

2.– 5. September 2013
in Nürnberg



Herbstcampus

Wissenstransfer
par excellence

Die Ecke

Das vert.x Framework – modular, polyglott, asynchron

Eberhard Wolff

adesso AG

vertex

polyglot - asynchronous - modular

Eberhard Wolff

@ewolff

adesso

Architecture and Technology Manager

adesso AG

vertex

polyglot - asynchronous - modular

Eberhard Wolff

@ewolff

adesso

Architecture and Technology Manager

adesso AG

About me

- Architecture & Technology Manager at adesso
- Speaker
- Author
- Blog: <http://ewolff.com>
- Twitter: @ewolff
- <http://slideshare.com/ewolff>
- <http://prezi.com/user/ewolff>

vertx

polyglot - asynchronous - modular

Eberhard Wolff

@ewolff

adesso

Architecture and Technology Manager

adesso AG

polyglot

vert.x supports

Java	JavaScript (DynJS/ Rhino)
Groovy	Coffeescript
Clojure (post 2.0)	
Scala (post 2.0)	Xtend
Python (Jython)	Ruby (JRuby)

Natural API

**language specific
layer**

vert.x core (Java)

JavaScript → Java

```
fileSystem.writeFile = function(path, data, handler) {  
  if (typeof data === 'string') {  
    data = new org.vertx.java.core.buffer.Buffer(data);  
  }  
  jfs.writeFile(path, data, wrapHandler(handler));  
  return fileSystem;  
}
```

vertx

polyglot - asynchronous - modular

Eberhard Wolff

@ewolff

adesso

Architecture and Technology Manager

adesso AG

vertx

asynchronous

Eberhard Wolff

© 2011 Pearson Education, Inc.

Embedded vert.x

```
Vertx vertx = VertxFactory.newVertx();
vertx.createHttpServer()
    .requestHandler(new Handler<HttpServerRequest>() {
        public void handle(HttpServerRequest req) {
            req.response.headers().set("Content-Type",
                "text/html; charset=UTF-8");
            req.response()
                .end("<html><body><h1>Hello!</h1></body></html>");
        }
    }).listen(8080);
```

JavaScript

```
var vertx = require('vertx')
```

```
vertx.createHttpServer()
```

```
  .requestHandler(function(req) {
```

```
    req.response.headers().set('Content-Type',  
    'text/html; charset=UTF-8');
```

```
    req.response.end(
```

```
      '<html><body><h1>Hallo!</h1></body></html>');
```

```
  }).listen(8080);
```



```
{  
  "rant" :  
    "Anonymous Inner Classes are a  
    sorry replacement for closures"  
}
```

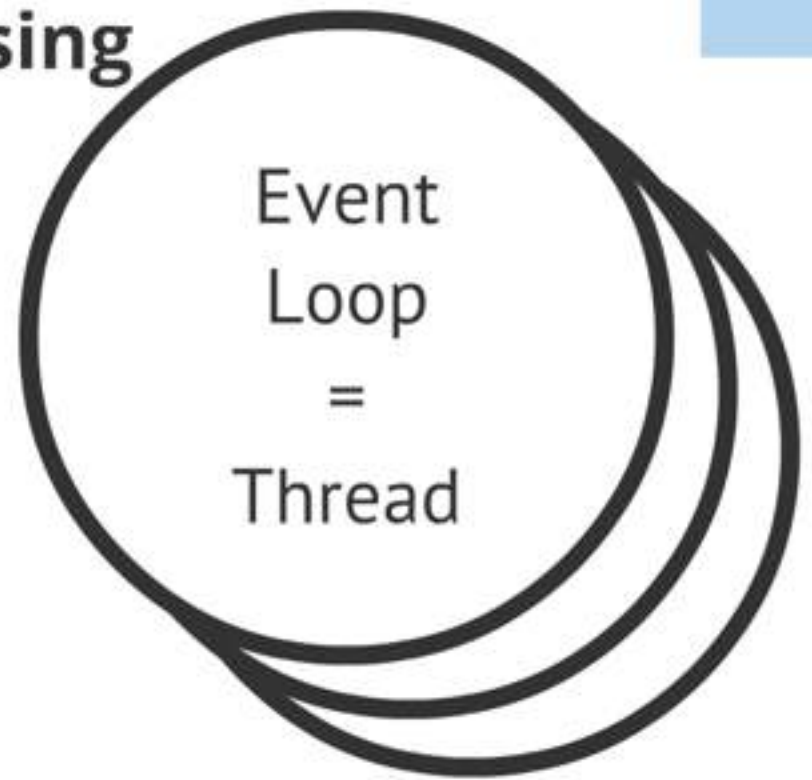
Asynchronous Processing in a Nutshell

One thread - many network
connections

Thread Loop = # CPU cores

Do not block the Event Loop!

Worker outside Event Loop
for long running operations



vertx

polyglot - asynchronous - modular

Eberhard Wolff

@ewolff

adesso

Architecture and Technology Manager

adesso AG



How can modules exchange information?

Vertices



Let's share some DDA data



modular

Vertices

Smallest components for `vert.x`

```
public class HttpServerExample extends Verticle {  
    public void start() {  
        vertx.createHttpServer()  
            .requestHandler(new Handler<HttpRequest>() {  
                public void handle(HttpRequest req) {  
                    req.response.headers().put("Content-Type",  
                        "text/html; charset=UTF-8");  
                    req.response  
                        .end("<html><body><h1>Hello!</h1></body></html>");  
                }  
            }).listen(8080);  
    }  
}
```



```
vertx run server.HttpServerExample  
-cp "classes:mykewlverticle.jar"
```

```
vertx run HttpServerExample.js
```

```
vertx run HttpServerExample.java  
-instances 10 -cluster
```

Modules

```
{  
  "main": "server.RestServer"  
}
```



start via command line

```
vertx runmod com.ewolff~rest-server~0.0.1
```

deploy in code:

```
vertx.deployModule('com.ewolff~rest-server~0.0.1');
```

```
public class RestServer  
  extends Verticle {
```

```
  ...
```

```
}
```


How can modules
exchange information?

Shared data

- Shared Maps or Sets
- May only store immutable data (String, Integer, ...)
- Immutable - No concurrency issues
- Limited scaling

Verticle Verticle Verticle



JSON



Event Bus

Point 2 point & publish / subscribe

Let's store some JSON
data in  mongoDB

Deploy MongoDB Module

```
JsonObject mongoPersistorConfig = new JsonObject()
    .putString("address", "vertx.mongopersistor");
    .putString("db_name", "test_db");

container.deployModule(
    "io.vertx~mod-mongo-persistor~2.0.0-final",
    mongoPersistorConfig, 1);
```

Save

```
JsonObject command = new JsonObject()
    .putString("action", "save")
    .putString("collection", "customers")
    .putObject("document", data);
```

```
vertx.eventBus().send("vertx.mongopersistor", command,  
    new Handler<Message<JsonObject>>() {  
        public void handle(Message<JsonObject> mongoReply) {  
            ...  
        }  
    });
```

Find

```
JsonObject command = new JsonObject()
    .putString("action", "findone")
    .putString("collection", "customers")
    .putObject("matcher",
        new JsonObject().putString("_id", "42"));

vertx.eventBus().send("vertx.mongopersistor", command,
    new Handler<Message<JsonObject>>() {
        public void handle(Message<JsonObject> reply) {
            JsonObject found = reply.body().getObject("result");
        }
    });
```

Find in JavaScript

```
var eb = vertx.eventBus;  
eb.send('vertx.mongopersistor', {  
  action : 'findone',  
  collection : 'customer',  
  matcher : {  
    "_id" : 42  
  }  
}, function(reply) {  
  var found = reply.result;  
});
```


Event Bus

- fits asynchronous model
- Decoupling: only data - no method calls
- Can be distributed
- JSON - support polyglot modules

vert.x Module Concept

Independent deployment easily possible

- JSON data - no shared classes

Have you ever tried to deploy modules independently in Java EE?

vertex

polyglot - asynchronous - modular

Eberhard Wolff

@ewolff

adesso

Architecture and Technology Manager

adesso AG

vertex



Platform

- Cloud
- Mobile
- Web

Business

- Analytics
- CRM
- Marketing
- Operations

Tools

- Cloud
- Mobile
- Web

Details

- Cloud
- Mobile
- Web

[http://v](#)

[https://v](#)

[https://v](#)

[https://v](#)

[http://v](#)

[http://v](#)

[und-Eve](#)

[http://a](#)

Feature

- TCP/SSL
- WebSockets
- SockJS

Modules

- MongoDB
- JDBC
- Redis
- Memcached
- Mailer
- Web Server
- Session Manager
- Shell
- Auth manager
- Work Queue

Tools

- Maven PlugIn
- Testtool for in-container testing
- Openshift support

Details

- 2.0.1
- Apache 2.0 license
- Eclipse project
- Developers from Red Hat and Pivotal

Links

<http://vertx.io/tutorials.html>

<https://github.com/vert-x>

<https://github.com/vert-x/vertx-gradle-template>

<https://github.com/vert-x/vertx-maven>

<http://www.infoq.com/news/2013/05/high-volume-vertx>

<http://www.heise.de/developer/artikel/vert-x-asynchrone-und-Event-getriebenes-Java-Webframework-1587065.html>

<http://jaxenter.de/artikel/vertx-alles-wird-alles>

vertx

polyglot - asynchronous - modular

Eberhard Wolff

@ewolff

adesso

Architecture and Technology Manager

adesso AG