

The background of the slide features a stylized landscape with layers of mountains in shades of purple and blue, and a dense forest of evergreen trees in the foreground. The overall color palette is cool and atmospheric.

APACHE  CON  
DENVER

WESTIN DENVER DOWNTOWN  
APRIL 7-9, 2014

# JAX-RS 2.0 with Apache CXF

Presented For The Apache Foundation By  
 **LINUX FOUNDATION**

APACHE CON

DENVER

WESTIN DENVER DOWNTOWN  
APRIL 7-9, 2014

Sergey Beryozkin  
Talend  
[sberyozkin.blogspot.com](http://sberyozkin.blogspot.com)

Presented For The Apache Foundation By  
**LINUX FOUNDATION**

# Apache CXF Overview

- Ships production-quality WS and RS modules
- Common runtime, transports, interceptors, features
- Runs in Servlet and OSGi containers
- Blueprint, Spring, CDI (in progress)
- CXF 3.0: JAX-RS 2.0, OAuth2, WSSec Streaming, WebSocket
- Major Focus on Security, Interoperability, Compatibility
- SubProjects: Fediz SSO, DOSGi CXF
- Philosophy: what works best for the user – wins

# Brief History of JAX-RS

- Server and client HTTP services made easy
- Great for REST and HTTP applications in general
- JAX-RS 1.0: Oct 2008, Paul Sandoz, Mark Hadley, Sun
- JAX-RS 1.1: Nov 2009
- Jersey: RI. RestEasy (Bill Burke). Apache CXF catches up.
- Apache Clerezza, Apache Wink; other projects adopt JAX-RS
- Oracle acquires Sun, Paul and Mark leave, is JAX-RS dead ?
- JAX-RS 2.0: Marec Potociar and Santiago Pericas-Geertsen
- JAX-RS 2.0: May 2013, major improvements
- Next: 2.1 planning is under way, 3.0 will follow

# JAX-RS 1.0 features

- Root and subresource service resources
- Annotations mark resource methods
- URI, headers, payloads captured as method parameters
- Message Body providers handle reads/writes
- Injected contexts help with the request processing
- Response: beans or beans and headers
- Very good support for conditional HTTP requests
- Basic server streaming and link builder support
- Code example...



# Typical JAX-RS API terms

- `@Path("root")`, `@Path("/book/{id}/{type}")`
- `@GET`, `@POST`, `@PUT`, `@DELETE`, custom HTTP methods
- `@PathParam`, `@QueryParam`, `@HeaderParam`
- `@Produces`, `@Consumes`, `MediaType`
- `UriInfo` context, `UriBuilder`, `URI`
- `EntityTag`, `CacheControl`, `Cookie`, `NewCookie`, `Variant`
- `SecurityContext`, `HttpHeaders`, `Request`, `UriInfo` contexts
- `ResponseBuilder`, `Response`
- `MessageBodyWriter`, `MessageBodyReader`, `Providers`

# JAX-RS 2.0 features

- Client API: sync and async calls, fluent style
- Top class support for asynchronous servers
- Server and client in/out filters and interceptors
- Targeted and dynamic binding of filters
- Extended exception hierarchy
- Injection support for sub-resources
- Support for grouping and converting parameters
- Link header builder
- Matching algorithm improvements



# Typical 2.0 API terms

- ClientBuilder, Client, WebTarget
- ClientRequestFilter, ClientResponseFilter
- ContainerRequestFilter, ContainerResponseFilter
- WriterInterceptor, ReaderInterceptor
- @NameBinding, @Priority
- Feature, Configuration, DynamicFeature
- ParamConverterProvider, BeanParam, Form
- Link, Link.Builder
- AsyncResponse, RequestInfo, RequestContext



## 2.0: Client API

- ClientBuilder → Client → WebTarget → Invoker
- SyncInvoker, AsyncInvoker (Future, InvocationCallback)
- JAX-RS 1.0 Response is reused on the client side (+1)
- Type-safe reads: Response.readEntity(Book.class), etc
- Client in/out filters and interceptors
- Support for setting SSL properties
- Sophisticated configuration and features support
- Nice mapping of error statuses to exceptions
- Code Example...

# 2.0: Server Filters

- In/out filters and reader/writer interceptors
- Advanced customization of requests and responses
- In filters can block requests, interceptors – reads/writes
- Global or method-specific filters (@NameBinding)
- Priority-ordered chains (@Priority)
- Bound statically or dynamically (DynamicFeature)
- Normal and exceptional responses go via out chains
- Code example...

## 2.0: Suspended requests

- The complexity is hidden by simple API
- The request can be suspended once or many times
- TimeoutHandler controls the number of suspensions
- Suspended requests can be resumed or canceled
- Requests resumed with responses or exceptions
- Notifications of the responses being sent
- Notifications of the disconnected clients (write time only)
- Code example...

## 2.0: Miscellaneous

- User-friendly exception hierarchy
- NotFoundException = new WebApplicationException(404)
- ResourceInfo Context: current resource class and method
- ResourceContext Context: injection into subresources
- HTTP Link Header Builder
- Matching across multiple roots with the same Path
- Server and client parameter conversion support
- Capturing of request parameters and contexts into beans
- Features and configuration
- Code example...

# My JAX-RS favourites

- All of JAX-RS is cool, but the following shines:
- 1.0: Request Context and Conditional HTTP – real gem
- 2.0: Asynchronous/suspended invocations – top design





# JAX-RS 2.0 Compliance

- CXF tested against jaxrstck-2.0\_26-Feb-2013
- Final 2.0 TCK is not available to Apache =>
- Technically complete but no 'formal compliance' tag
- Apache projects working with JCP specifications may be affected by the TCK unavailability



# Links

- CXF JAX-RS: [cxf.apache.org/docs/jax-rs.html](http://cxf.apache.org/docs/jax-rs.html)
- Examples: [github.com/sberyozkin/jaxrs20examples](https://github.com/sberyozkin/jaxrs20examples)
- JAX-RS: <https://jax-rs-spec.java.net/>



# Questions ?

APACHE  CON  
**DENVER**  
WESTIN DENVER DOWNTOWN  
APRIL 7-9, 2014

Presented For The Apache Foundation By  
 **LINUX FOUNDATION**