

# Neues aus dem 52° North WPS Projekt

Benjamin Proß, FOSSGIS, 20.03.2014

- Aktuelle Entwicklungen im WPS
- Testing
- WPS 2.0

- Version 3.2.0
- Unterstützt alle Features und Operations der WPS Spezifikation Version 1.0.0
- Unterstützung von:
  - GRASS 7
  - Sextante
  - R
  - Java/Python
- Java Prozess/R Skript Upload via Web UI
- Parser, Generatoren, Validierung von Input Daten

# Aktuelle Entwicklungen im WPS

- Github
- Admin App
- wps-js
- Provenance

# Der 52° North WPS auf GitHub

52n

- <https://github.com/52North/WPS>
- Umgezogen von SVN inklusive Historie
- Fork-und-Pull Muster
- Entwickler haben eigenes WPS-Repository (Fork)
- Spielen Neuerungen/Bugfixes zurück ins 52° North Repository (Pull-Request)
- Pull-Requests werden von Entwicklern getestet/akzeptiert

# Die neue Admin App

- Google Summer of Code 2013
- Allgemeine WPS Konfiguration
  - Service Provider, ...
- 52°North spezifische Konfiguration
  - Parser, Generatoren, Formate, ...
- Test Client
- User Interface und Persistenz Schicht

# Die neue Admin App

52n

The screenshot shows the web interface of the 52° North WPS Admin App. At the top, there is a navigation bar with the following elements: the text '52 North', a 'Home' link, a 'Community' link, a 'Resources' dropdown menu, a 'Username...' input field, a 'Password...' input field, a 'Remember Me' checkbox, and a blue 'Login' button. Below the navigation bar is a sidebar menu on the left with the following sections: 'CONFIGURATIONS' (containing links for Server, Repositories, Generators, and Parsers), 'SETTINGS' (containing links for Users, Log, Service Identification, and Service Provider), 'TESTING' (containing a link for Test Client), and a link for Backup & Restore. The main content area on the right is titled '52° North WPS' and contains three sections: 'Documentation' with two bullet points linking to the OGC website and the 52°North Geoprocessing Community website; 'Administration' with the text 'Please login to access administration pages.'; and 'Examples' with a 'Requests' sub-section containing a bullet point linking to 'GetCapabilities request using HTTP GET', and a 'Test Client' sub-section with the text 'Open the test client of this WPS instance here: 52°North WPS test client.'

- <https://github.com/52North/wps-js>
- JavaScript WPS Client
- Baut auf OpenLayers auf
- Unabhängig von WPS Implementierung



http://geoprocessing.demo.52north.org:8080/wps/WebProcessingService ▾

## Capabilities [Call By Click](#)

Abstract: Service based on the 52° North implementation of WPS 1.0.0

Provider name: 52North

## Execute Process

org.n52.wps.server.algorithm.SimpleBufferAlgorithm ▾

undefined

### Inputs:

**DATA\***   asReference

text/xml; subtype=gml/3.1.1; http://schemas.opengis.net/gml/3.1.1/base/gml.xsd ▾

**WIDTH\***

### Outputs:

**RESULT**

text/xml; subtype=gml/3.1.0; http://schemas.opengis.net/gml/3.1.0/base/feature.xsd ▾

Execute

- OGC Testbed 10
  - Provenance in RDF triples
  - Informationen über Daten und Prozesse
  - Basierend auf einer Ontologie
  - Semantik für den WPS

# RDF Provenance

52n

```
@prefix usgs_data: <http://www.usgs.gov/projects/ows8> .
```

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
```

```
...
```

```
usgs_data:USGSFeature rdfs:subClassOf ows:Feature .
```

```
usgs_data:USGSFeature_CWFID_ST_FIRE_STATION a  
usgs_data:USGSFeature ;
```

```
ows:hadGeometry usgs_data:USGSPosition_10c80 ;
```

```
ows:hadProperty usgs_data:USGS_address_f9acb ;
```

```
ows:hadProperty usgs_data:USGS_name_6adb4 .
```

```
usgs_data:USGS_Position rdfs:subClassOf ows:Point .
```

# RDF Provenance

f2n:52N\_Execution\_8c72d44edf a f2n:52N\_WPSExecution .

f2n:52N\_Execution\_8c72d44edf prov:used  
f2n:Generic\_Process\_v120 .

f2n:52N\_Execution\_8c72d44edf  
prov:startedAtTime 2014-03-18T09:09:17^^xsd:dateTime ;  
prov:endedAtTime 2014-03-18T09:09:17^^xsd:dateTime .

f2n:Generic\_Process\_v120 prov:qualifiedAttribution [  
    a prov:Attribution ;  
    prov:agent f2n:benjamin ;  
    prov:hadRole f2n:Developer ;  
]

- Präsentation auf dem TC Ende März
- Engineering Report
- Weitere Schritte geplant

# 52° North WPS Tests

- Ende 2012 Code Sprint
- Integration Tests
- GET/POST Tests für:
  - GetCapabilities
  - DescribeProcess
  - Execute

# WPS 1.0 Compliance Tests

- OGC Compliance Tests
- Jeder OGC Standard hat eine Abstract Test Suite im Anhang
- Bestehen zeigt, dass ein Webservice dem Standard entspricht (Konformität)
  - Wichtig für Interoperabilität
  - Beispiel WFS 2.0 - Basic WFS
  - Clients wissen genau was sie erwarten können

# WPS 1.0 Compliance Tests

- OGC Compliance Tests für WPS 1.0
  - Seit 2007 noch kein Release
  - Beta r1
    - Getestet, Bugs gefunden
  - Beta r2
    - 52°North WPS besteht alle Tests, ABER...
    - Tests entsprechen nicht dem was im Standard spezifiziert ist
      - U.a. wird keine Prozess-Ausführung (Execute) getestet



# WPS 1.0 Compliance Tests

52n

- Was wird getestet:
  - GET(KVP), POST(XML)
  - GetCapabilities, Sprache, Versionen, Service Parameter
  - DescribeProcess, Sprache, Versionen, Service Parameter, Nicht-Valider Prozess Identifier
  - Keine Execute Tests

# 52° North WPS Tests

- Ende 2012 Code Sprint
- Integration Tests
- GET/POST Tests für:
  - GetCapabilities
  - DescribeProcess
  - Execute

- Schwierigkeiten:
  - Offene Datentypen - Tester wissen nicht genau, was erwartet wird
  - Fehler können durch falsche Datentypen hervorgerufen werden.
  - Woher passende Testdaten nehmen

- Lösungsversuch:
  - Spezifizierung eines Echo-Prozesses
  - Eingabe-werte Literal-/BoundingBox/-Complex-Data
  - Werden jeweils wieder zurückgegeben
  - Dadurch ist testbar:
    - Versteht der WPS Literal-/BoundingBox-/Complex-Data
    - Reagiert der WPS entsprechend auf valide/nicht valide Execute Requests

- Lösungsversuch:
  - Echo-Prozesses auch asynchron ausführbar
  - Dadurch ist asynchrones Verhalten testbar:
    - Status Abfrage
    - Ergebnis Abfrage

# WPS Compliance Tests

52n

- Lösungsvorschlag an WPS 1.0 Testentwickler weitergegeben
- Wird in der WPS 2.0 SWG diskutiert

- Change Requests (CR) für WPS 1.0
- 34 offizielle CRs
  - 20 akzeptiert (einige modifiziert)
  - 3 verschoben
  - 10 abgelehnt
  - 1 zurückgezogen

- Verschachtelte Inputs
- Unterteilung in Kern und Erweiterungen
- Status Kontrolle
- Bessere Definition von WPS Profilen
- Bessere Methode Prozesse synch/asynch auszuführen
- Bessere Test Suite



- Verschachtelte Inputs
  - Input Typen als „Datentyp“ eines Inputs
  - Beziehung von Inputs
  - Erlaubt strukturierte Inputs
  - Ebene der Schachtelung sollte niedrig gehalten werden

```
<ows:Identifier>someprocess</ows:Identifier>
<wps:Input>
  <ows:Identifier>foo1</ows:Identifier>
  <wps:DirectReference xlink:href="http://foodata1"/>
</wps:Input>
<wps:Input>
  <ows:Identifier>foo2</ows:Identifier>
  <wps:Input>
    <ows:Identifier>nested_foo</ows:Identifier>
    <wps:DirectReference xlink:href="http://foodata2"/>
  </wps:Input>
  ...
```

- Unterteilung in Kern und Erweiterungen
  - Bessere Erweiterbarkeit
  - Kern Konzept: WPS Service Model
  - Neue Funktionalität kann eingeführt werden ohne Kern Konzepte/Schemas zu ändern
  - Offen für andere Kommunikations-Konzepte wie REST
  - Offen für andere Formate wie SensorML

- Status Kontrolle
  - Einführung von Methoden zum Pausieren und Fortsetzen von Prozessen

DRAFT

- Bessere Definition von WPS Profilen
  - Klare Definition von Prozess/Service Profilen
  - Profile sind eine Möglichkeit die Interoperabilität zu verbessern

DRAFT

- Bessere Methode Prozesse synchron/asynchron auszuführen
  - Nur ein Boolean Attribut muss gesetzt werden
  - Es kann pro Prozess angegeben werden, ob asynchrone Ausführung unterstützt wird

DRAFT

- OGC TC Ende März
  - Präsentation/Diskussion des aktuellen Stands
  - Roadmap bis zur Veröffentlichung
    - Bis Sommer 2014
  - 52°North WPS 4

Danke für Ihre Aufmerksamkeit!

Mehr Informationen:

<http://52north.org/wps>

b.pross@52north.org