A road map to WCS 2.0 support in iGUESS

Luís de Sousa

03-06-2015

LUXEMBOURG
INSTITUTE
OF SCIENCE
AND TECHNOLOGY

Outline

LUXEMBOURG INSTITUTE OF SCIENCE AND TECHNOLOGY

1. Server side support
2. Client side support
3. Road map

MapServer

LUXEMBOURG INSTITUTE OF SCIENCE AND TECHNOLOGY

- WCS 2.0 supported since version 6.0.
- ***GetCapabilites*** functions in version 6.4.
- ***DescribeCoverage*** fails with "not supported" message
  - Some threads on the web with similar problems;
  - Special configuration needed?
- Version 7.0 has been release in beta - requires panstaking compilation.

GeoServer

LUXEMBOURG INSTITUTE OF SCIENCE AND TECHNOLOGY

- WCS 2.0 supported since version 2.5.
- Gent running version 2.4.3.
- Requests must be tested.
- Easy to upgrade? I.e., can the configuration files be easily migrated?

ArcGIS Server

- WCS 2.0.1 supported only since version 10.3.1 (released already in 2015).
- Easy for our partners to upgrade?
- Requests must be tested:
  - Are the OGC namespaces used correctly?
  - CRS strings reported how?
  - Can more than one CRS be specified?

Client side support

	WCS version		
Library	1.1.0	1.1.1	2.0
OpenLayers 2	✓	✓	×
OpenLayers 3	×	×	×
Leaflet	×	×	×
Google Maps	×	×	×
GeoExt 2	×	×	×
OWSLib	✓	✓	×

Main steps

1. Install testable server instances.
   1. Understand what is going wrong with MapServer 6.4 - compile version 7 if needed.
   2. Install a GeoServer 2.8.x instance.
   3. Obtain an instance of ArcGIS Server 10.3.1.
2. Determine requests formulation that operate with all servers:
   1. ***GetCapabilites***.
   2. ***DescribeCoverage***.
   3. ***GetCoverage***.
3. Add WCS 2.0 support to OWSLib - no alternative foreseable at this stage.
4. Add WCS 2.0 support to one of the web mapping libraries.

Main steps

1. Install testable server instances.
   1. Understand what is going wrong with MapServer 6.4 - compile version 7 if needed.
   2. Install a GeoServer 2.8.x instance.
   3. Obtain an instance of ArcGIS Server 10.3.1.
2. Determine requests formulation that operate with all servers:
   1. ***GetCapabilites***.
   2. ***DescribeCoverage***.
   3. ***GetCoverage***.
3. Add WCS 2.0 support to OWSLib - no alternative foreseable at this stage.
4. Add WCS 2.0 support to one of the web mapping libraries.

Main steps

1. Install testable server instances.
   1. Understand what is going wrong with MapServer 6.4 - compile version 7 if needed.
   2. Install a GeoServer 2.8.x instance.
   3. Obtain an instance of ArcGIS Server 10.3.1.
2. Determine requests formulation that operate with all servers:
   1. ***GetCapabilites***.
   2. ***DescribeCoverage***.
   3. ***GetCoverage***.
3. Add WCS 2.0 support to OWSLib - no alternative foreseable at this stage.
4. Add WCS 2.0 support to one of the web mapping libraries.

Main steps

1. Install testable server instances.
   1. Understand what is going wrong with MapServer 6.4 - compile version 7 if needed.
   2. Install a GeoServer 2.8.x instance.
   3. Obtain an instance of ArcGIS Server 10.3.1.
2. Determine requests formulation that operate with all servers:
   1. ***GetCapabilites***.
   2. ***DescribeCoverage***.
   3. ***GetCoverage***.
3. Add WCS 2.0 support to OWSLib - no alternative foreseable at this stage.
4. Add WCS 2.0 support to one of the web mapping libraries.

Main steps

1. Install testable server instances.
   1. Understand what is going wrong with MapServer 6.4 - compile version 7 if needed.
   2. Install a GeoServer 2.8.x instance.
   3. Obtain an instance of ArcGIS Server 10.3.1.
2. Determine requests formulation that operate with all servers:
   1. ***GetCapabilites***.
   2. ***DescribeCoverage***.
   3. ***GetCoverage***.
3. Add WCS 2.0 support to OWSLib - no alternative foreseable at this stage.
4. Add WCS 2.0 support to one of the web mapping libraries.

Main steps

1. Install testable server instances.
   1. Understand what is going wrong with MapServer 6.4 - compile version 7 if needed.
   2. Install a GeoServer 2.8.x instance.
   3. Obtain an instance of ArcGIS Server 10.3.1.
2. Determine requests formulation that operate with all servers:
   1. ***GetCapabilites***.
   2. ***DescribeCoverage***.
   3. ***GetCoverage***.
3. Add WCS 2.0 support to OWSLib - no alternative foreseable at this stage.
4. Add WCS 2.0 support to one of the web mapping libraries.

Main steps

1. Install testable server instances.
   1. Understand what is going wrong with MapServer 6.4 - compile version 7 if needed.
   2. Install a GeoServer 2.8.x instance.
   3. Obtain an instance of ArcGIS Server 10.3.1.
2. Determine requests formulation that operate with all servers:
   1. ***GetCapabilites***.
   2. ***DescribeCoverage***.
   3. ***GetCoverage***.
3. Add WCS 2.0 support to OWSLib - no alternative foreseable at this stage.
4. Add WCS 2.0 support to one of the web mapping libraries.

WCS 2.0 support - web client

LUXEMBOURG INSTITUTE OF SCIENCE AND TECHNOLOGY

- The *easy* way - expand the existing OpenLayers 2 API
  - definitly locks iGUESS to a rapidly aging library;
  - the WCS 1.1.1 API has not been merged yet.
- The *hard* way - code a WCS API from scratch on OpenLayers 3:
  - no real base to start on;
  - even WFS support is minimal - OL 3 focusing on WMS.
- The *reasonable* way - code a WCS API on GeoExt 2:
  - detach interaction with data services from WMS - API remains functional if/when GeoExt moves on to OL 3;
  - existing WFS API can be a good starting base;
  - Chris worked on it (could not find the code).

WCS 2.0 support - web client

- The *easy* way - expand the existing OpenLayers 2 API
  - definitly locks iGUESS to a rapidly aging library;
  - the WCS 1.1.1 API has not been merged yet.
- The *hard* way - code a WCS API from scratch on OpenLayers 3:
  - no real base to start on;
  - even WFS support is minimal - OL 3 focusing on WMS.
- The *reasonable* way - code a WCS API on GeoExt 2:
  - detach interaction with data services from WMS - API remains functional if/when GeoExt moves on to OL 3;
  - existing WFS API can be a good starting base;
  - Chris worked on it (could not find the code).

WCS 2.0 support - web client

LUXEMBOURG INSTITUTE OF SCIENCE AND TECHNOLOGY

- The *easy* way - expand the existing OpenLayers 2 API
  - definitly locks iGUESS to a rapidly aging library;
  - the WCS 1.1.1 API has not been merged yet.
- The *hard* way - code a WCS API from scratch on OpenLayers 3:
  - no real base to start on;
  - even WFS support is minimal - OL 3 focusing on WMS.
- The *reasonable* way - code a WCS API on GeoExt 2:
  - detach interaction with data services from WMS - API remains functional if/when GeoExt moves on to OL 3;
  - existing WFS API can be a good starting base;
  - Chris worked on it (could not find the code).

WCS 2.0 support - web client

- The *easy* way - expand the existing OpenLayers 2 API
  - definitly locks iGUESS to a rapidly aging library;
  - the WCS 1.1.1 API has not been merged yet.
- The *hard* way - code a WCS API from scratch on OpenLayers 3:
  - no real base to start on;
  - even WFS support is minimal - OL 3 focusing on WMS.
- The *reasonable* way - code a WCS API on GeoExt 2:
  - detach interaction with data services from WMS - API remains functional if/when GeoExt moves on to OL 3;
  - existing WFS API can be a good starting base;
  - Chris worked on it (could not find the code).

WCS 2.0 support - web client

LUXEMBOURG INSTITUTE OF SCIENCE AND TECHNOLOGY

- The *easy* way - expand the existing OpenLayers 2 API
  - definitly locks iGUESS to a rapidly aging library;
  - the WCS 1.1.1 API has not been merged yet.
- The *hard* way - code a WCS API from scratch on OpenLayers 3:
  - no real base to start on;
  - even WFS support is minimal - OL 3 focusing on WMS.
- The *reasonable* way - code a WCS API on GeoExt 2:
  - detach interaction with data services from WMS - API remains functional if/when GeoExt moves on to OL 3;
  - existing WFS API can be a good starting base;
  - Chris worked on it (could not find the code).

WCS 2.0 support - web client

- The *easy* way - expand the existing OpenLayers 2 API
  - definitly locks iGUESS to a rapidly aging library;
  - the WCS 1.1.1 API has not been merged yet.
- The *hard* way - code a WCS API from scratch on OpenLayers 3:
  - no real base to start on;
  - even WFS support is minimal - OL 3 focusing on WMS.
- The *reasonable* way - code a WCS API on GeoExt 2:
  - detach interaction with data services from WMS - API remains functional if/when GeoExt moves on to OL 3;
  - existing WFS API can be a good starting base;
  - Chris worked on it (could not find the code).

Recap

1. Install testable server instances.
   1. Understand what is going wrong with MapServer 6.4 - compile version 7 if needed.
   2. Install a GeoServer 2.8.x instance.
   3. Obtain an instance of ArcGIS Server 10.3.1.
2. Determine requests formulation that operate with all servers:
   1. ***GetCapabilites***.
   2. ***DescribeCoverage***.
   3. ***GetCoverage***.
3. Add WCS 2.0 support to OWSLib - no alternative foreseable at this stage.
4. Add WCS 2.0 support to GeoExt 2.