Workshop WPS

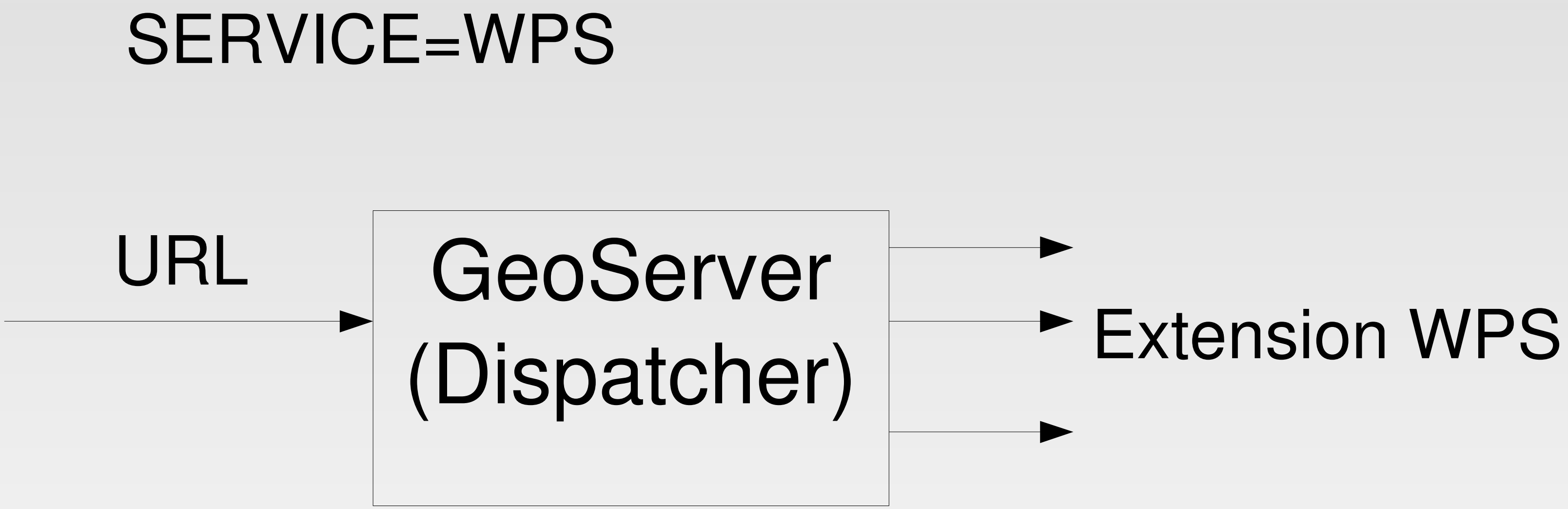
Hello WPS

ubuntu

Introduction

- GeoServer can be extended by extensions
- Extension adds some functionality – in our case handles WPS requests
- Geoserver in a case of URL (or in a body of POST request) with parameter SERVICE with value WPS transferes URL to the extension and waits for response

Principle



Extension installation

- Unpack to WEB-INF/lib
- Server restart

WPS Extension

- Reads loaded Java Beans and search for those that implements GeoServerProcess interface
- From that beans reads annotations for
  - DescribeProcess
  - DescribeResult

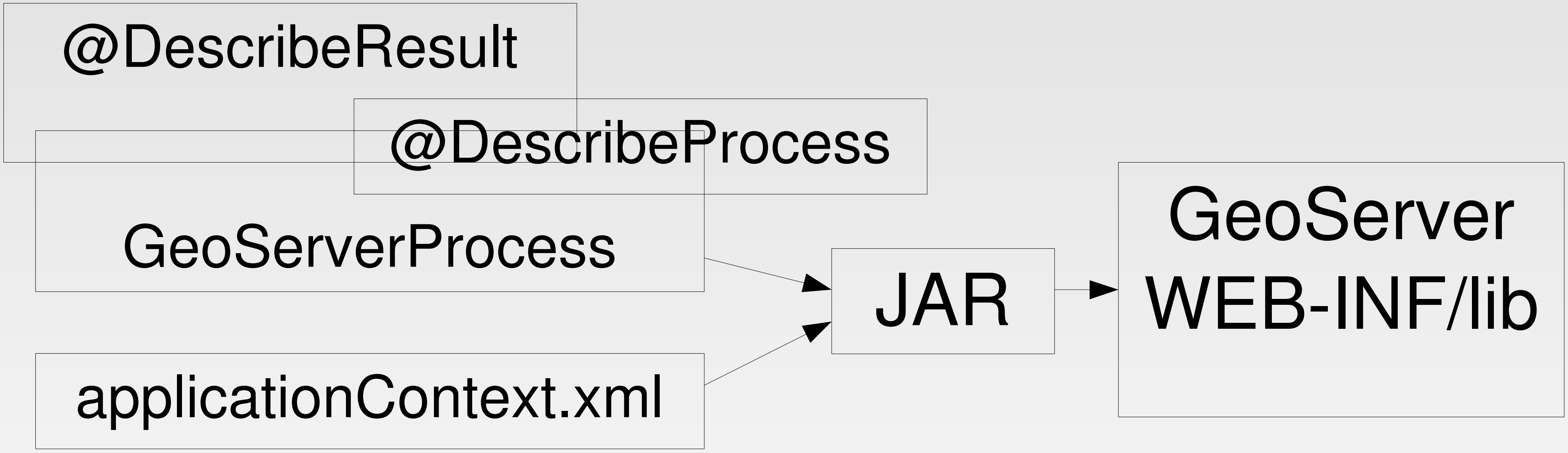
Load Java Bean

- XML configuration file
- applicationContext.xml
- Part of JAR for installation
- Installation – copy to WEB-INF/lib

Custom process

- JAR file
- Class that implements interface GeoServerProcess
- Class described by annotations DescribeResult
- File applicationContext.xml

Principle


@DescribeResult
@DescribeProcess
GeoServerProcess
applicationContext.xml
JAR
GeoServer
WEB-INF/lib

Custom process

```
import org.geotools.process.factory.DescribeParameter;

import org.geotools.process.factory.DescribeProcess;

import org.geotools.process.factory.DescribeResult;

import org.geoserver.wps.gs.GeoServerProcess;


@DescribeProcess(title="helloWPS", description="Hello WPS
  Sample")

public class HelloWPS implements GeoServerProcess {

   @DescribeResult(name="result", description="output result")

   public String execute(@DescribeParameter(name="name",
  description="name to return") String name) {

       return "Hello " + name;

   }

}
```

Custom proces

```
<?xml version="1.0" encoding="UTF-8"?>

  <!DOCTYPE beans PUBLIC "-//SPRING//DTD BEAN//EN"
  "http://www.springframework.org/dtd/spring-beans.dtd">

  <beans>

    <bean id="helloWPS"
  class="org.geoserver.hello.wps.HelloWPS"/>

  </beans>
```

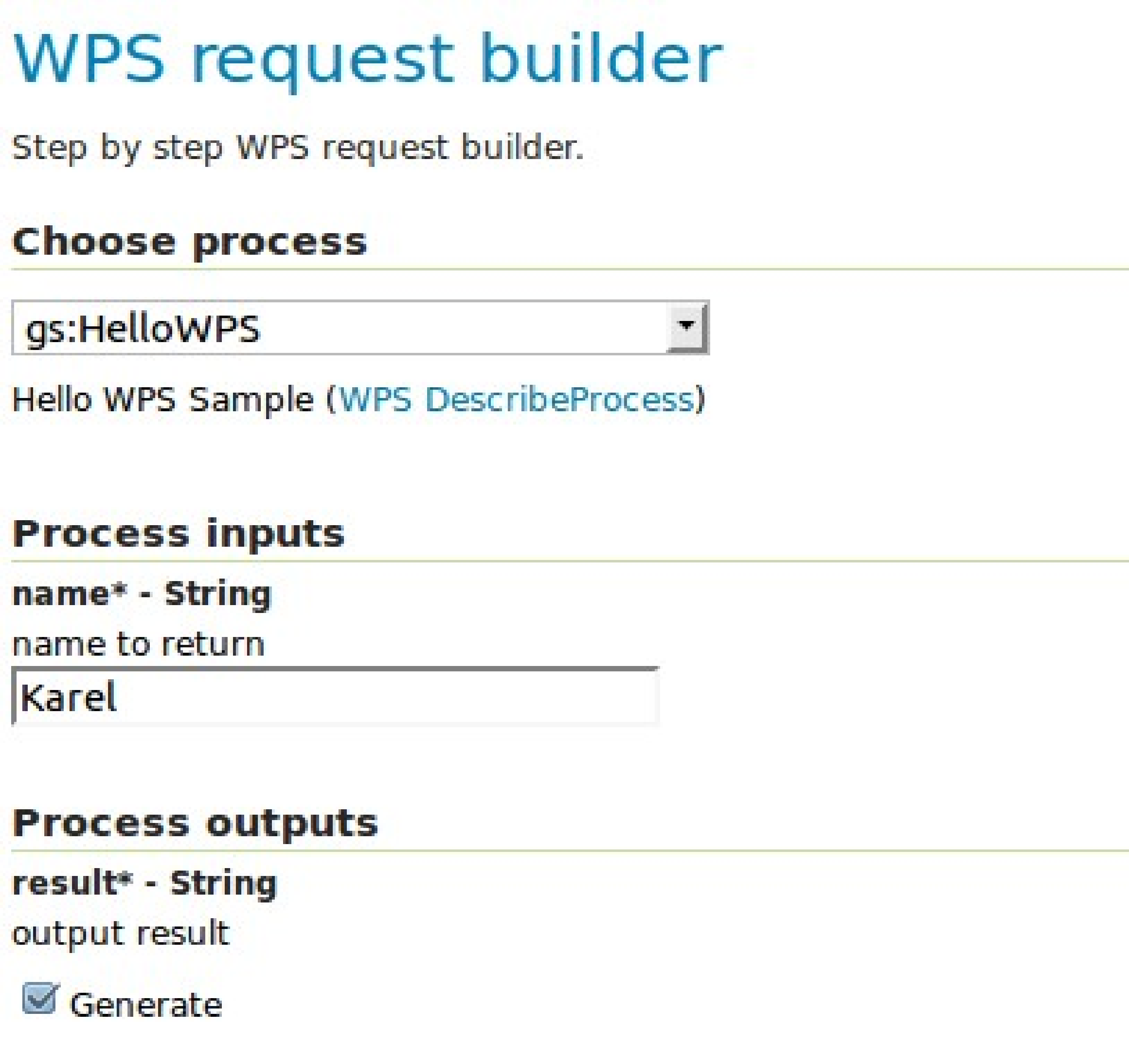
Custom proces

- JAR file
  - Class HelloWPS
  - File applicationContext.xml

Custom proces

- JAR file
    - Copy to WEB-INF/lib/
- Server restart

Testing



WPS request builder

Step by step WPS request builder.

Choose process

gs:HelloWPS ▾

Hello WPS Sample (WPS DescribeProcess)

Process inputs

name* - String

name to return

Karel

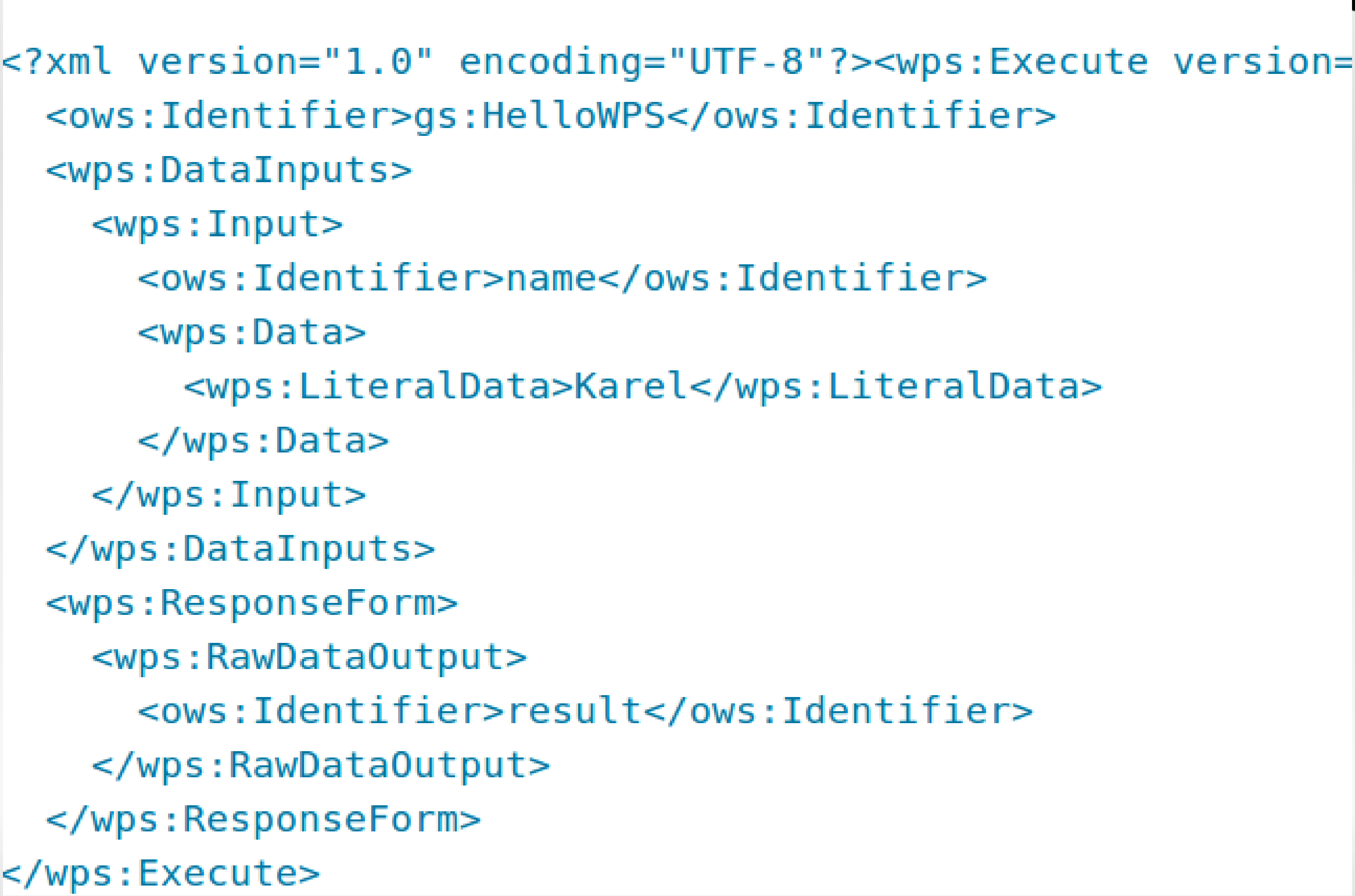
Process outputs

result* - String

output result

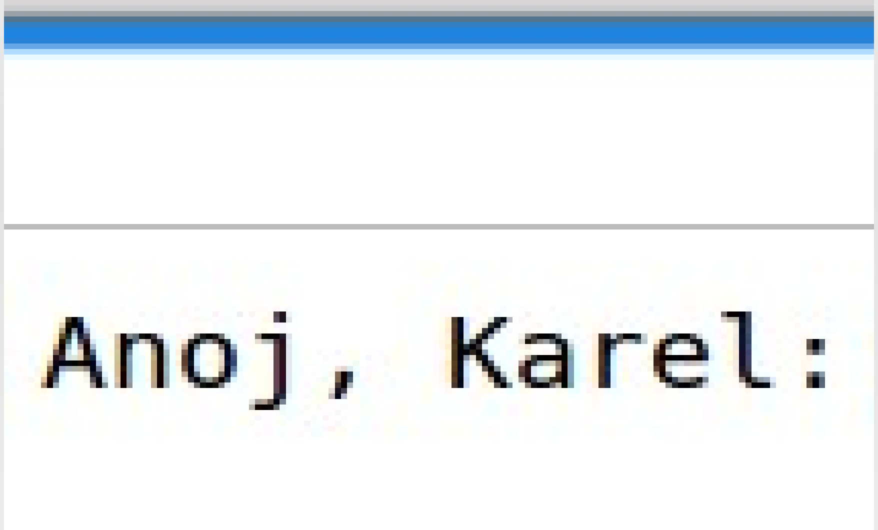
☑ Generate

Testing / 2



```
<?xml version="1.0" encoding="UTF-8"?><wps:Execute version=
  <ows:Identifier>gs:HelloWPS</ows:Identifier>
  <wps:DataInputs>
    <wps:Input>
      <ows:Identifier>name</ows:Identifier>
      <wps:Data>
        <wps:LiteralData>Karel</wps:LiteralData>
      </wps:Data>
    </wps:Input>
  </wps:DataInputs>
  <wps:ResponseForm>
    <wps:RawDataOutput>
      <ows:Identifier>result</ows:Identifier>
    </wps:RawDataOutput>
  </wps:ResponseForm>
</wps:Execute>
```

Testing / 3



Anoj, Karel: