

Handling Rejections



Rejections

If a request doesn't match a filter directive, it's rejected

- reject = pass the request to another branch in the routing tree
- a rejection is NOT a failure

Rejections are aggregated

```
get {  
  // Route 1  
} ~  
post {  
  // Route 2  
} ~  
parameter('myParam') { myParam =>  
  // Route 3  
}
```

← if request is not a GET, add rejection

← if request is not a POST, add rejection

← if request has a query param, *clear the rejections list*
(other rejections might be added within)

We can choose how to handle the rejection list

Recap

Handle rejections

- rejection handler = function between a rejection list and a route
- rejection handling directives

```
val myHandler: RejectionHandler = rejections => ...
```

```
handleRejections(myHandler) {  
  // Route with potential rejections ← rejections are "caught" and handled  
}
```

Implicit rejection handlers and builders

```
implicit val myDefaultHandler = RejectionHandler  
  .newBuilder()  
  .handle { ← careful with ordering  
    case r: MethodRejection => ...  
  }  
  .handleAll[MissingQueryParameterRejection] { r => ← alternative  
    ...  
  }  
  .handleNotFound { ... } ← return a Route in any case  
  .result()
```

Akka rocks

