



## **Useful extensions to the OGC Web Processing Service based on a Python client/server implementation**

Ag Stephens, Stephen Pascoe, and Philip Kershaw  
Science & Technology Facilities Council, Didcot, Oxon, United Kingdom

Web Processing Service (WPS) is an Open Geospatial Consortium (OGC) standard that is still in its infancy. It provides a generic framework for implementing Web Services that cannot be delivered by the commonly adopted Web Mapping Service (WMS), Coverage Service (WCS) and Feature Service (WFS). With support for asynchronous processes the WPS also provides a container for managing offline processes, scheduling and service-chaining when handling large requests lasting minutes, hours or even days.

Already deployed operationally behind a high-availability UK government projections website, the CEDA OGC Web Services WPS (COWS-WPS) is an emerging Python based framework built on a host of open source tools to allow fast prototyping and deployment of Web Services as WPS “processes”. New processes can be added by the simple addition of a configuration file, describing inputs and outputs, and a Python wrapper module that interfaces the processing code with the WPS framework.

Each process is secured by a set of authorisation roles using the Security middleware developed at CEDA. This enables service calls to be intercepted and checked against an XML-based authorisation policy. Dual OpenID and PKI based authentication schemes are supported..

The most recent COWS-WPS development has been a web-client that is currently tightly-coupled to the server implementation. The client auto-generates a web-form for each process deployed under the WPS, this includes an interactive map for bounding box selection and other typed selections such as date/time. By extending the list of known types beyond that of listed in the WPS specification a rich client application can be produced. For general development of Web Services this provides rapid deployment of web-clients without the need to produce bespoke interfaces for each new service.

Experimentation with the client-server interactions has led to an understanding of certain limitations of the WPS 1.0.0 specification. Service-chaining with the OGC WCS has been investigated with the WPS acting as an asynchronous proxy to the WCS. In this configuration, the WCS parameters reflected in the GetCapabilities and DescribeCoverage responses are made visible to the WPS Process Descriptions (returned from requests to the DescribeProcess method) by the use of “dynamic parameters”. These dynamic parameters are not available to the WPS client until certain service calls have been made. This means the WPS client must be capable of calling out to services in order to populate the form options made available to the user.

If WPS is to realise its potential in the OGC service landscape then this approach is considered an essential extension to the existing standard. Whilst there are many ways it could be implemented the support for “dynamic parameters” in the DescribeProcess response is a clear and relatively small change to the existing specification.