable with:
RestEasy, Wink, DWR, Struts, Grails, Scala, Wicket, GWT, Cometd, JSF etc.

Resources

- Grizzly, Comet
 - > <https://grizzly.dev.java.net>
 - > dev@grizzly.dev.java.net, users@grizzly.dev.java.net
- Atmosphere
 - > <https://atmosphere.dev.java.net>
 - > dev@atmosphere.dev.java.net,
users@atmosphere.dev.java.net

Resources

Java EE 6 and GlassFish v3

- **Java EE 6 Home**

java.sun.com/javaee

- **Java EE 6 Downloads**

java.sun.com/javaee/downloads

- **Upcoming Training**

java.sun.com/javaee/support/training

Java EE 6

- **Sun GlassFish Enterprise Server v3 Home**

www.sun.com/glassfishv3

- **Community Page**

glassfish.org

- **The Aquarium Blog**

blogs.sun.com/theaquarium

- **White Papers/Webinars**

<http://www.sun.com/glassfish/resources>

GlassFish



Thank you

