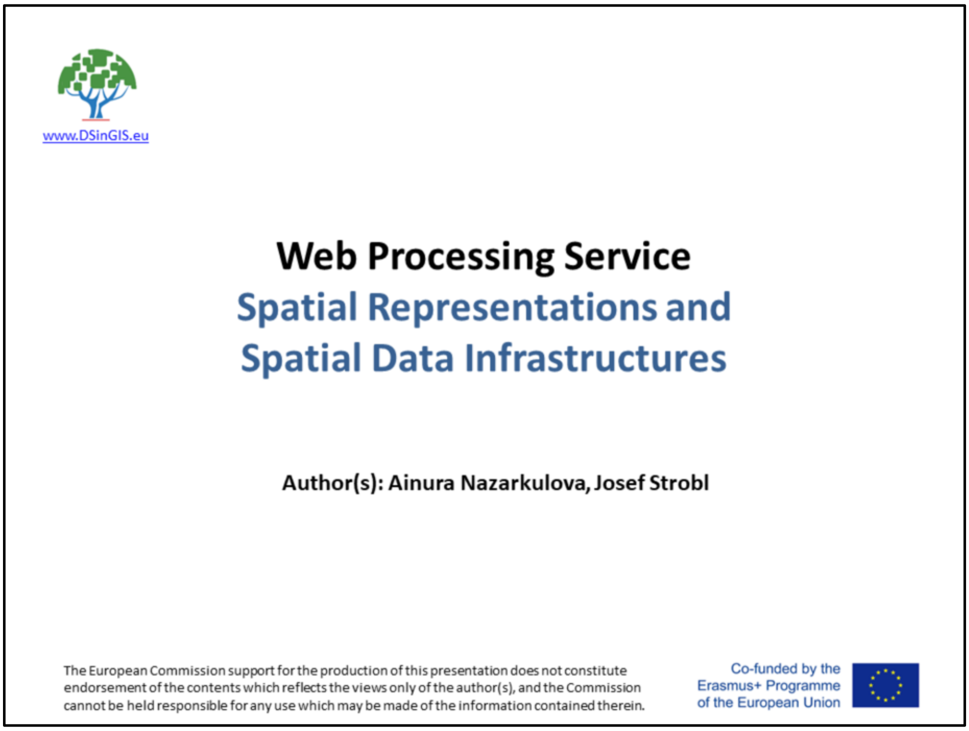


www.DSinGIS.eu

Web Processing Service
Spatial Representations and Spatial Data Infrastructures

Author(s): Ainura Nazarkulova, Josef Strobl

The European Commission support for the production of this presentation does not constitute endorsement of the contents which reflects the views only of the author(s), and the Commission cannot be held responsible for any use which may be made of the information contained therein.

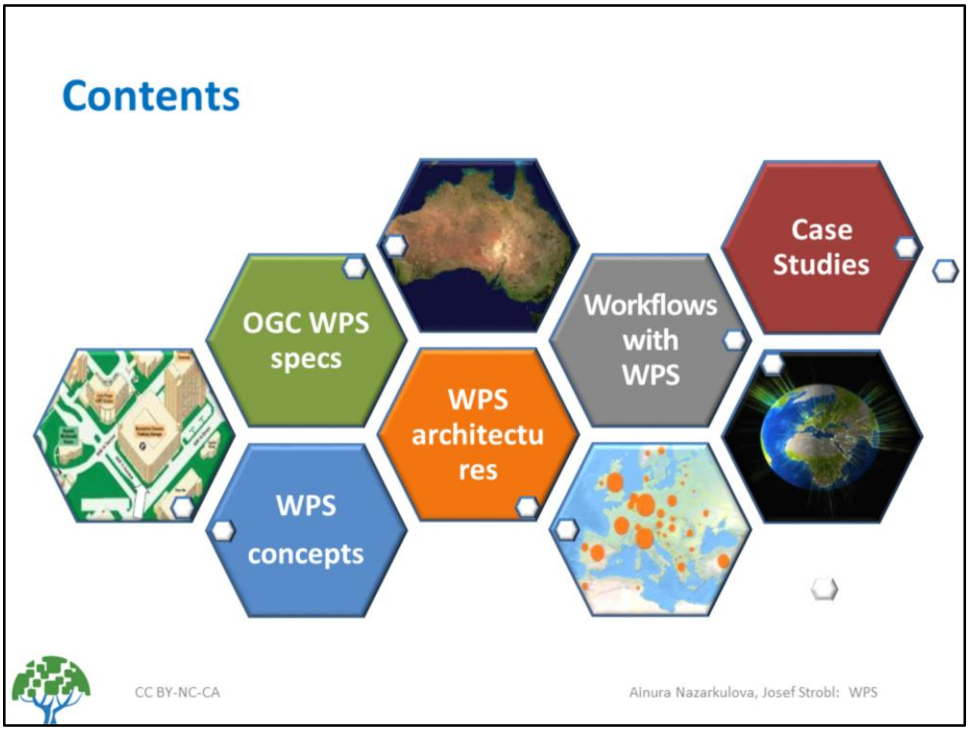
Co-funded by the Erasmus+ Programme of the European Union

The main purpose of many Geographic Information System implementations is not only to serve as a system-of-records (ie, to hold and manage data, and to provide access), but rather to help creating added value by extracting information from data.

Spatial analysis operations serve this purpose, by applying spatial concepts like ‚distance‘ or ‚overlap‘ to spatial data. Traditionally, spatial analysis required the installation of software on a user‘s computer, with all related requirements of performance, data storage, versioning, licensing etc.

Within the framework of distributed SDI it is obvious, that the task of geospatial analysis should be handled through dedicated services as well, allowing users to complete analytical processing in a cloud-based environment accessed from a thin client, like a web browser.

See: https://en.wikipedia.org/wiki/Web_Processing_Service



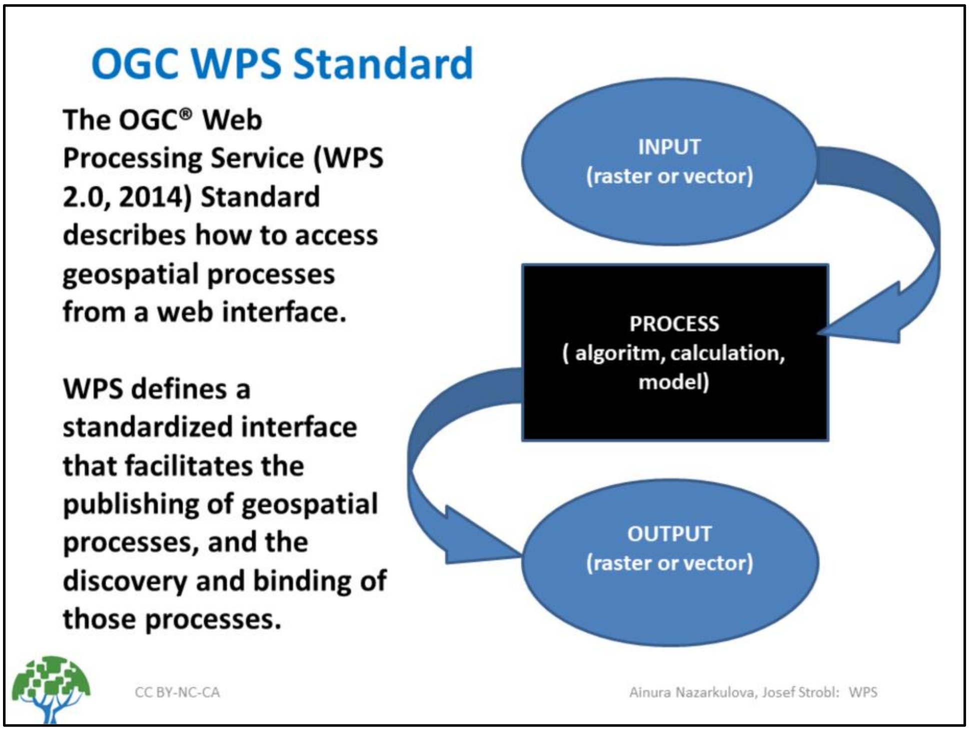
Contents

Ainura Nazarkulova, Josef Strobl: WPS

This lecture starts with introducing and discussing WPS concepts and specifications by WPS.

It then continues with exploring architectural details of client-server communication and the flow of requests and data, explaining some of the challenges of WPS as a very generic standard needing a lot of work from implementors.

Finally, several practical applications demonstrate the value of distributed geospatial processing in several application domains.



OGC WPS Standard

The OGC® Web Processing Service (WPS 2.0, 2014) Standard describes how to access geospatial processes from a web interface.

WPS defines a standardized interface that facilitates the publishing of geospatial processes, and the discovery and binding of those processes.

INPUT
(raster or vector)

PROCESS
(algoritm, calculation, model)

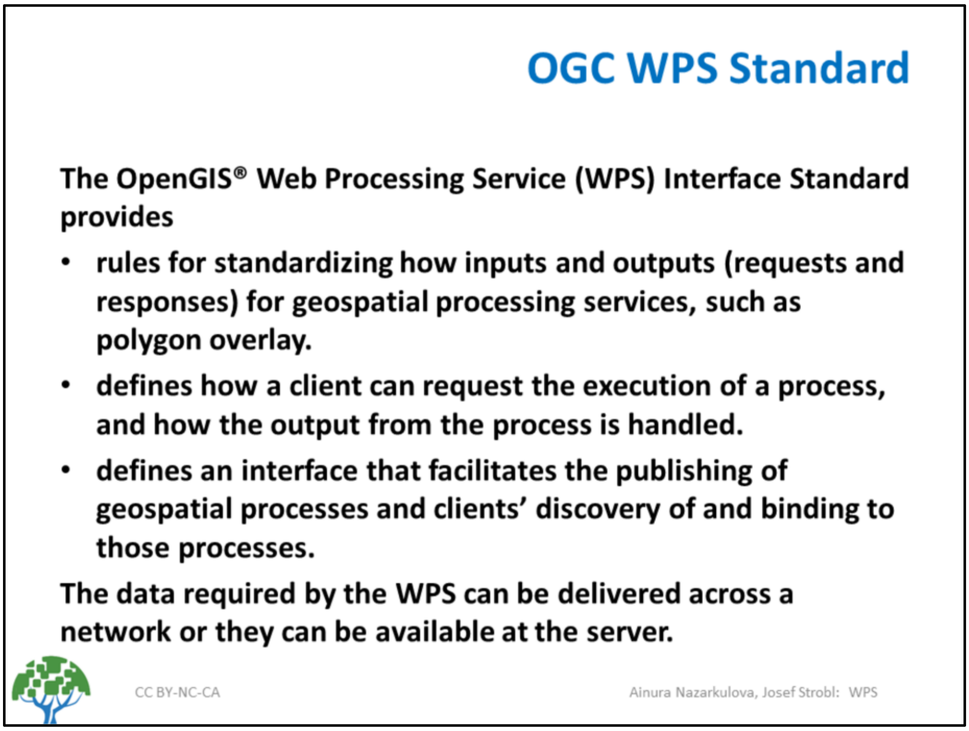
OUTPUT
(raster or vector)

CC BY-NC-CA

Ainura Nazarkulova, Josef Strobl: WPS

Spatial analysis is the core of any GIS and Web GIS too. The basic analysis tools such as measurements, proximity, map overlay, etc. are available nowadays with many commercial and open source web applications (Fu and Sun , 2010). Such tools are used in daily life for various tasks like finding the optimum driving path, using proximity to find nearby bus stop.

The OGC® Web Processing Service (WPS) Standard describes how to access geospatial processes from a web interface, where process is a function that for each input returns a corresponding output.
Processes include any algorithm, calculation or model that operates on spatially referenced vector or raster data. A WPS may offer calculations as simple as subtracting one set of spatially referenced numbers from another (e.g., determining the difference in influenza cases between two different seasons), or as complicated as a global climate change model.
Development of WPS specifications has been done within the OGC, with currently valid documents accessible via http://www.opengeospatial.org/standards/wps



OGC WPS Standard

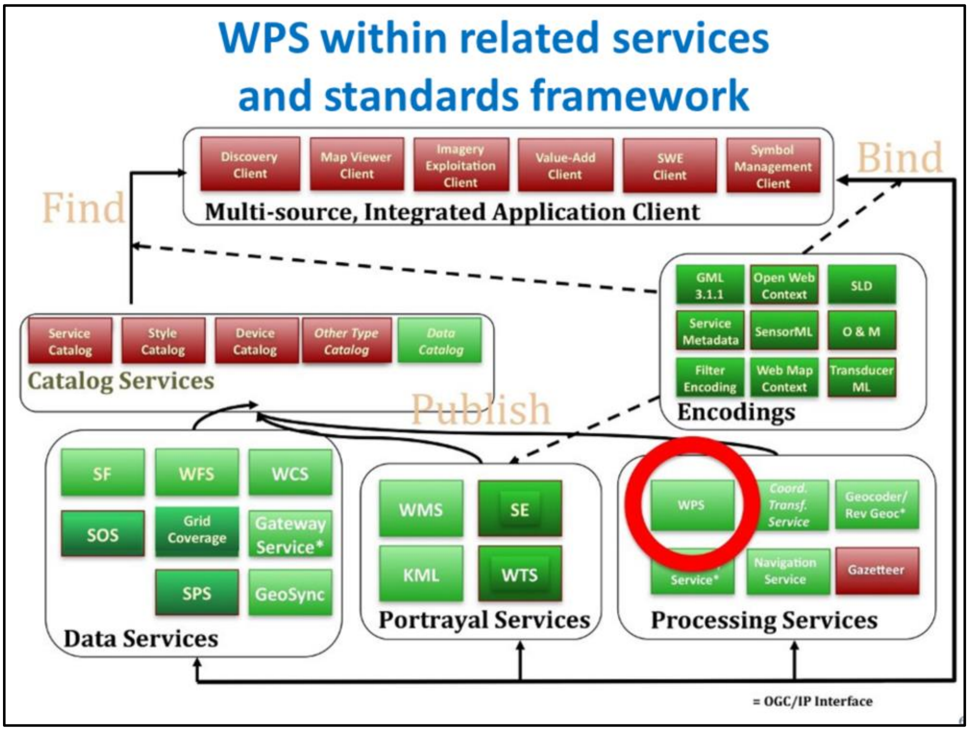
The OpenGIS® Web Processing Service (WPS) Interface Standard provides

- rules for standardizing how inputs and outputs (requests and responses) for geospatial processing services, such as polygon overlay.
- defines how a client can request the execution of a process, and how the output from the process is handled.
- defines an interface that facilitates the publishing of geospatial processes and clients' discovery of and binding to those processes.

The data required by the WPS can be delivered across a network or they can be available at the server.

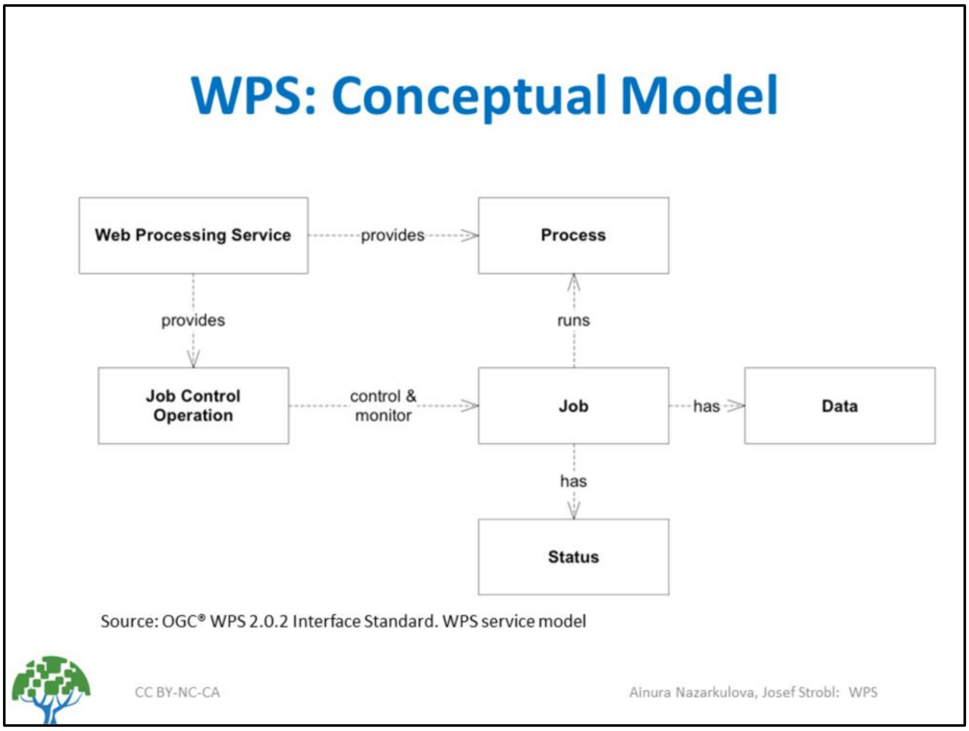
The OpenGIS® Web Processing Service (WPS) Interface Standard provides rules for standardizing inputs and outputs (requests and responses) for geospatial processing services, such as polygon overlays and their inputs and outputs (https://live.osgeo.org/en/standards/wps_overview.html)
WPS is a generic interface in that it does not identify any specific processes that are supported. Instead, each implementation of WPS defines the processes that it supports, as well as their associated inputs and outputs. WPS can be thought of as an abstract model of a web service, for which profiles need to be developed to support use, and standardized to support interoperability.
The widely used WPS standard (OGC, 2007) describes a service interface and communication protocol for processing data in a Web service environment. WPS clients and servers provide and invoke a broad range of geoprocessing functions on the Web.

www.DSinGIS.eu



WPS within related services and standards framework

Explore and discuss the position of processing services within a broader setting of open web services underpinning an SDI architecture. Check how these services might inter-relate and interoperate. Which ones from these services and encodings can you identify and briefly describe?
As with the other OGC specifications GML and CAT, it is the development, publication, and adoption of profiles which define the specific uses of this specification.

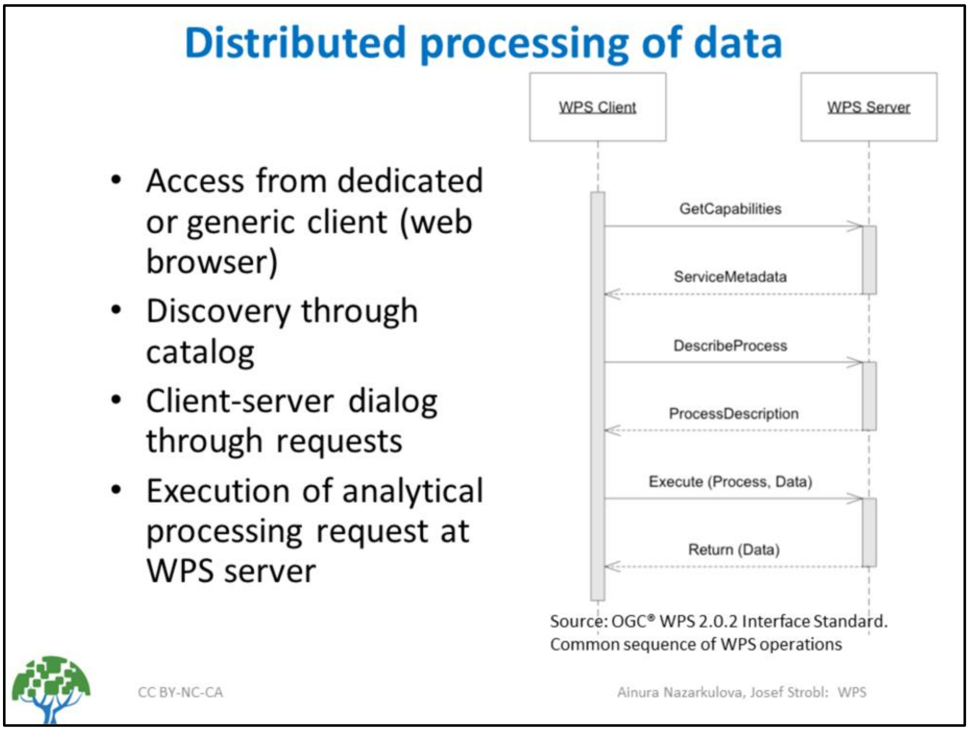


WPS: Conceptual Model

Source: OGC® WPS 2.0.2 Interface Standard. WPS service model

The WPS service model defines basic properties of any WPS server. A WPS server is a web service that provides access to pre-defined processes and provides job control operations to instantiate, control and monitor processing jobs.
The widely used WPS standard (OGC, 2007) describes a service interface and communication protocol for processing data in a Web service environment. WPS clients and servers provide and invoke a broad range of geoprocessing functions on the Web.
Geoprocessing functions in WPS are specified by process descriptions (DescribeProcess) which cover syntactic properties of a function's arguments (inputs and outputs), the supported data schemas, and encoding formats. Simple human-readable free-text elements for titles and abstracts may be used to supply basic semantic descriptions about the provided functionality.

www.DSinGIS.eu



Distributed processing of data

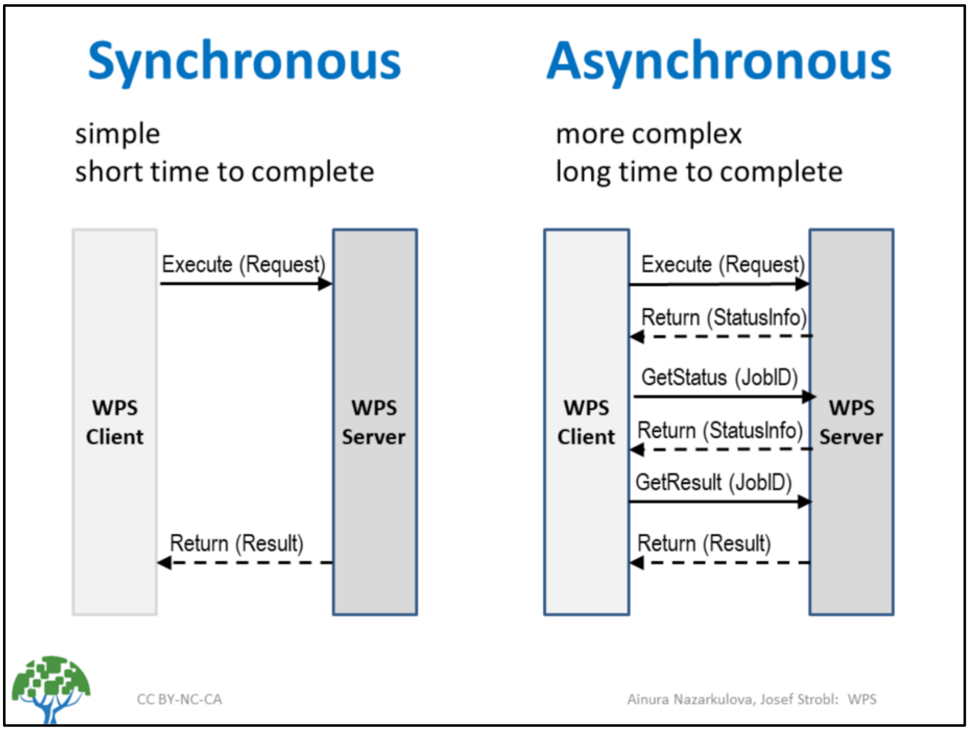
- Access from dedicated or generic client (web browser)
- Discovery through catalog
- Client-server dialog through requests
- Execution of analytical processing request at WPS server

Source: OGC® WPS 2.0.2 Interface Standard. Common sequence of WPS operations

CC BY-NC-CA

Ainura Nazarkulova, Josef Strobl: WPS

The general idea and concept of WPS is simple: the client sends a request to a known (a priori, or from a catalog discovery) service endpoint at a WPS server, essentially asking ‘*what can you do for me?’* – this is the GetCapabilities request.
Geoprocessing functions in WPS are specified by process descriptions (DescribeProcess) which cover syntactic properties of a function’s arguments (inputs and outputs), the supported data schemas, and encoding formats. Simple human-readable free-text elements for titles and abstracts may be used to supply basic semantic descriptions about the provided functionality.
Then the desired processing tasks is described, ‘fed’ with data directly from the client request or a WFS source, and execution is requested.
Again the resulting output data are directly returned to the client or handled via WFS.



Synchronous

simple
short time to complete

WPS Client

Execute (Request)

WPS Server

Return (Result)

Asynchronous

more complex
long time to complete

WPS Client

Execute (Request)

Return (StatusInfo)

GetStatus (JobID)

Return (StatusInfo)

GetResult (JobID)

Return (Result)

WPS Server

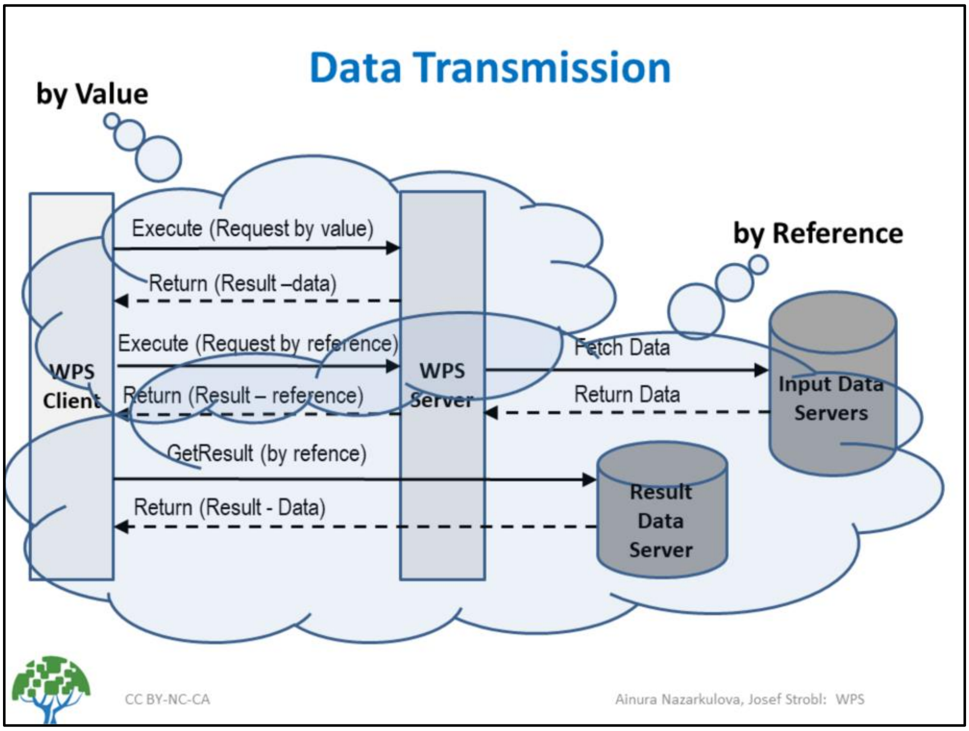
CC BY-NC-CA

Ainura Nazarkulova, Josef Strobl: WPS

WPS services and their accompanying tasks are either synchronous or asynchronous. Synchronous means that client applications will wait while the task executes on the server. Synchronous execution is a suitable approach for simple jobs that take a relatively short time to complete. In the synchronous case, a WPS client submits an execute request to the WPS server and keeps listening for a response until the processing job has completed and the processing result has been returned. This requires a persistent connection between client and server.

Asynchronous means that client applications don't have to wait for the task to finish on the server before moving on to other tasks. Asynchronous execution is preferable for jobs that may take a long time to complete. It is more complex. In this case, the client sends an execute request to the WPS server and immediately receives a status information response. This information confirms that the request was received and accepted by the server and that a processing job has been created and will be run in the future. The status information response also contains a processing job identifier that is used by the client when checking to see if execution has completed. In addition, the status information response contains the result location, i.e. the URL where the processing result can be found after the processing job is completed.

For both execution modes, task results can be stored on the server and referenced through a URL, or they can be streamed directly back to clients.



Data Transmission

by Value

Execute (Request by value)

Return (Result –data)

Execute (Request by reference)

Return (Result – reference)

GetResult (by refence)

Return (Result - Data)

WPS Client

WPS Server

by Reference

Fetch Data

Return Data

Input Data Servers

Result Data Server

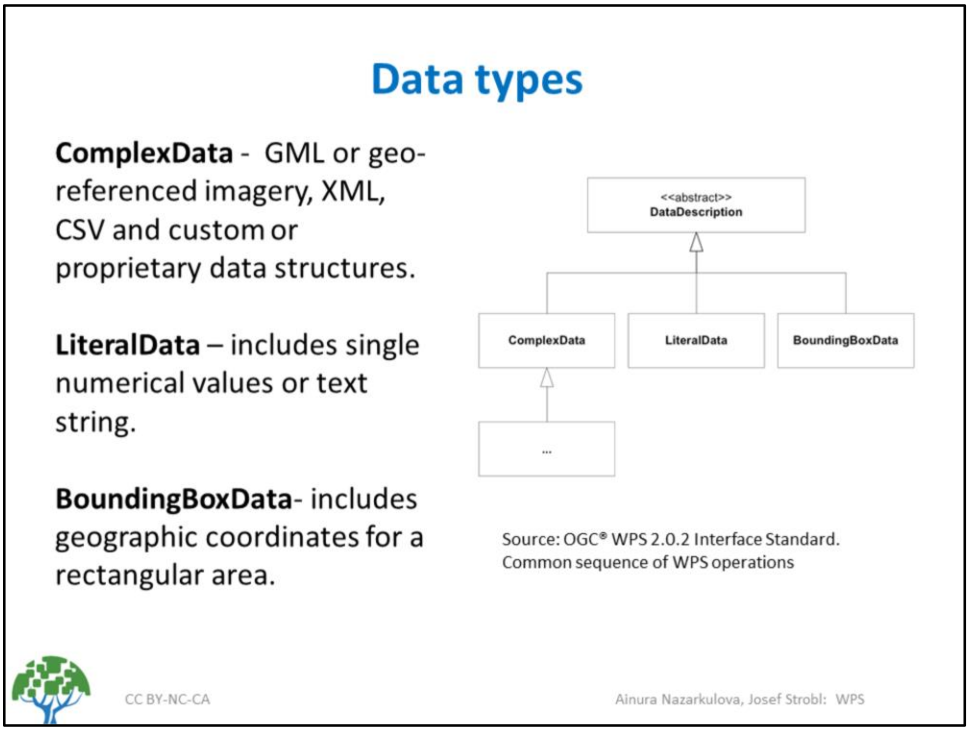
CC BY-NC-CA

Ainura Nazarkulova, Josef Strobl: WPS

WPS allows for the provision of input data in two different methods. Data can either be embedded in the Execute request (by value), or referenced as a web accessible resource (by reference).

In the former approach, WPS acts as a stand-alone service. In the latter fashion, WPS acts as middleware service for data, by obtaining data from an external resource in order to run a process on the local implementation.

The picture shows the transmission patterns in the pure form. However, mixed patterns are possible. Typically, small or atomic data such as integers, doubles or short strings are submitted by value. Large data inputs (outputs) are usually supplied by reference.
WPS allows existing software interfaces to be wrapped up and presented to the network as web services. Implementations of WPS can thus be considered middleware for software.



Data types

ComplexData - GML or geo-referenced imagery, XML, CSV and custom or proprietary data structures.

LiteralData – includes single numerical values or text string.

BoundingBoxData- includes geographic coordinates for a rectangular area.

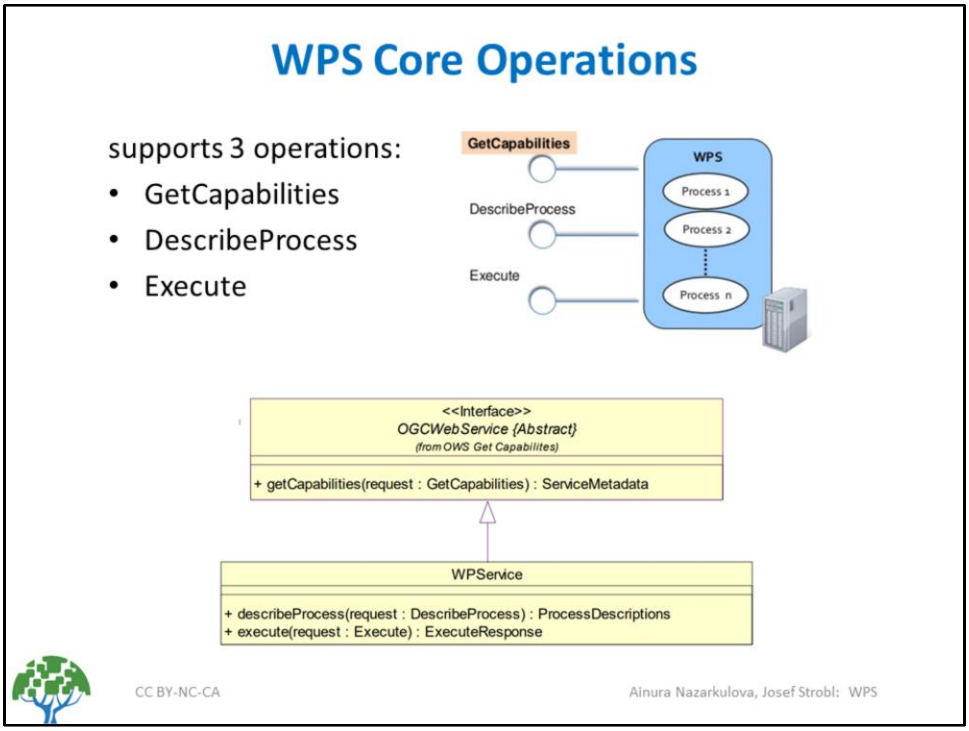
<<abstract>>
DataDescription
ComplexData
LiteralData
BoundingBoxData
...

Source: OGC® WPS 2.0.2 Interface Standard. Common sequence of WPS operations

Ainura Nazarkulova, Josef Strobl: WPS

WPS specification defines three common data types for process input and output data:

- Complex Data, such as GML or geo-referenced imagery. This type is kept generic regarding the content and may be extended to provide more detailed domain-specific information.
- LiteralData, defined as a value with an optional unit. It encodes atomic data such as scalars, linear units, or well-known names. Domains for LiteralData are a combination of data types (e.g. Double, Integer, String), a given value range, and an associated unit (e.g. meters, degrees Celsius).
- BoundingBoxData, defined as a minimum bounding rectangle in geographic coordinates. It serves a variety of purposes in spatial data processing. Some simple applications are the definition of extents for a clipping operation or the definition of an analysis region. The domain for bounding box data is described by a listing of supported Coordinate Refenence Systems.



WPS Core Operations

supports 3 operations:

- GetCapabilities
- DescribeProcess
- Execute

<<Interface>>
OGCWebService {Abstract}
(from OWS Get Capabilites)
+ getCapabilities(request : GetCapabilities) : ServiceMetadata

WPService
+ describeProcess(request : DescribeProcess) : ProcessDescriptions
+ execute(request : Execute) : ExecuteResponse

Ainura Nazarkulova, Josef Strobl: WPS

The WPS service model specifies the following operations that may be invoked by a WPS client and performed by a WPS server:

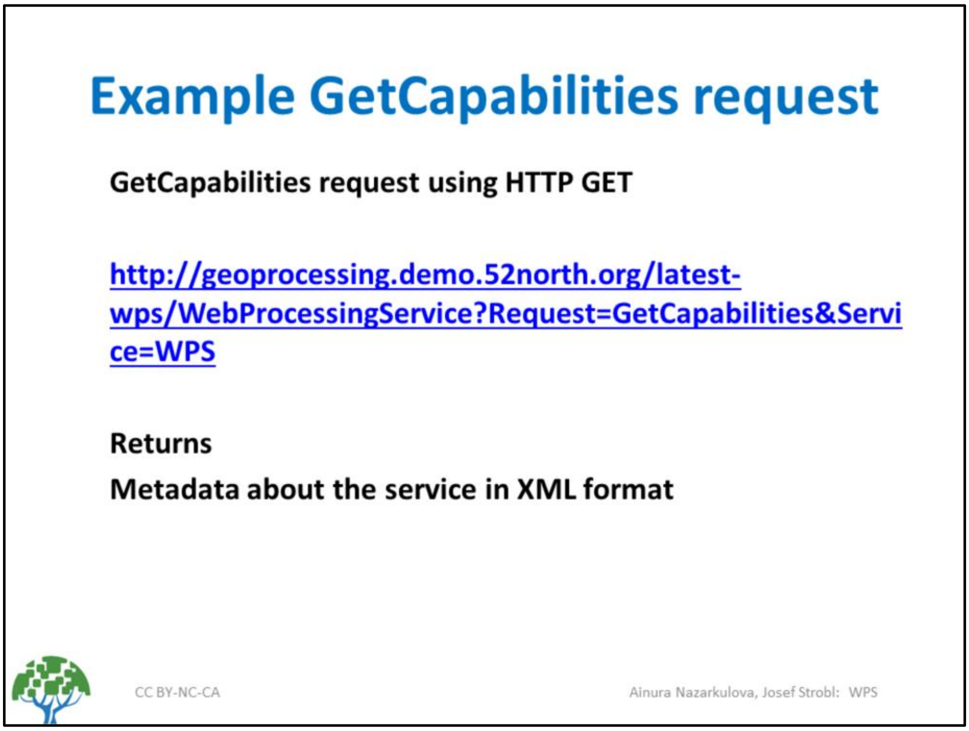
GetCapabilities – This operation allows a client to request information about the server’s capabilities and processes offered and returns service-level metadata (what the service can do)

DescribeProcess – This operation allows a client to request detailed metadata on selected processes offered by a server . Result is a description of the process along with its inputs & outputs

Execute – given the required input, this operation will result in the provision of the described output

GetStatus – This operation is optional and allows a client to query status information of a processing job

GetResult – This is also optional operation which allows a client to query the results of a processing job



Example GetCapabilities request

GetCapabilities request using HTTP GET

http://geoprocessing.demo.52north.org/latest-wps/WebProcessingService?Request=GetCapabilities&Service=WPS

Returns

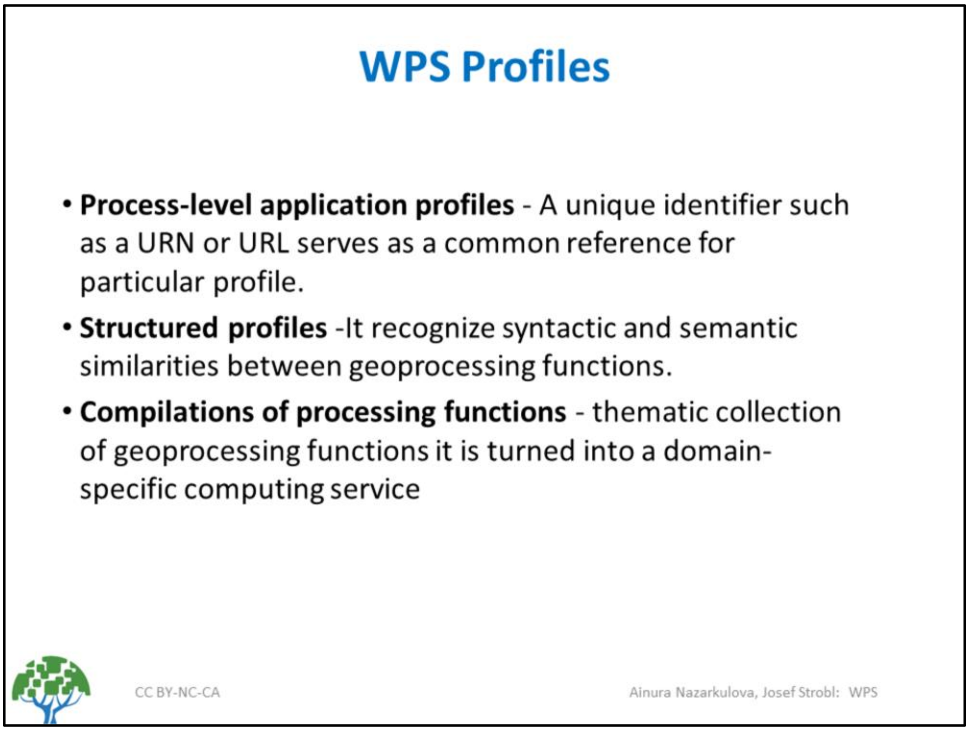
Metadata about the service in XML format

CC BY-NC-CA

Ainura Nazarkulova, Josef Strobl: WPS

In the GetCapabilities request, the server answers with the metadata of the OGC WPS service. This metadata describes the abilities of the requested server.
The GetCapabilities operation returns the names and descriptions of the different pre-programmed processes supported by the specific WPS server.
GetCapabilities operation returns an XML document that gives the information required to consume the service programmatically and automatically. The capability document contains the following information:

- ServiceIdentification & ServiceProvider information: describes the title of the service and the name (plus additional contact information) of the person who published the service.
- OperationMetadata: documents the operations supported by the service: GetCapabilities, DescribeProcess, Execute.
- ProcessOfferings: gives information about the spatial analysis task that is served out as WPS service. In the image displayed above, the “spline” interpolation method has been published as WPS.
- WSDL (Web Service Description Language): via this URL, the service can be used in other distributed platforms that supports WSDL. For instance, the WSDL can be used by an orchestration like Business Process Execution Language (BPEL) to couple different geoprocessing services together.



WPS Profiles

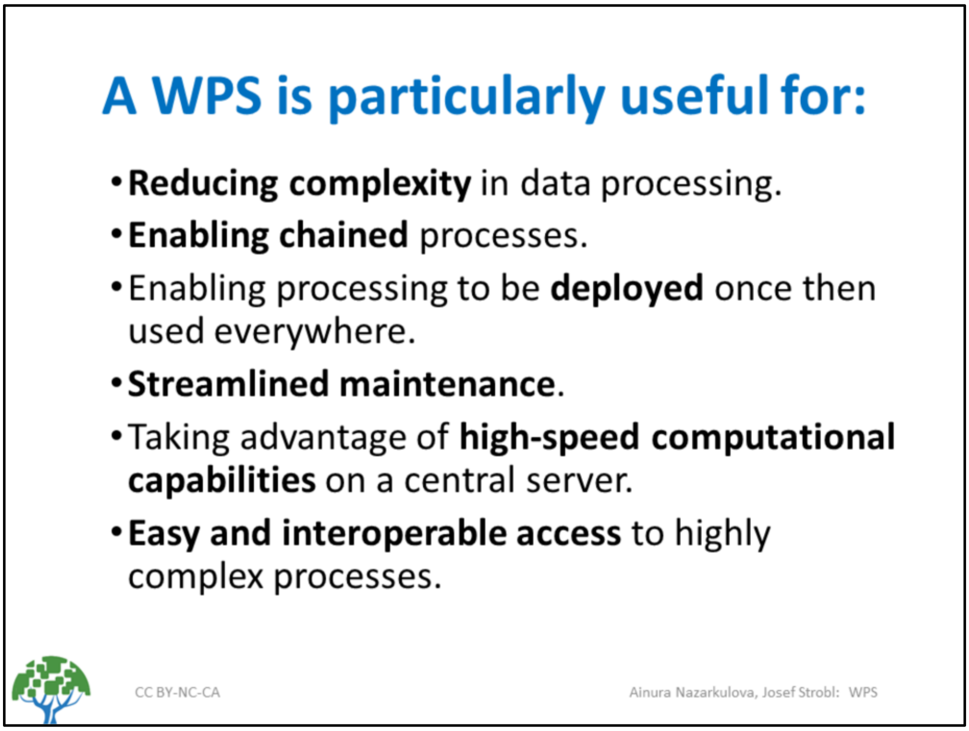
- **Process-level application profiles** - A unique identifier such as a URN or URL serves as a common reference for particular profile.
- **Structured profiles** -It recognize syntactic and semantic similarities between geoprocessing functions.
- **Compilations of processing functions** - thematic collection of geoprocessing functions it is turned into a domain-specific computing service

Ainura Nazarkulova, Josef Strobl: WPS

In order to support typical publish–find–bind scenarios, process profiles have been proposed for the WPS 2.0 specification as a means to contract and possibly standardize the content of WPS servers. Profiles and compliant services might be cataloged in public Web based repositories and potential clients would simply browse these repositories to find processing services that implement the desired functionality. These can be roughly divided into three categories.

- *Process-level application profiles* - contracts the process definition, process inputs and outputs and their concrete data encodings formats. In this regard, each process-level application profile exists in complete isolation.
- *Structured profiles* introduce an inheritance mechanism between individual profiles. Commonalities between similar functions, such as identical input and output parameters or common data formats are collected in base classes that can be extended to obtain more specific profiles.
- *Compilations of processing functions* - share some aspects with thematic GIS toolboxes, e.g. for hydrology, terrain analysis, or spatial multi-criteria evaluation.

Compared to structured profiles, compilations of processing functions are less driven by structural similarity at an algorithmic or syntactic level. They rather recognize similarities between geoprocessing functions from the perspective of a particular application domain. While the creation of thematic collections is extremely useful for specialized functions that clearly belong to a single domain, the approach provides little help for structuring and comparing more general functions which are used across multiple domains.



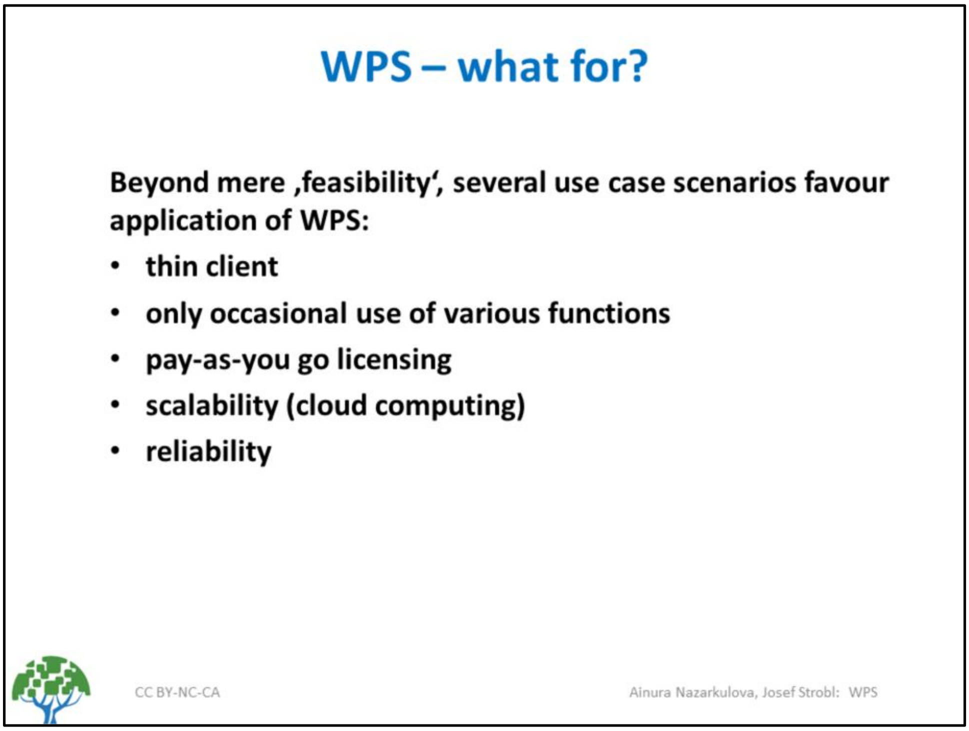
A WPS is particularly useful for:

- **Reducing complexity** in data processing.
- **Enabling chained** processes.
- Enabling processing to be **deployed** once then used everywhere.
- **Streamlined maintenance**.
- Taking advantage of **high-speed computational capabilities** on a central server.
- **Easy and interoperable access** to highly complex processes.

Ainura Nazarkulova, Josef Strobl: WPS

A WPS is particularly useful for:

- Reducing complexity in data processing by providing plug & play algorithms.
- Enabling processes to be chained together.
- Enabling processing to be deployed once then used everywhere.
- Streamlined maintenance. Processes/models are maintained in a central place by the entities who created them.
- Taking advantage of high-speed computational capabilities on a central server.
- Easy and interoperable access to highly complex processes, such as climate change models.



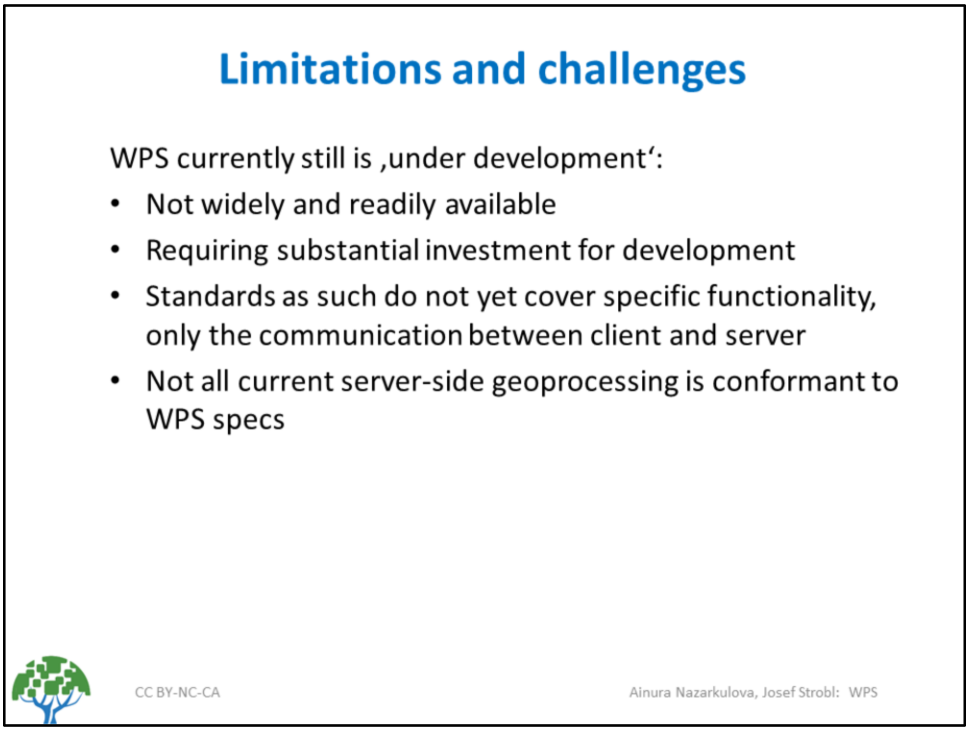
WPS – what for?

Beyond mere ‚feasibility', several use case scenarios favour application of WPS:

- thin client
- only occasional use of various functions
- pay-as-you go licensing
- scalability (cloud computing)
- reliability

Ainura Nazarkulova, Josef Strobl: WPS

The main advantage of WPS id interoperability of network enabled data processing. It allows organizations to deliver calculations to users independent of the underlying software.
It supports multiple web services approaches. It defines equivalent KVP Get, XML Post, and SOAP interfaces, allowing the user to choose the most appropriate interface.
Exposing processes through WPS allows organistations to reuse the same process.
WPS exploits the power of distributed /cloud computing. It is designed to enable distributed processing of geospatial data located anywhere on the Internet.
It gives fast, reliable access to “near real time” calculations.

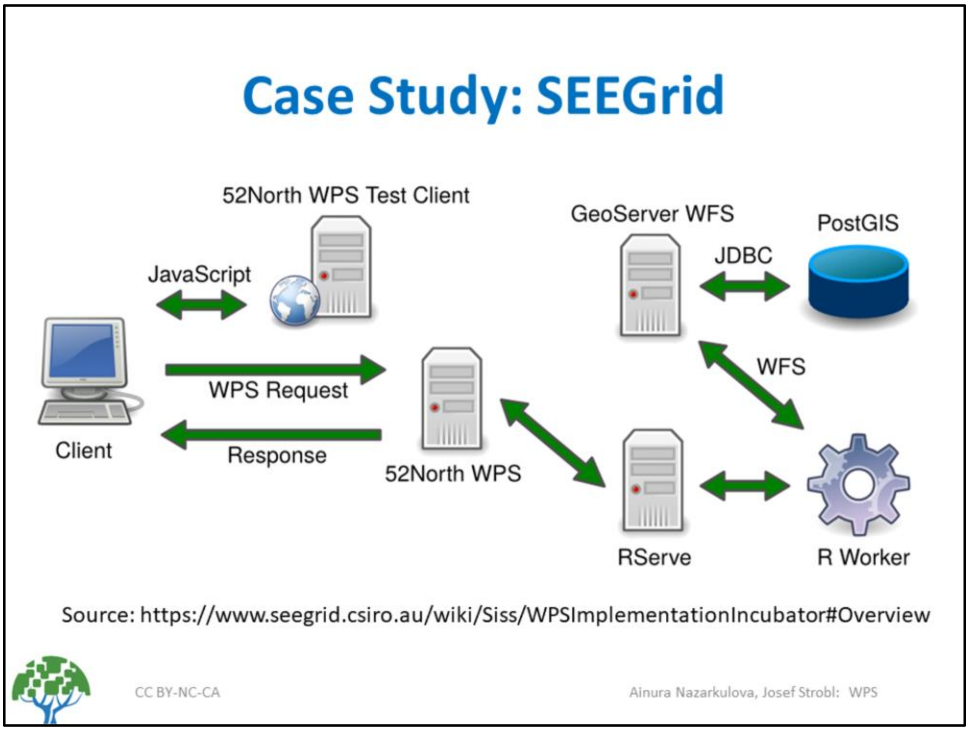


Limitations and challenges

WPS currently still is ‚under development‘:

- Not widely and readily available
- Requiring substantial investment for development
- Standards as such do not yet cover specific functionality, only the communication between client and server
- Not all current server-side geoprocessing is conformant to WPS specs

Actual generic WPS implementation is less simple than this sounds. WPS still is one of the more challenging and less developed services within the OGC portfolio.
WPS currently still is ‚under development‘:
Web proccesing services are not widely and readily available as WMS, WFS and WCS.
It requiressubstantial investment for development due to the generic specification.
Standards as such do not yet cover specific functionality, only the communication between client and server
Not all current server-side geoprocessing is conformant to WPS specifications



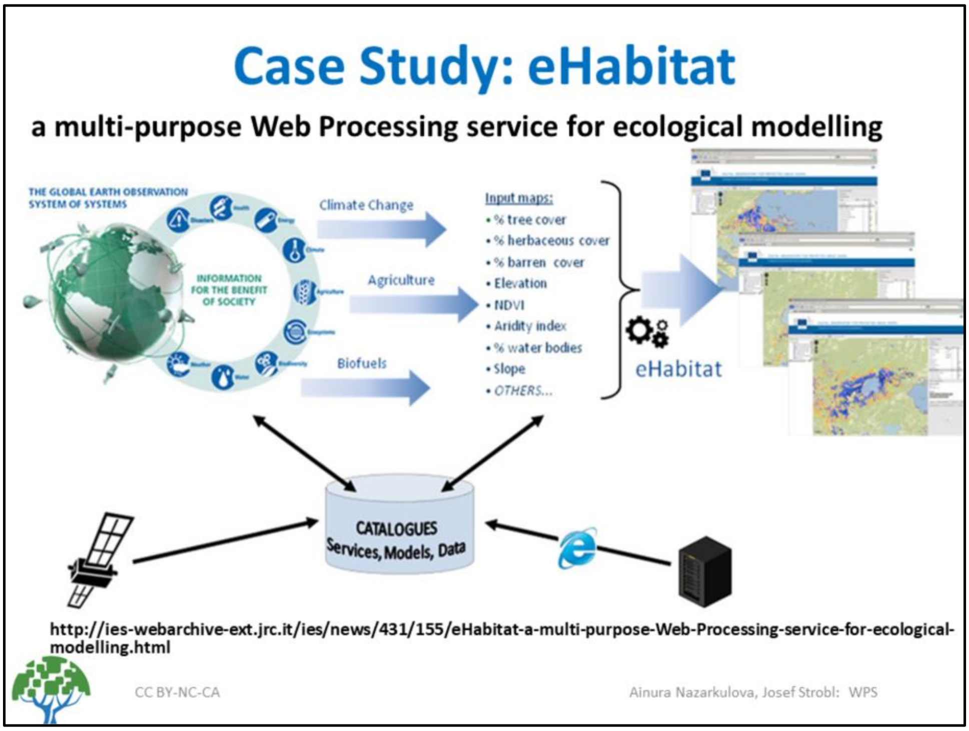
Case Study: SEEGrid

Source: https://www.seegrid.csiro.au/wiki/Siss/WPSImplementationIncubator#Overview

CC BY-NC-CA

Ainura Nazarkulova, Josef Strobl: WPS

The SEE Grid TWiki is a web-based collaboration area for the SEE Grid community. The aim of this project is to identify and establish the geoscience community specific interoperability “standards” for the APAC Grid network.



Case Study: eHabitat

a multi-purpose Web Processing service for ecological modelling

THE GLOBAL EARTH OBSERVATION SYSTEM OF SYSTEMS

INFORMATION FOR THE BENEFIT OF SOCIETY

Climate Change

Agriculture

Biofuels

Input maps:

- % tree cover
- % herbaceous cover
- % barren cover
- Elevation
- NDVI
- Aridity index
- % water bodies
- Slope
- OTHERS...

eHabitat

CATALOGUES Services, Models, Data

http://ies-webarchive-ext.jrc.it/ies/news/431/155/eHabitat-a-multi-purpose-Web-Processing-service-for-ecological-modelling.html

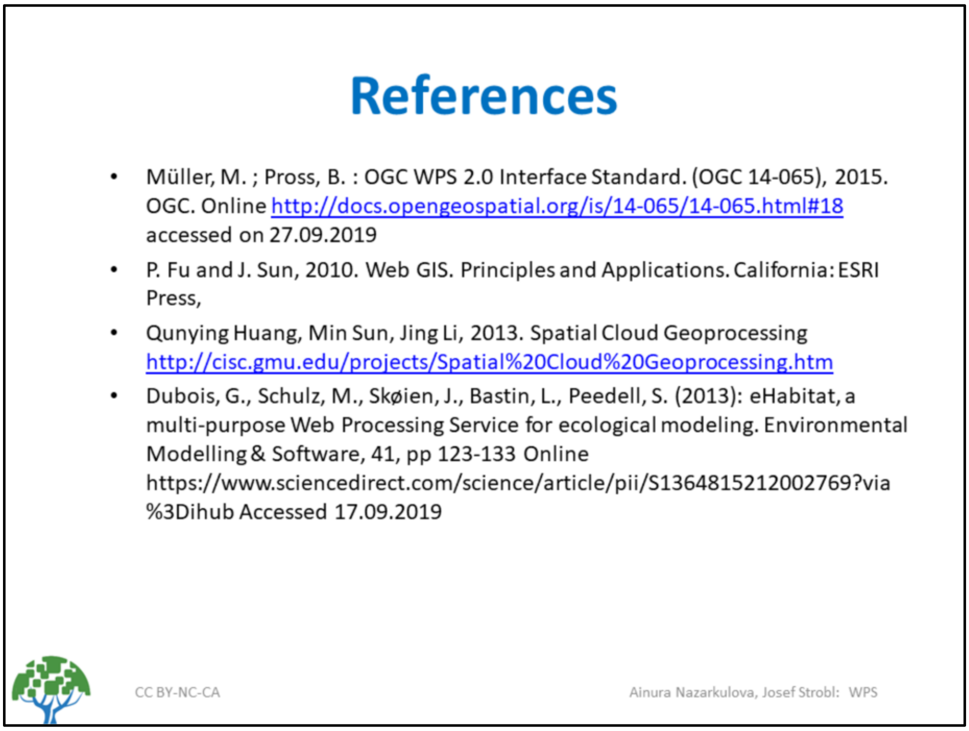
CC BY-NC-CA

Ainura Nazarkulova, Josef Strobl: WPS

eHabitat, a multi-purpose Web Processing service for ecological modelling

eHabitat is the core modelling service of the Digital Observatory for Protected Areas (DOPA), which is being developed at the JRC in collaboration with organisations including the Global Biodiversity Information System (GBIF), the UNEP-World Conservation Monitoring Centre (WCMC), Birdlife International and the Royal Society for the Protection of Birds (RSPB). DOPA aims to combine distributed databases with open, interoperable web services to help users assess the state of protected areas worldwide. eHabitat is a relatively simple and versatile modelling service which brings enough flexibility to DOPA to enable it to carry out ecological forecasting and consider alternative future scenarios using input from a variety of sources. Because of its simplicity, eHabitat can be re-used for a variety of purposes, from socio-economic modelling and ecological forecasting to the optimisation of environmental monitoring networks

Source: Dubois, G., Schulz, M., Skøien, J., Bastin, L., Peedell, S. (2013): eHabitat, a multi-purpose Web Processing Service for ecological modeling. Environmental Modelling & Software, 41, pp 123-133 Online https://www.sciencedirect.com/science/article/pii/S1364815212002769?via%3Dihub Accessed 17.09.2019



References

- Müller, M. ; Pross, B. : OGC WPS 2.0 Interface Standard. (OGC 14-065), 2015. OGC. Online http://docs.opengeospatial.org/is/14-065/14-065.html#18 accessed on 27.09.2019
- P. Fu and J. Sun, 2010. Web GIS. Principles and Applications. California: ESRI Press,
- Qunying Huang, Min Sun, Jing Li, 2013. Spatial Cloud Geoprocessing http://cisc.gmu.edu/projects/Spatial%20Cloud%20Geoprocessing.htm
- Dubois, G., Schulz, M., Skøien, J., Bastin, L., Peedell, S. (2013): eHabitat, a multi-purpose Web Processing Service for ecological modeling. Environmental Modelling & Software, 41, pp 123-133 Online https://www.sciencedirect.com/science/article/pii/S1364815212002769?via%3Dihub Accessed 17.09.2019