Title:

A5.2-D3 [3.0] A Lightweight Introduction to the HUMBOLDT Framework V3.0

Author(s)/Organisation(s):

Daniel Fitzner (FhG), Thorsten Reitz (FhG)

Working Group:

Architecture Team / WP5

References:

A5.2-D3 [3.1] Specification Introduction and Overview V3

A5.2-D3 [3.2] Mediator Service Component Specification

A5.2-D3 [3.3] Conceptual Schema Specification and Mapping

A5.2-D3 [3.3.1] HUMBOLDT Alignment Editor

A5.2-D3 [3.4] Context Service Component Specification

A5.2-D3 [3.5] Workflow Design and Construction Service Component Specification

A5.2-D3 [3.6] Processing Components General Model and Implementations

A5.2-D3 [3.7] Information Grounding Service Component Specification

A5.3-D3 Humboldt Commons / Framework Common Data Model V3

Quality Assurance:

☑	Review WP Leader:	Thorsten Reitz (FhG)
☐	Review dependent WP leaders:	
☑	Review Executive Board:	Eva Klien (FHG), Christine Giger (ETHZ), Dániel Kristóf (FOMI)
☐	Review others:	

Delivery Date: **02.12.2009**

Short Description:

This document gives a short introduction to the HUMBOLDT framework specification version 3. It contains a description of the concepts and components from a high-level point of view and gives pointers to the relevant specification documents. Further, it gives a detailed example of how the HUMBOLDT components are used within one of the HUMBOLDT scenarios (*Protected Areas*).

Keywords:

HUMBOLDT specification V3, components, example, high-level overview

History:

Version	Author(s)	Status	Comment
001	Daniel Fitzner	New	
002	Thorsten Reitz		Description of HALE included
003	Daniel Fitzner		Reworked according to comments from E.Klien
004	Daniel Fitzner		Integrated comments and text passages from C.Giger
005	Daniel Fitzner	FINAL	Finalized

Table of contents

Standards and Definitions

This section summarizes the standards and definitions used in this document.

Name:	***Definition:***
OGC Web Feature Service (WFS)	OGC standard defining a generic interface for web services offereing georeferenced feature-/vector-data.
OGC Web Map Service (WMS)	OGC standard defining an interface for web services offering georeferenced maps.
OGC Web Processing Service (WPS)	OGC standard defining a generic interface for webservices offering geoprocessing functionality (e.g. GIS operations).
Interface Control Document (ICD)	A document that describes the external interfaces that a software component exposes to external clients, including a description of the inputs and outputs. The ICDs for the HUMBOLDT components are specified using WSDL and XML-schema.
Web Service Description Language (WSDL)	A XML-based language for describing web services (i.e. the operations and input/output types) and how to access them.
Feature Portrayal Service (FTS)	Portrayal services for geodata (features) that are *loosely* coupled with the data sources that deliver the data to be portrayed.

1 Introduction and the HUMBOLDT Vision

In HUMBOLDT, harmonisation is defined as the process of *"creating the possibility to combine data from heterogeneous sources into integrated, consistent and unambiguous information products, in a way that is of no concern to the end-user"* (A 3.5-D1). Figure 1 gives a more detailed view on this overall HUMBOLDT vision.

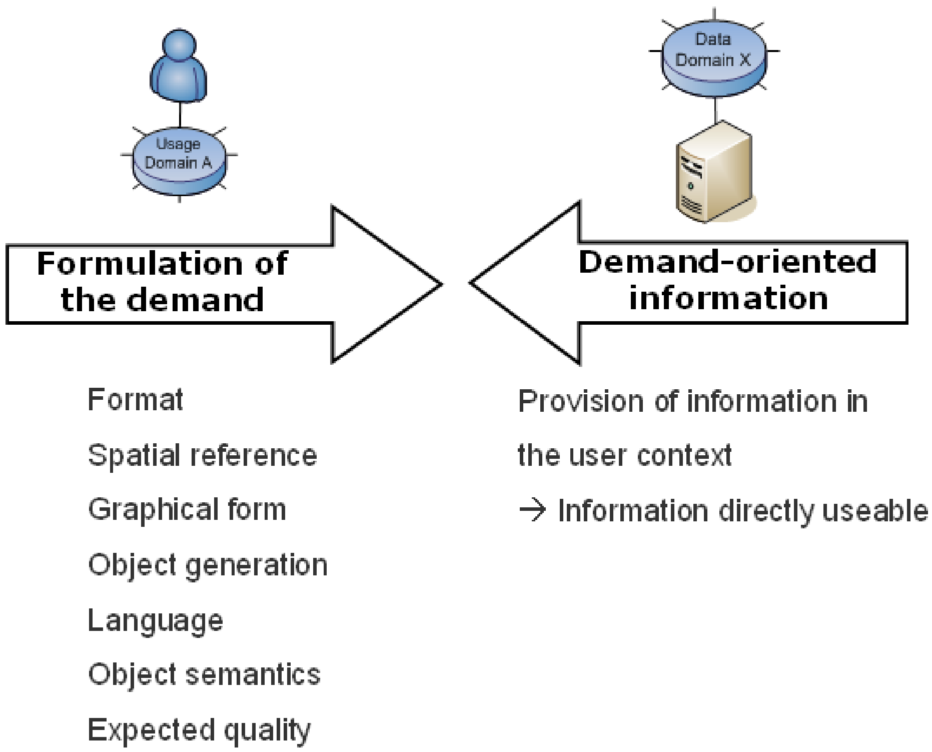

Figure 1: The HUMBOLDT Vision

Users shall be enabled to formulate their demand for geospatial data, involving requirements on the format, spatial reference system, data schema or language of the data. Based on a formalised description of the demand, information shall be provided to the user that is directly usable, i.e. that meets the requirements.

The aim of the HUMBOLDT framework is to provide the software for achieving or at least supporting this vision. This document is intended to give a high-level introduction on how this is done.

2 The HUMBOLDT Software

This document offers a lightweight introduction to the HUMBOLDT framework. Further, it is intended to be a short summary of the framework specification version 3 by pointing to the relevant specification documents.

The software produced in the HUMBOLDT project can be divided into the following main categories:

The HUMBOLDT Data Harmonisation Toolkit:

The HUMBOLDT data harmonisation toolkit consists of a set of tools with end-user interfaces. Details can be found in Section 2.1 .

The HUMBOLDT Framework for Service Integration:

The HUMBOLDT Framework for Service Integration consists of software components that enable users to publish and consume geodata harmonised to their requirements, via OGC- or INSPIRE-compliant interfaces. Details can be found in Section 2.2.

The HUMBOLDT Processing Components:

The HUMBOLDT Project comes with a set of components / services for geodata harmonisation. Details on these components and information on how to extend the framework with additional components can be found in Section 2.3 .

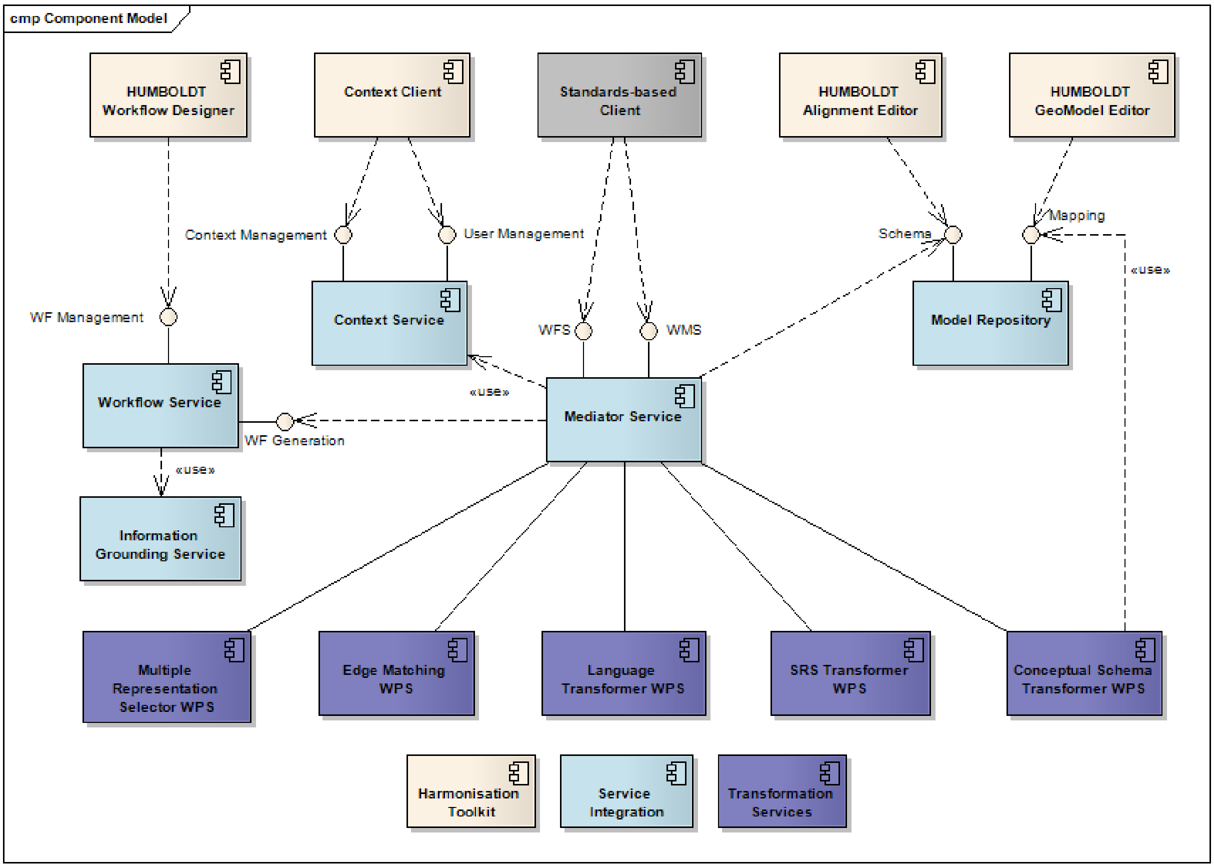

Figure 2: The HUMBOLDT Software for Data Harmonisation

2.1 The HUMBOLDT Data Harmonisation Toolkit

The HUMBOLDT data harmonisation toolkit consists of a set of tools with graphical user interfaces that enable users to perform different tasks related to the HUMBOLDT software, such as conceptual schema mapping, workflow or context definition.

2.1.1 The GeoModel Editor

The HUMBOLDT GeoModel Editor is an easy-to-use editor for application experts, aiming at collecting all required information on the geodata. It provides a formal representation of the information on the geodata in order to feed it into a harmonisation process.

Based on the findings on modelling principles (cf. HUMBOLDT Annual Report 2008) it uses a “spatial UML” (HUMBOLDT modelling language), which uses similar concepts as UML but is strictly restricted to geodata modelling, taking into account all requirements and recommendations of the INSPIRE data specification guidelines.

The HUMBOLDT GeoModel Editor is producing and providing a graphical and a textual representation of the data model containing basic spatial data types. It was implemented on a model-based framework (Eclipse) and thus is able to support the so-called vertical mapping, which is the serialisation to transfer standards or other representations (e.g. XMI, GML, INTERLIS, ISO19131). Its detailed specification was already produced for Framework Version 2 and can be found in the respective deliverable A5.2-D2 [2.2.1].

2.1.2 The Alignment Editor HALE

The HUMBOLDT Alignment Editor, short HALE, is a tool with a rich graphical user interface for defining mappings between concepts in conceptual schemas (application schemas created with the HUMBOLDT GeoModel Editor), as well as for defining transformations between attributes of these schemas.

The HUMBOLDT Alignment Editor has several properties which make it stand out from other data transformation definition tools. Among these are the following aspects:

- Definition of a mapping that is independent from concrete data sources;
- Continuous quality assurance of the created mapping;
- A Task-based user interface that makes a complex mapping process manageable and deterministic;
- The possibility to use geographic data to interactively visualize and test defined mappings;
- HALE also allows to document limitations that a mapping has, e.g. by providing descriptions of known mismatches.

More detailed information on the HUMBOLDT Alignment Editor can be found in the document *A5.2-D3 [3.3.1] HUMBOLDT Alignment Editor*. Information on the general approach for schema specification and mapping in HUMBOLDT, e.g. on the mapping language used etc., can be found in the document *A5.2-D3 [3.3] Conceptual Schema Specification and Mapping*.

2.1.3 The Context Client

The Context Client allows users to create and manage contexts and user / organisation profiles. Within HUMBOLDT, the term *Context* refers to a set of constraints on geospatial data sets, such as constraints on language, spatial reference system or bounding box. *Contexts* can be linked to *Organisation-* and / or *User Profiles,* where information on users and organisations is maintained.

The Context Client is the graphical user interface for the HUMBOLDT Context Service. Currently, the Context Client does not have a specification document.

2.1.4 The Workflow Frontend

The Workflow Frontend is a workflow designer and the graphical user interface of the Workflow Design and Construction Service (WDCS). It allows users to *register*, *manage* and *graphically compose* geoprocessing components into workflows. The Workflow Frontend therefore offers a quite similar functionality as e.g. the GUI of the *ArcGIS Model Builder* or a BPEL Workflow Designer. Currently, the Workflow Frontend does not have a specification document.

2.2 The HUMBOLDT Framework for Service Integration

The HUMBOLDT Framework for Service Integration consists of software components that enable users to publish geodata and to consume geodata via OGC-conformant interfaces, harmonised and transformed to their requirements. It consists of the following services:

- *the Context Service* (CS), a service for managing product descriptions for transformation results, users and organisations;
- *the Mediator Service* (MS), a harmonisation workflow execution engine that offers transformation, download and view service interfaces for seamless integration with existing environments;
- *the Workflow Design and Construction Service* (WDCS), a component for analysing harmonisation needs and for constructing workflows;
- *the Information Grounding Service* (IGS), is the HUMBOLDT catalogue. It provides cascading catalogue functionalities by periodically harvesting external catalogues.
- *the Model Repository* (MR), a conceptual schema and mappings repository.

2.2.1 The Context Service

The HUMBOLDT Context Service enables user management and context management for HUMBOLDT users. A HUMBOLDT context is a set of constraints on geospatial data sets, such as constraints on the **schema**, **spatial reference system** or **language**. A context can be linked to users and organisations.

Therefore, the *HUMBOLDT Context Service* supports the HUMBOLDT vision by enabling users to formulate and formalise their demand on geospatial data. It allows users to specify the behaviour of a standard OGC conformant data providing web service, i.e. the *HUMBOLDT Mediator Service*, by providing their requirements on spatial data in the form of a set of constraints, i.e. the context.

A detailed specification of this service component can be found in the document *A5.2-D3 [3.4] Context Service Specification*.

2.2.1.1 Use of the Context

A detailed description of the potential elements of a context, i.e. the constraint types, can be found in the document *A5.3-D3 HUMBOLDT Commons Specification / Framework Common data model 3.0.* The context as specified by users then plays a role in the following two processes.

- First, it is used for **discovery** in HUMBOLDT; the capabilities that are returned to the users show the subset of services that can potentially fulfil the user requirements, either directly or after harmonisation.
- Second, in case a user issues a request for data, the information within the context is used as a source for filtering / transforming / **harmonising** instance data. This transformation is performed by the Mediator Service and consists of either the application of a filter on a data

service (e.g. a WFS Filter), the application of some processing directly accessible to the framework or the application of some external processing component, e.g. external WPS registered to the framework.

2.2.1.2 Interfaces and Collaborations of the Context Service

The HUMBOLDT context service offers two interfaces. A *Graphical User Interface*, called the *Context Client*, can be employed by users to specify the context.

Second, the Context Service acts as a ***web service*** to external components and delivers (previously specified and stored) contexts via this interface. Within the HUMBOLDT framework, the web service interface of the Context Service is used by the HUMBOLDT Mediator Service to retrieve a user context.

The interface control document can be found in the specification of the HUMBOLDT Context Service (see *A5.2-D3 [3.4] Context Service Specification* and *A5.2-D3 Interface Control Documents for HUMBOLDT framework components*) in the form of a WSDL file.

2.2.2 The Mediator Service

The HUMBOLDT Mediator Service is the main controller of the other HUMBOLDT components. It offers a number of standard OGC interfaces like WMS, WFS or WCS to clients. In contrast to standard OGC services, it does not hold a data store but assembles the data sets dynamically via the *HUMBOLDT Information Grounding Service (IGS)*. This means, in the presence of a service request (*OGC GetCapabilities*, *GetMap*, *GetFeature* etc.) it dynamically discovers data sources via the IGS that match the context of the user (the set of constraints) either directly or that can be transformed such that they satisfy the context.

In case of a request on the capabilities (*OGC GetCapabilities*), the Mediator Service only returns metadata (i.e. the capabilities) on the discovered data sources. In case of a request on spatial objects (*OGC GetMap, GetFeature*), it retrieves the discovered data sets, e.g. via executing some external Web Feature Service, harmonises the data according to the context and delivers the harmonised data to the client.

The possibility for harmonisation depends on a number of prerequisites such as the manual definition of a schema transformation (e.g. using HALE) or the accessibility of coordinate transformation services to the framework. The harmonisation, i.e. the execution of a chain of geoprocessing / harmonisation components is based on workflow definitions delivered by the *HUMBOLDT Workflow Design and Construction Service*.

What makes the *HUMBOLDT Mediator Service* unique is that it combines a number of different functionalities and hides them behind standard OGC interfaces. It is a *Workflow Engine*, capable of executing chains of geoprocessing services. Further, it is a *Feature Portrayal Service*, dynamically portraying Features e.g. from Web Feature Services and serving them via the OGC WMS interface.

The document *A5.2-D3 [3.2] Mediator Service Component Specification* gives a detailed specification of the component.

2.2.2.1 Interfaces and Collaborations of the Mediator Service

As described above, the HUMBOLDT Mediator Service offers standard OGC-interfaces to external clients.

In case of a request for capabilities, the Mediator Service retrieves the context of the user via the web service interface of the *Context Service* and subsequently data sources matching the context via the *Information Grounding Service*.

In case of a request for data (GetFeature, GetMap), the Mediator Service retrieves the Context from the *Context Service* and then requests and retrieves executable workflows from the *Workflow Design and Construction Service*.

2.2.3 The Workflow Design and Construction Service

The HUMBOLDT Workflow Design and Construction Service enables users to *register* geoprocessing functionality, either encapsulated as OGC WPS or directly implemented on the platform on which the Mediator Service is deployed and to specify chains of registered components.

Further, it provides the functionality to support automated harmonisation. In case of an incoming request, i.e. a set of constraints specified in the HUMBOLDT Constraint Model (*see A5.3-D3 HUMBOLDT Commons Specification / Framework Common Data Model V3*), the WDCS either identifies a so-called *Basic Workflow* that represents the geoprocessing-based solution to a given task and that has been specified manually (see Section 3 for a detailed example). In case, such a *Basic Workflow* is available for the request, the WDCS employs the *Information Grounding Service* for discovering data sources that provide suitable input. In case, no such *Basic Workflow* is available, the WDCS directly forwards the request to the *Information Grounding Service*.

In case, the discovered data sources violate some of the constraints, the WDCS identifies suitable harmonisation services that are able to harmonise the data. The following points make the WDCS unique:

- It enables **the reuse of workflows** / chains of geoprocessing functionality in different contexts. This is achieved via the concept of manually defined so-called *Basic Workflows* that represent the geoprocessing-based solution to a given task. *Basic Workflows* do not contain bindings to concrete data sources or data sets. Since the bindings are discovered at runtime in the presence of a request, a single workflow definition can be easily reused and executed on a number of different data sources (covering different areas etc.) dynamically.

- Second, the WDCS enables the HUMBOLDT Mediator Service to perform **automated harmonisation**. The WDCS holds an *execution logic* that is essentially a mapping of *constraint - data source* combinations to transformation categories. This execution logic and the semantic annotation of harmonisation processing components with such categories enables the WDCS to dynamically and automatically identify harmonisation processing components that are able to solve a constraint violation. Therefore, the manual semantic annotation of harmonisation services is a prerequisite for automated harmonisation. Details on the form and scope of such annotations can be found in the documents *A5.2-D3 [3.6] Processing Components General Model and Implementations* and *A5.2-D3 [3.5] Workflow Design and Construction Service Specification.* For example, in the presence of a schema constraint violation for a data source, the WDCS is capable of dynamically identifying a schema transformation service (e.g. the *HUMBOLDT Conceptual Schema Transformer*) and delivering the information to the *Mediator Service* on how to execute that service (i.e. how to instantiate its inputs, with which values etc.) such that the output after execution satisfies the initial request / schema constraint.

The document *A5.2-D3 [3.5] Workflow Design and Construction Service Component Specification* gives more details on workflows in HUMBOLDT.

2.2.3.1 Interfaces and Collaborations of the WDCS

The WDCS offers a Graphical User Interface, called the *Workflow Frontend*, which can be used by users to register processing components and to specify chains / compositions of such components (i.e. workflows). The GUI offers therefore quite a similar functionality as e.g. the GUI of the *ArcGIS Model Builder* or a BPEL Workflow Designer but additionally provides assistance to the user in the composition process by e.g. comparing the input/output type definitions of the components to be connected. For example, this prevents users from connecting components processing *raster* data with processing component accepting *vector* data and hence reduces the risk of runtime errors when executing the composition.

Further, the WDCS offers a web service interface to clients (in HUMBOLDT, the Mediator Service, but could be any workflow execution service). Via this interface, it accepts requests expressed in the HUMBOLDT constraint model and delivers executable workflows in different workflow dialects, e.g. well-known formats such as BPEL as well as HUMBOLDT specific formats.

The *interface control document* can be found in the specification document in the form of a WSDL file.

2.2.4 The Information Grounding Service

The HUMBOLDT Catalogue is called the *Information Grounding Service* (IGS). The IGS is a cascading catalogue in the sense that it holds information on other catalogues and metadata stores in addition to metadata of data sources.

However, the IGS is not a usual catalogue. Common catalogue services accept requests for data sources that contain constraints on the data and they return the metadata of all data sources that satisfy the constraints. For example, users pass a query to catalogue services saying e.g.: “give me all data sources specified in spatial reference system X”. Common catalogues would return all data sources satisfying the request, i.e. those that are specified (or can be delivered) in spatial reference system X. What makes the HUMBOLDT Information Grounding Service unique is that it does not only return those data sources that directly satisfy a user request but additionally those that can **potentially** be transformed to satisfy the request. This sort of relaxed data source discovery supports the HUMBOLDT vision of the formulation of demand vs. provision of directly usable information, since it enables the harmonisation of geospatial data within the framework. The document *A5.2-D3 [3.7] Information Grounding Service Component Specification* gives a detailed specification of the service.

2.2.4.1 Interfaces and Collaborations of the Information Grounding Service

The HUMBOLDT Information Grounding Service offers a web service interface to clients.

The interface control document can be found in the specification document in the form of a WSDL file.

2.2.5 The Model Repository

The Model Repository is a service component that allows maintenance of application schemas (e.g. those created with the HUMBOLDT GeoModel Editor) and mappings between those (e.g. those created with the HUMBOLDT Alignment Editor). Within Version 2 and 3 of the HUMBOLDT framework, the Model Repository has not been part of specification work and hence we refer to the document available for Version 1, namely *A5.2-D1 Model Repository Component Specification*.

2.3 The HUMBOLDT Processing Components / Transformation Services

The HUMBOLDT framework handles (orchestrates etc.) processing components. The framework is able to handle processing services that are encapsulated within web services that conform to the OGC Web Processing Service standard or processing services that are directly implemented on the platform on which the HUMBOLDT Mediator Service is deployed. A more detailed description of the subsequent overview can be found in the document *A5.2-D3 Processing Components General Model and Implementations.*

2.3.1 Process Types

There are two types of HUMBOLDT processing components. Those that perform some sort of task related to the **HUMBOLDT constraint model** are ***called harmonisation processing components***. For example, a process performing spatial reference system transformation is, according to the HUMBOLDT characterisation, a harmonisation processing component. All others that perform tasks that do not directly relate to the HUMBOLDT constraint model, e.g. buffer, overlay or some application specific transformation, are so-called ***non-harmonisation processing components***.

Harmonisation processing components and non-harmonisation processing components are handled differently. Both can be composed using the WDCS GUI while only harmonisation processing components are handled automatically. However, this automated handling of harmonisation processing components by the WDCS is still subject to research and therefore not available in the current implementation of that service.

2.3.2 Harmonisation Processing Components

There are several processing components already implemented (or about to be implemented) as part of the framework. A detailed description on each of them can be found in the document *A5.2-D3 [3.6] Processing Components General Model and Implementations*.

Coordinate Reference System Transformer:

The Coordinate Transformation Service is a WPS implementation of a service that allows transforming coordinates between various geographic reference systems, i.e. Geoids and projections. Details on which spatial reference systems can be transformed can be found in the specification.

Conceptual Schema Transformer:

The Conceptual Schema Transformer is a Web Processing Service (WPS 1.0.0) that is able of applying a schema transformation to a source WFS dataset (Application Schema A) in order to provide a target dataset (Application Schema B). A schema mapping between schema A and schema B has to be defined in order to accomplish the transformation.

Multiple-Representation Merger:

The Multiple Representation Merger is a Web Processing Service (WPS 1.0.0) that is capable of fusing Features of data sets with a spatial overlap, such as along a common border, where water bodies are part of both data sets.

Edge-Matching Transformer:

The Edge Matching Service (EMS) is a WPS (1.0.0) implementation of a service that aligns edges and points of vector geometries so that they will be gapless.

Language Transforming Processor

The Language Transforming Processor enables the language transformation of data sets (attribute name and attribute value transformations). It employs the Language Transformer for translation.

Language Transformer:

The Language Transformer transforms single terms from one language to another. It uses an internal database that holds translations and that can be populated via the interface of the language transformer. Further, it employs the General Multilingual Environmental Thesaurus for retrieving transformations.

2.3.2.1 Extending the HUMBOLDT Framework with Processing Functionality

Extending the HUMBOLDT framework with additional processing components requires two steps, as shown in Figure 3.

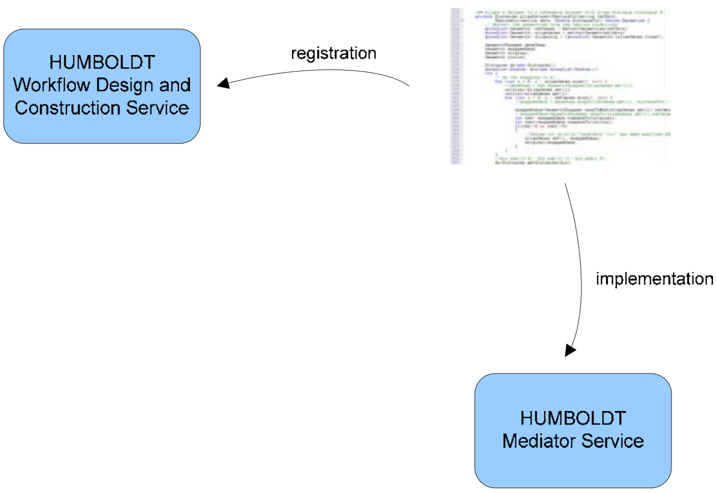

Figure 3: Implementation and Registration of new Processing Functionality

First, the process must be implemented and accessible to the *HUMBOLDT Mediator Service (MS)*. Details on what exactly to do and how to enable the MS to access the processing can be found in the document *A5.2-D3 [3.6] Processing Components General Model and Implementations*. In case, a process is or should be encapsulated within an OGC conformant WPS, there is no implementation on framework side required.

Independent on whether the processes are implemented as part of the framework or encapsulated within some external OGC WPS, they must be registered to the ***HUMBOLDT Workflow Design and Construction Service***. Details on the registration process can be found in the documents *A5.2-D3 [3.6] Processing Components General Model Specification* and the *A5.2-D3 [3.5] Workflow Design and Construction Service Specification*.

Extending the HUMBOLDT framework with a new harmonisation processing component requires the **semantic description / annotation** of that component, since harmonisation components could then be handled automatically.[1] How this is exactly done can again be found in the document *A5.2-D3 [3.6] Processing Components General Model and Implementations.*

[1] The semantic annotation of processing components **already implemented** in the scope of HUMBOLDT (see section 2.3.2) is not necessary since they are known at the time of programming the framework and therefore not handled dynamically. Further, the semantic annotation of harmonisation processing services does not refer semantic web-styled annotations with ontologies. Instead, the annotations can be compared to the so-called *WPS Application Profiles* in the OGC-world.

3 Scenario Integration of the HUMBOLDT Software

In order to illustrate how the HUMBOLDT software could be integrated into existing applications, the HUMBOLDT Scenario Protected Areas is used as an example. In the Protected Areas scenario, there are three goals that the organizations and people working in it want to achieve:

- *Improved tourism valorisation of the protected areas;*
- *Improved protected areas management;*
- *INSPIRE-IR conforming data provision and reporting.*

To reach the first goal, several different, sometimes conflicting views have to be merged: On the one hand, allowing the touristic exploitation of an area can have big advantages, such as making people understand why a certain area needs to be protected, but also to ensure a sustainable income for people living in the region of the protected area.

One the other hand, some areas are highly sensitive and should not be disturbed at all. Balancing the interests of nature conservation and touristic exploitation, which has to include making an interesting offer, therefore has been a topic since the first national parks and other protected areas have been introduced. To make good decisions and to offer interesting experiences to tourists, a wide range of different data need to be taken into account. These include information on protected areas and special protection zones, on flora and fauna habitats, on topography and hydrography and of course on touristic infrastructure and points of interest. All these different data sets have been accumulated over a manual process in the past, and have essentially been thrown together, with different problem occurring as a result of that. When the data sets are available, interesting and sustainable hiking paths and other touristic infrastructure can be planned and selected.

The second goal has much in common with the first one with respect to the required data sets, but has a broader focus – the big picture of protected areas management is required, which involves working with lots of different information sources, setting up reports and monitoring on-going developments and projects.

The third goal, the INSPIRE-IR-conforming data provision, has a different focus. Here, the protected areas agency is not providing data for its own use, but rather for the use of others partaking in a common European Spatial Data Infrastructure. This has many benefits such as making the data collected better accessible to the public, to researchers and to decision makers across Europe. To achieve this goal, the agency needs to provide the data that originates from its activities, such as information on species distribution and on habitats, via standardized services and using harmonised application schemas. For this purpose, a set of application schemas on protected sites, species distribution and other aspects have been or will be drafted by INSPIRE Data Specification Drafting teams. The agency now needs to ensure that data it collects and that falls within one of the themes defined by the INSPIRE directive is made available for View and Download services, again according to the regulations of the directive.

The following End-to-End example describes the task of creating a web portal that that can be used by hikers for identifying suitable hiking routes. The main goal is to keep the portal independent from concrete data sources in order to enable the reuse by different user groups e.g. for retrieving hiking routes in different areas. Another goal is to enable users to employ OGC-conformant clients for retrieving data from the portal. Hence, the portal must offer standardized OGC-interfaces for data access, such as WFS or WMS. The aim of this example is to show, how the task is achieved using the

HUMBOLDT framework components. The two other goals are also referenced where necessary, explaining the common methodology which can be used to satisfy all three goals.

Within the scenario description, the *data integrator* is embodied by Luigi, a programmer and IT-expert. Luigi is responsible for maintaining the IT-infrastructure of the protected areas management agency. The *data custodian* is represented by Carla, an expert in geospatial data models and – together with Luigi - responsible for the data sets involved. Finally, the end user of geodata is embodied by Mario Rossi Buhl, a regional officer at the Territorial Planning Department of an Italian region. Mario is the person responsible and with the initial need for the web portal on hiking paths. He is supported by Luigi and Carla in achieving his task.

Before the portal can built, the data sets involved in the calculation need to be identified. In order to be able to do this, Mario makes the methodology for the calculation of the hiking routes explicit, which is described as follows:

- The first and most important goal is that the hiking routes must lead through the protected areas managed by the protected areas management agency, Mario’s employer. The goal is to make hikers familiar with the preservation of nature and to improve the income of the people living in that area.
- Further, the hiking routes must lead close to stopping places, e.g. places for overnight staying.
- Additionally, there are some special protected areas that should not be entered by humans and therefore be avoided by the hiking paths.
- Finally, the delivered data on the hiking paths should include some information on the area that is crossed, such as forest or wood.

Based on these requirements, the following data sets involved can be identified.

Footpaths and Hiking Trails: This data set delivers data on potential hiking paths.

Protected Areas: This data set delivers both, data on protected areas that should be crossed by the hiking paths, as well as data on special protected areas that must be avoided by the paths.

Stopping Places: This data set delivers data on stopping places such as places to stay overnight or panoramic places.

Vegetation: This data set delivers data on the vegetation coverage, such as forest, wood or rocks.

Based on the methodology and the data sets involved, an abstract chain of processing steps can be identified for calculating the paths. This chain is shown in Figure 4.

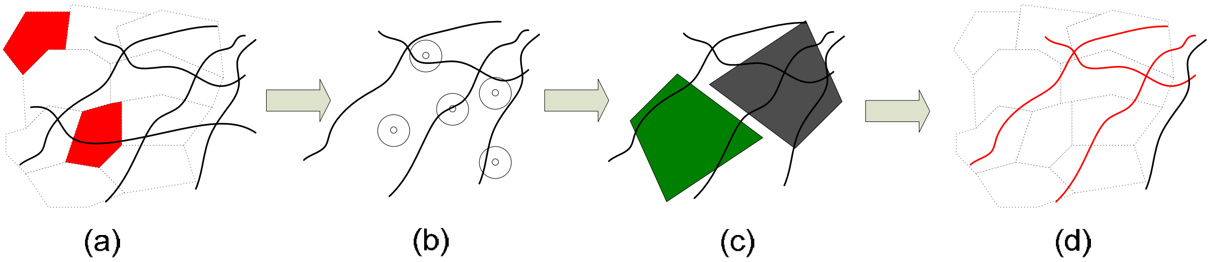

Figure 4: Methodology for the Calculation of Hiking Paths

First, those areas are selected that should not be crossed by the paths (a) and the paths not crossing such areas are identified. Then, a buffer is calculated around stopping places and only those hiking paths are selected, that are close to (at least one) of them (b). Further, the information on vegetation is attached to the hiking paths in (c) and finally, only those hiking paths are selected that cross protected areas (d).

Note that the identification and setting up of an application processing chain is of less importance for the first and second goal. Especially the INSPIRE-IR conforming data provision usually does not require sophisticated application specific processing.

3.1 Setting up the Portal

The aim of this section is to show how the HUMBOLDT framework can be used for building the web portal described in the previous section. Before most of the HUMBOLDT components can be applied and the web portal can be deployed, there are several preconditions. Some are concerned with setting up the infrastructure (web services, HUMBOLDT components) and some with preparing the data sets involved for further processing.

3.1.1 Setting up the Infrastructure

Involved HUMBOLDT User Groups: Data Integrator (Luigi)

HUMBOLDT components: Information Grounding Service (IGS), Workflow Design and Construction Service (WDCS)

Luigi, the *data integrator*, is responsible for maintaining the IT infrastructure of the protected areas management agency. For doing this, Luigi has to accomplish several tasks, exemplarily shown in the following. First, he downloads and deploys all HUMBOLDT components necessary for building the web portal on hiking paths. In this example, all HUMBOLDT components are needed.

Further, all of the involved data sets need to be published via OGC conformant interfaces, e.g. WFS in the case of vector-data, WCS for raster etc. and they must be registered to some geospatial catalogue. These two tasks are outside the HUMBOLDT framework itself and do not involve HUMBOLDT components.

For being able to apply the framework and to profit from its capabilities, Luigi registers the individual catalogues that hold the metadata of available data sources to the *Information Grounding Service* (IGS). The IGS is the HUMBOLDT component responsible for the discovery of geospatial data services. It does not replace existing catalogues but serves as a catalogue of catalogues and is therefore capable of discovering a huge number of different data sets, e.g. those covering different administrative areas and registered to different catalogues.

Furthermore, the processing components required for calculating the hiking paths (such as the *Intersection*-operation) need to be registered to the HUMBOLDT framework. Since most of the processing components involved in the calculation of the hiking paths are well-known and widely used such as Intersection and Buffer, they can be reused from already existing applications and do not need to be implemented again. HUMBOLDT enables this reuse by allowing users to register geoprocessing functionality implemented within OGC Web Processing Services to the framework.

3.1.2 Schema Definition and Schema Mapping

Involved HUMBOLDT User Groups: Data Custodian (Carla)

HUMBOLDT components: Model Editor, Alignment Editor, Model Repository

Since the processing chain identified in the previous section includes some processing that can only be applied to data that adheres to a data schema known at the time of building the processing chain (e.g. selection based on non-spatial attributes), the data schemas of all data sets need to be mapped to a (previously defined) integrated schema. For example, the existing data services that deliver data on “Protected Areas” use different and heterogeneous data schemas. In order to be able to process all the different heterogeneous data sets, an integrated “Protected Areas” schema needs to be defined and all individual source schemas need to be mapped to that schema. Figure 5 shows an extract of the integrated schema on “Protected Areas”. This schema mapping enables – at runtime – the automated translation of data instances (e.g. GML instance files) from the source to the target schema and the subsequent data processing w.r.t. the target schema.

For defining the integrated schemas, Carla employs the **HUMBOLDT Model Editor**, a UML-based editor enriched with geo-specific constructs.

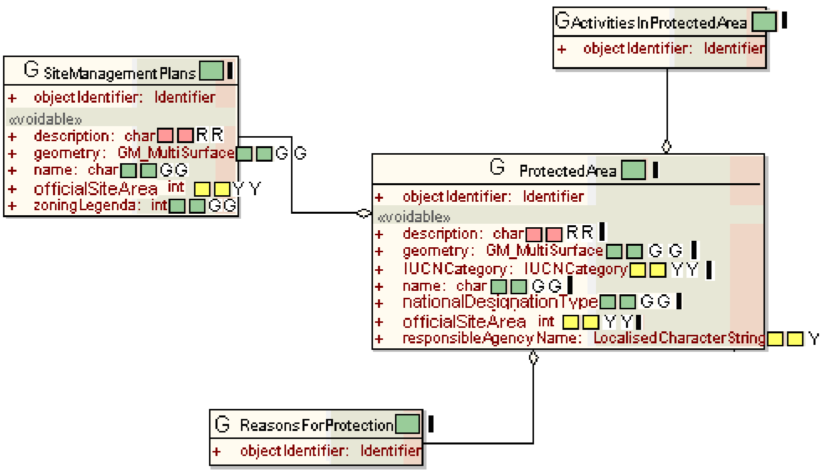

Figure 5: Extract Protected Areas Integrated Schema

The component responsible for the definition of mappings between a source and a target schema is **the HUMBOLDT Alignment Editor (HALE)**, shown in Figure 6. Carla employs HALE for mapping the individual data schemas to the integrated ones. Finally, she stores the newly defined schema and the mapping definitions in the **HUMBOLDT Model Repository**.

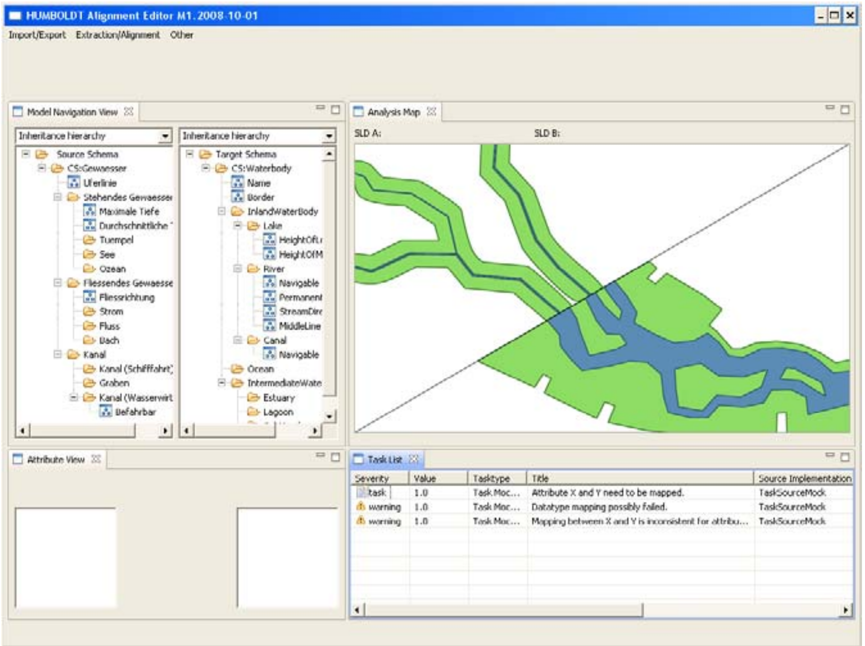

Figure 6: The HUMBOLDT Alignment Editor

At runtime, the mappings are published to the conceptual schema transformer, a WPS that executes them on data instances.

3.1.3 Definition of the Application Specific Processing Chain

Involved HUMBOLDT User Groups: End User (Mario), supported by Data Integrator (Luigi)

HUMBOLDT Components: Workflow Design and Construction Service (WDCS), WDCS GUI, Geospatial Processing Components (e.g. WPS) registered to HUMBOLDT

The abstract chain of processing steps shown in Figure 4 can be transferred to a chain of concrete geoprocessing functionality. This is done by end user Mario using the graphical user interface of the **Workflow Design and Construction Service (WDCS),** called the **Workflow Frontend (WF)**. The GUI enables him to connect the processing components registered to the system (e.g. external WPS) into a chain whose execution calculates the hiking paths according to the methodology for hiking routes calculation described in the introduction. The resulting chain of geoprocessing functionality is called a **Basic Workflow** in HUMBOLDT terminology. Figure 7 shows the input/output signature of the Basic Workflow “Sustainable Hiking Paths”, abstracting from the concrete chain of processing.

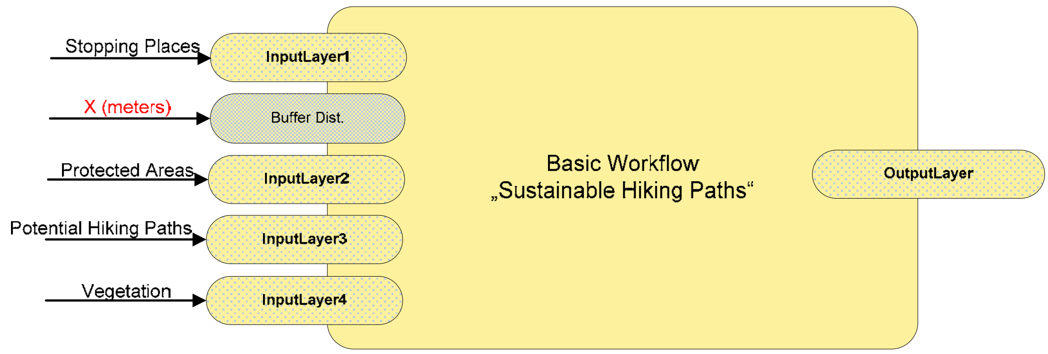

Figure 7: Abstract view on the Basic Workflow "Sustainable Hiking Paths"

The Basic Workflow “Sustainable Hiking Paths” is abstract in the sense that it is data independent. This means, it does not hold information on concrete data services that deliver input. Instead, it contains constraints on the potential input data sets, such as schema constraints form the integrated “Protected Areas” schema. During execution, concrete data services (grounding services) that satisfy the input constraints are automatically discovered and attached to the workflow, resulting in an **Executable Workflow**, in HUMBOLDT terminology.

The schema constraint are necessary, since – although most of the processing components within the workflow are schema-independent and operate on the GML-geometries – there are schema specific processing components encapsulated within the Basic Workflow, e.g. selection of features based on a non-spatial attribute of the “Protected Area” schema. This selection is defined at workflow design time without the knowledge of concrete data sources and, since it must be schema dependent, the schemas of the inputs must be fixed.

Finally, Mario registers the Basic Workflow to the HUMBOLDT system as the new Feature Type *“Sustainable Hiking Paths”.* The Basic Workflow is stored within the WDCS workflow repository and can now be requested as any other Feature Type.

Note that, especially the provision of INSPIRE-IR conformant data usually data does not require the definition of an application specific processing chain but is more concerned with schema mapping and harmonisation.

3.2 Retrieving “Sustainable Hiking Paths”

Involved HUMBOLDT User Groups: End User of Geodata / Geoinformation (Mario), Data Custodian (Carla)

Involved HUMBOLDT components: WDCS, MS, CS, IGS

Features of the Feature Type „Sustainable Hiking Paths“ can now be retrieved by end users of geoinformation. Every time such request is issued, the workflow is executed. Since the Basic Workflow is data independent, it can be reused by different user groups with different specific needs on the data to be returned, such as specific areas or spatial reference systems. The set of concrete end user constraints on the data to be returned is called a *Product Definition*.

3.2.1 Product Definition

Involved HUMBOLDT User Groups: End User of Geodata / Geoinformation (Mario), Data Custodian (Carla)

Components: Context Service, Context Service Client

For setting requirements, Mario accesses the **Context Service (CS)** by using the **Context Service Client**. Since the web portal delivers the data via standardized OGC-interfaces, only those constraints on the data to be returned that are allowed within standard OGC-requests can be directly passed to the system. The aim and raison d'être of the context service is to allow Mario to set additional constraints on the data to be returned that can not be set within standard OGC requests. A context example is shown in **Table 1** and explained in the following.

Constraint Type	**Constraint Value**
Feature Type:	"Sustainable Hiking Paths"
Spatial Reference System:	WGS84
Bounding Box:	{ …, …, / …,… }
Language:	Italian
Quality:	"… the data must be collected within the last five years."
…	

Table 1: A user's context

The Feature Type requested is "Sustainable Hiking Paths", which represents the Basic Workflow created by Mario beforehand. The data to be returned should adhere to the reference system *WGS84* and additionally cover a certain area represented by the bounding box constraint. Further, the requested language is Italian and the data sets involved must be collected within the last five years. While the Feature Type, the SRS and the BBox can be set within standard OGC requests, the language and quality constraint can not.

3.2.2 Workflow Construction and Execution

Involved HUMBOLDT User Groups: End User of Geodata / Geoinformation (Mario)

HUMBOLDT Components: Mediator Service, Context Service, Workflow Design and Construction Service, Information Grounding Service

After defining the context, Mario requests – using an OGC-conformant client – the data. The component responsible for handling the request is the HUMBOLDT Mediator Service.

Within the Mediator Service, the request is merged with the context, retrieved from the *Context Service*. The set of constraints of request and context can contain common elements, e.g. the Feature Type requested. In this case, the request-constraints override the constraints set within the context. For example, the bounding box constraint Mario sends within the direct request overrides the bounding box constraint previously set within the context.

Based on the user request and the context, the Basic Workflow “Sustainable Hiking Paths” is retrieved and enriched with request specific parameters, e.g. the specific bounding box and spatial reference system requested by Mario.

Discovery and automated Harmonisation:

The input descriptions of the Basic Workflow are now passed as a discovery query to the Information Grounding Service. The IGS returns – for each single input to the Basic Workflow such as “Protected Areas” – a pointer to a number of data services (e.g. WFS) that can deliver data.

But since Mario requested data for an area that crosses the boundary between two countries and that is therefore not served by one single data service, only a spatial combination of the two data sources covers the area requested by the Mario. Figure 8 shows the two data services discovered for “Protected Areas”.

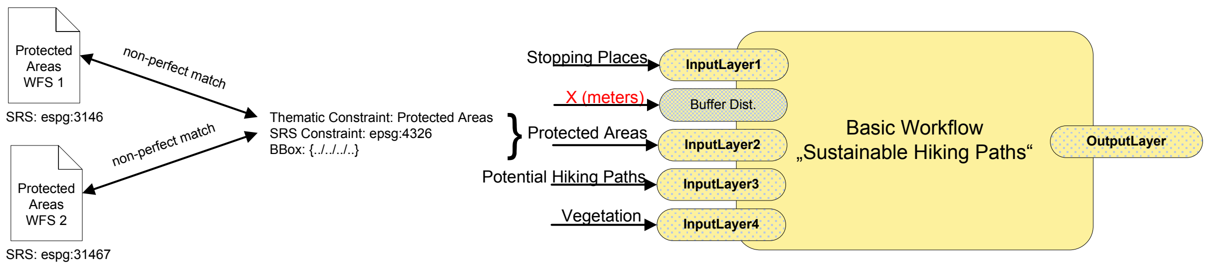

Figure 8: Discovery of input

The harmonisation requirements and the solution that is applied are shown for the “Protected Areas” layer in the following. The two data sources discovered (Protected Areas WFS1 and WFS2) can – if spatially combined – deliver input data to the Basic Workflow such that the output covers the area requested by Mario. However, both WFS deliver data within different reference systems since they are maintained by different data providers from different countries. Moreover, the method of data acquisition is different and therefore, the two data sets differ in precision, which is directly visible on the boundary of the two regions, as shown in Figure 9.

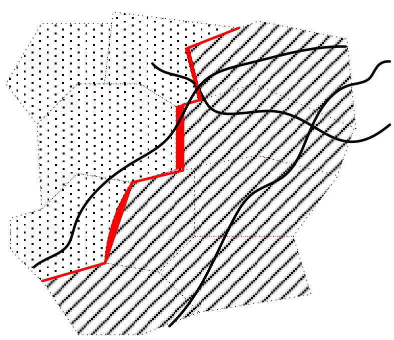

Figure 9: Harmonisation Requirements

Hence, before both WFS can serve as input to the Basic Workflow, they require harmonisation of spatial reference systems as well as the alignment of their common boundaries, a process known as *Edge Matching*. Based on the metadata of the discovered services WFS1 and WFS2 and the constraints on the input to the Basic Workflow representing "Protected Areas", the required harmonisation transformations are automatically identified and attached to the Basic Workflow. Additionally, the data delivered by both WFS is transformed to the integrated target schema for protected areas, based on the predefined schema mapping. The workflow resulting from the automated insertion of harmonisation transformers is shown in Figure 10.

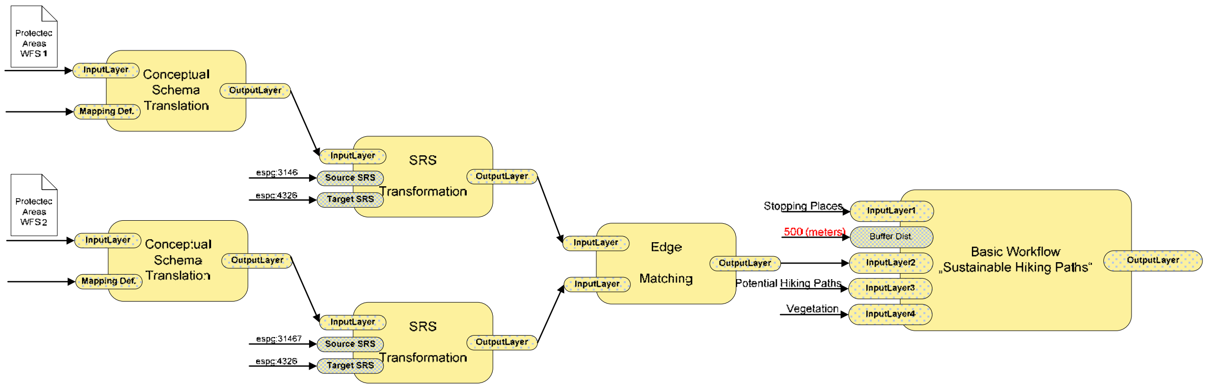

Figure 10: Executable Workflow (Abstraction)

The automated process of adding harmonisation transformations to an application specific processing chain is called workflow construction and is performed for every single input (except the Buffer distance, which is delivered by the user) to the Basic Workflow if required.

This automated harmonisation workflow construction takes place within the WDCS. When finished, an executable workflow description is returned to the Mediator Service. After executing the workflow, the Mediator Service returns the result to Mario. Finally, Mario uses the print-function of his client application and prints the hiking routes for his next vacation.