European Commission

CSPA Architecture

Antonino Virgillito

Istat

Outline

- CSPA Application Architecture
  - **Statistical Service Layers**
  - **CSPA Roles**
  - **Architecture Design**
- CSPA Technical Architecture
  - **Communication Plaftorm**

APPLICATION ARCHITECTURE

What is Application Architecture?

Application Architecture (AA) classifies and hosts the individual applications describing their deployment, interactions, and relationships with the business processes of the organization (e.g. estimation, editing and seasonal adjustment tools, etc.).

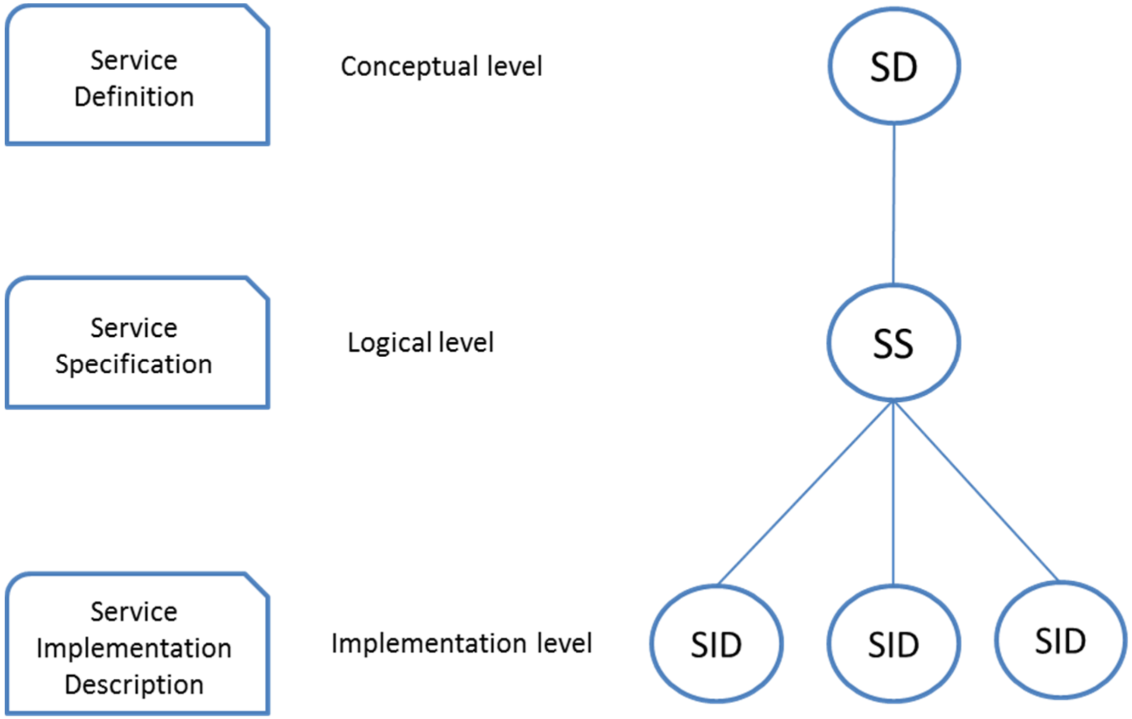
AA facilitates discoverability and accessibility, leading to greater reuse and sharing.

The CSPA Application Architecture is based on an architectural style called Service Oriented Architecture (SOA). This style focuses on Services (or Statistical Services in this case).

Statistical Services

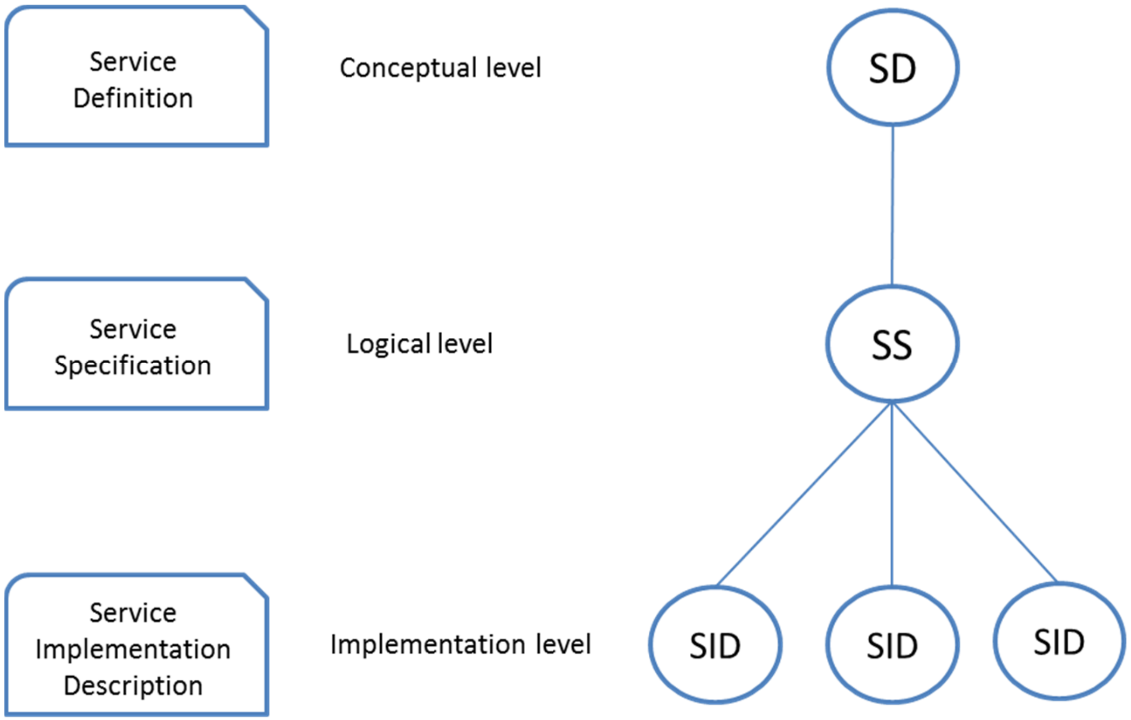
- The level of reusability promised by the adoption of a SOA is dependent on standard definitions of the services.
- CSPA has three layers to the description of any service:
  - **Service Definition**
  - **Service Specification**
  - **Service Implementation Description**

Statistical Services



Statistical Services

The capabilities of a Statistical Service are described in terms of the GSBPM sub process that it relates to, the business function that it performs and GSIM information objects which are the inputs and outputs.



Service Definition

Conceptual level

Service Specification

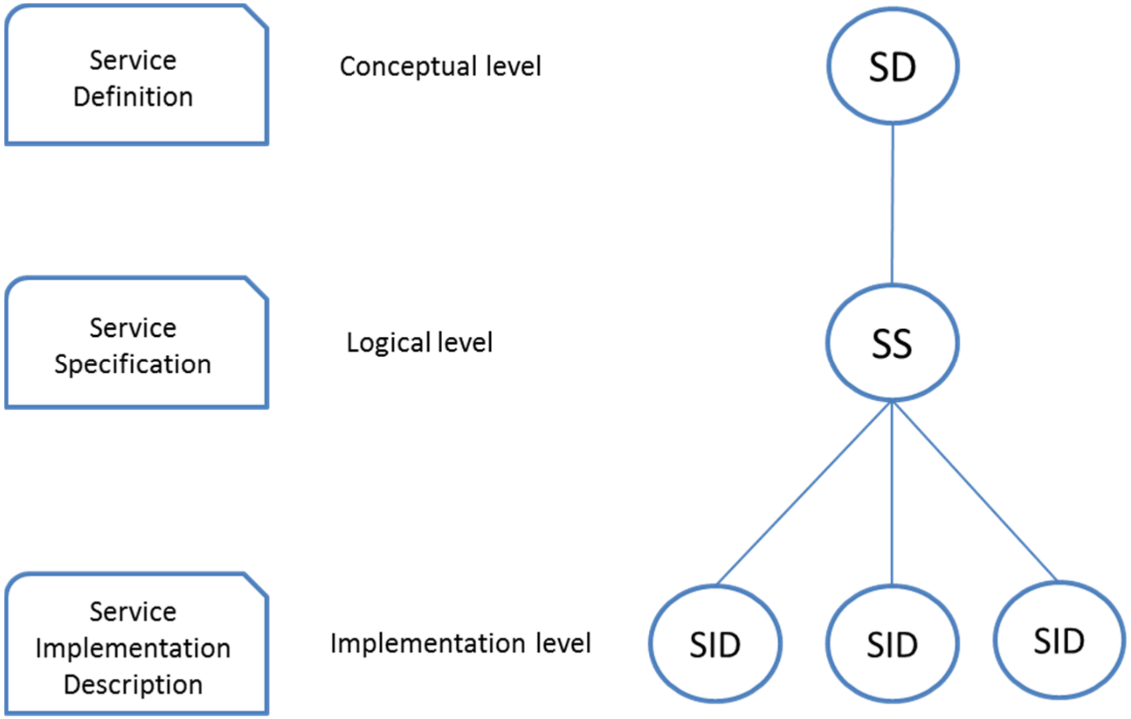
Logical level

Service Implementation Description

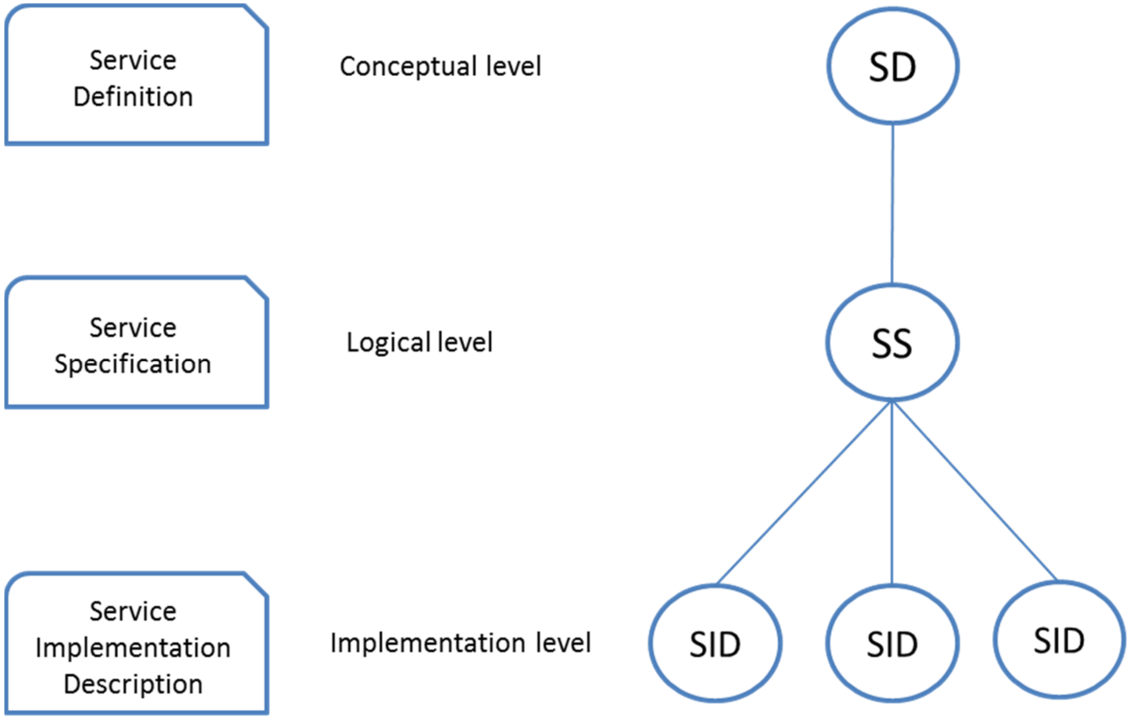
Implementation level

Statistical Services

The capabilities of a Statistical Service are fleshed out into business *functions* that have GSIM implementation level objects as inputs and outputs. This document also includes metrics and methodologies.



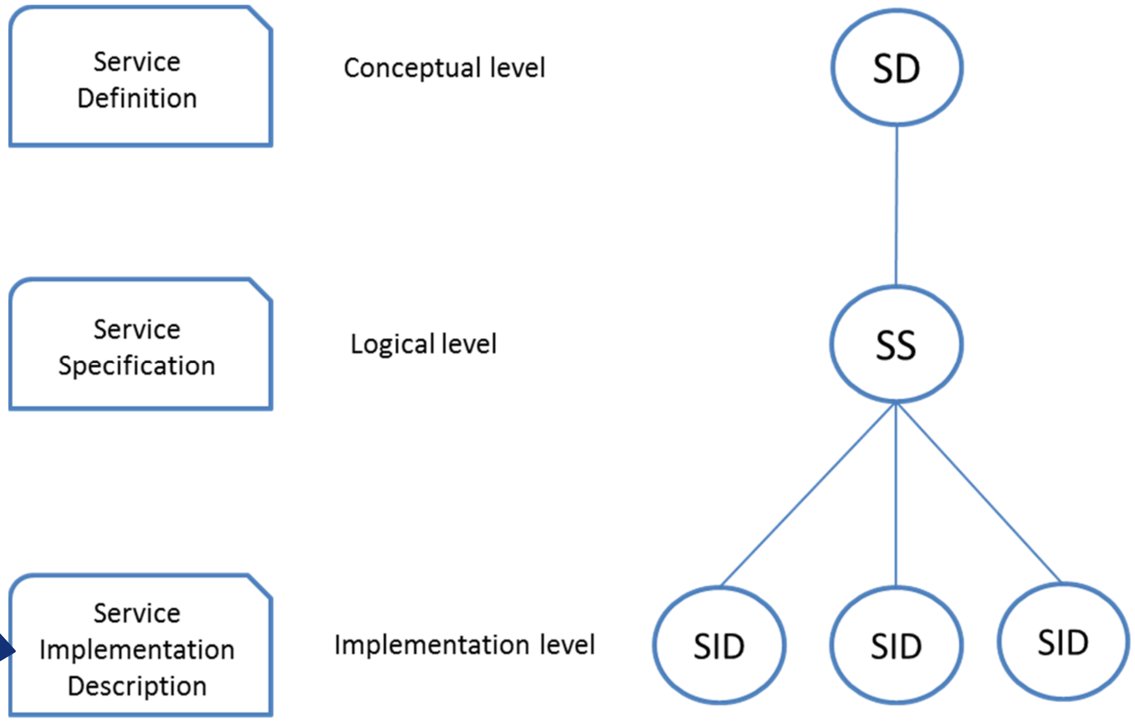
Statistical Services



The functions of the Statistical Service are refined into detailed *operations* whose inputs and outputs are GSIM implementation level objects.

Statistical Services

This layer fully defines the service contract, including communications protocols, by means of the Service Implementation Description. It includes a precise description of all dependencies to the underlying infrastructure, non-functional characteristics and any relevant information about the configuration of the application being wrapped, when applicable.



Service Definition

Conceptual level

SD

Service Specification

Logical level

SS

Service Implementation Description

Implementation level

SID

SID

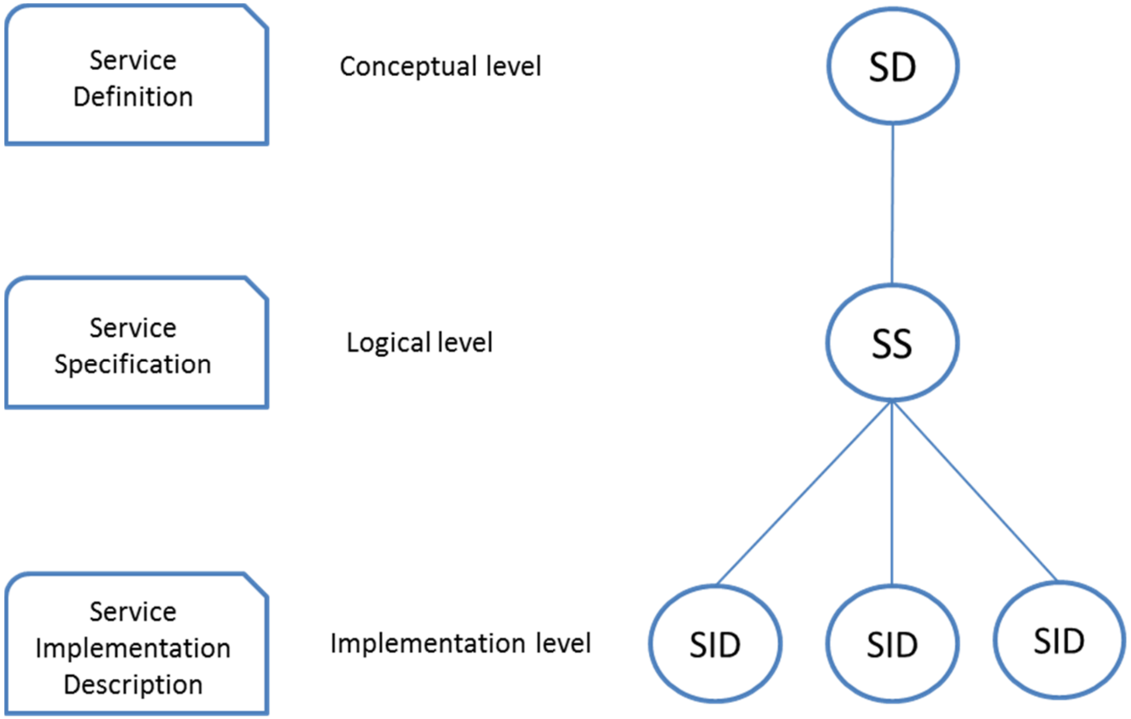
SID

Statistical Services

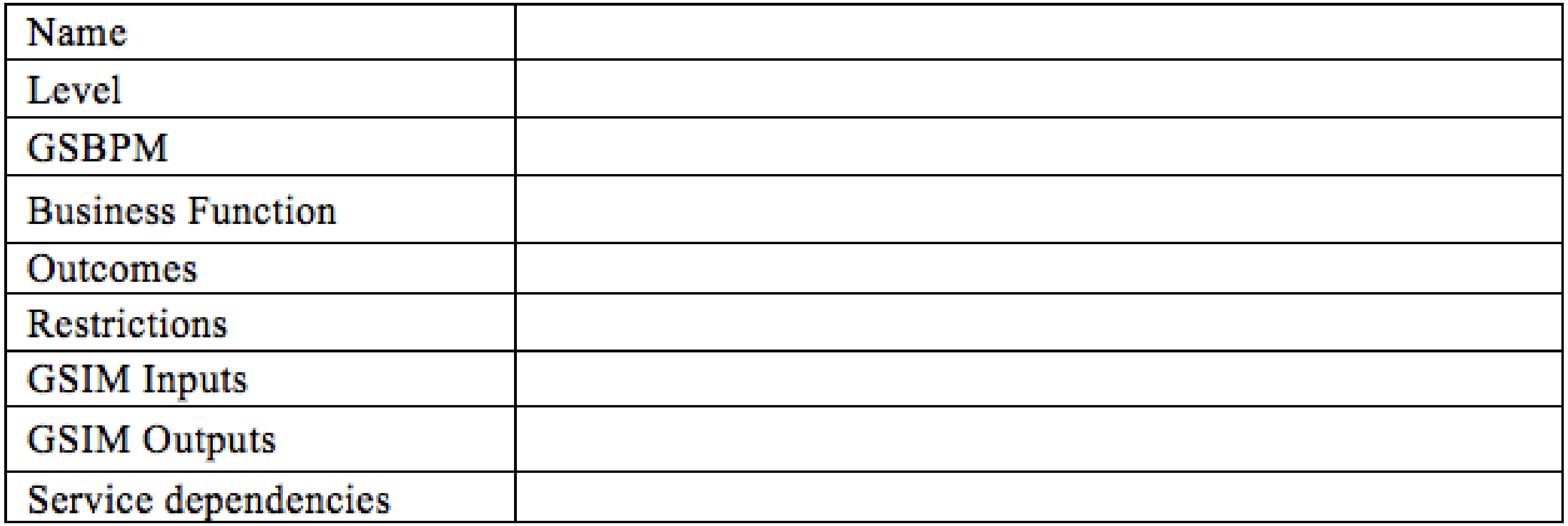
In general, there will be one Service Specification corresponding to a Service Definition, to ensure that standard data exchange can occur.

At the implementation level, services may have different implementations reflecting the environment of the supplying organization.

Each implementation must rigidly adhere to the data format specified in the Service Specification.

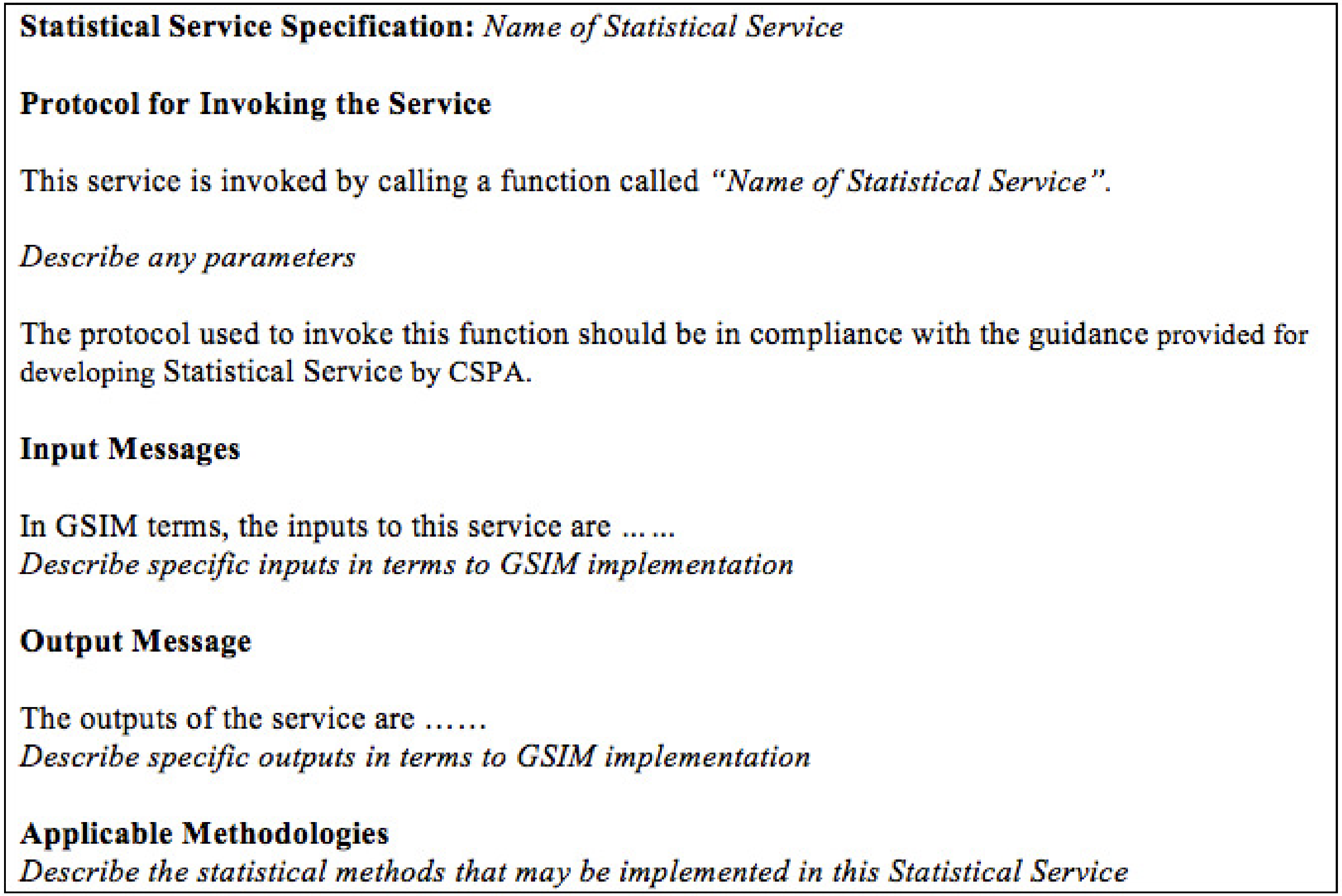


Statistical Service Definition Template



Name	
Level	
GSBPM	
Business Function	
Outcomes	
Restrictions	
GSIM Inputs	
GSIM Outputs	
Service dependencies	

Statistical Service Specification Template



Statistical Service Specification: *Name of Statistical Service*

Protocol for Invoking the Service

This service is invoked by calling a function called *"Name of Statistical Service"*.

Describe any parameters

The protocol used to invoke this function should be in compliance with the guidance provided for developing Statistical Service by CSPA.

Input Messages

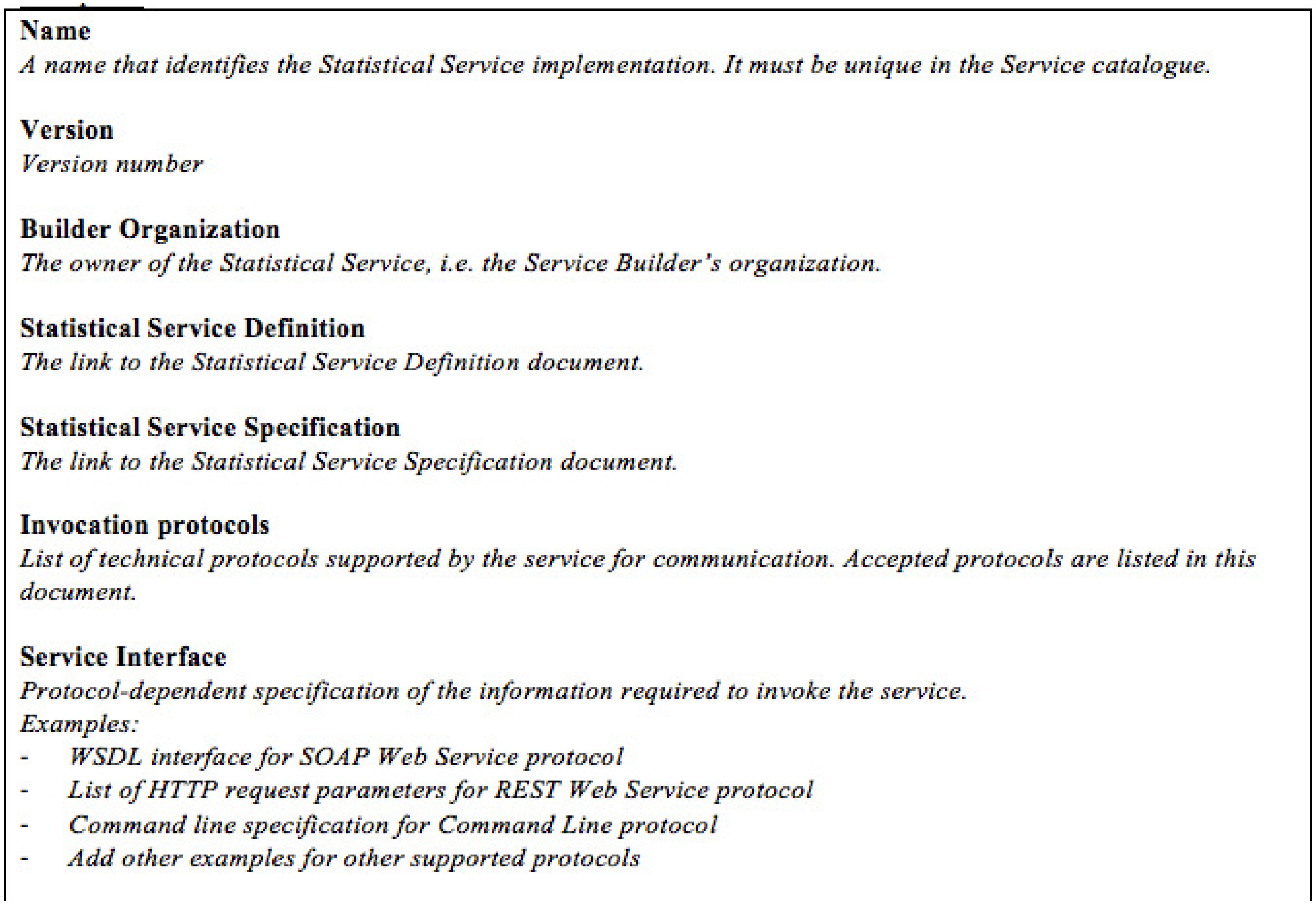
In GSIM terms, the inputs to this service are … …
Describe specific inputs in terms to GSIM implementation

Output Message

The outputs of the service are ……
Describe specific outputs in terms to GSIM implementation

Applicable Methodologies
Describe the statistical methods that may be implemented in this Statistical Service

Statistical Service Implementation Template 1/2



Name

A name that identifies the Statistical Service implementation. It must be unique in the Service catalogue.

Version

Version number

Builder Organization

The owner of the Statistical Service, i.e. the Service Builder's organization.

Statistical Service Definition

The link to the Statistical Service Definition document.

Statistical Service Specification

The link to the Statistical Service Specification document.

Invocation protocols

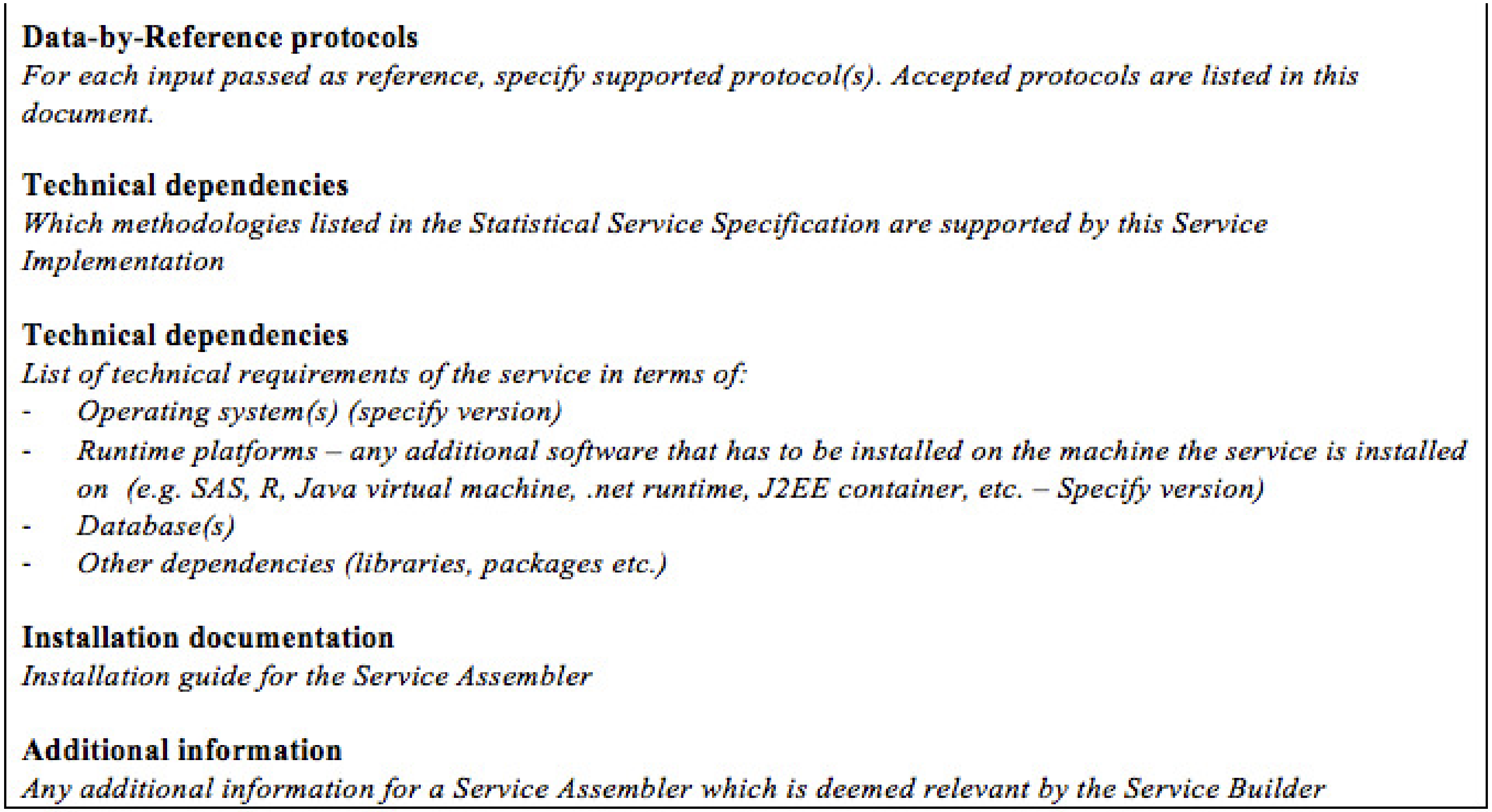
List of technical protocols supported by the service for communication. Accepted protocols are listed in this document.

Service Interface

Protocol-dependent specification of the information required to invoke the service.
Examples:

- *WSDL interface for SOAP Web Service protocol*
- *List of HTTP request parameters for REST Web Service protocol*
- *Command line specification for Command Line protocol*
- *Add other examples for other supported protocols*

Statistical Service Implementation Template 2/2



Data-by-Reference protocols

For each input passed as reference, specify supported protocol(s). Accepted protocols are listed in this document.

Technical dependencies

Which methodologies listed in the Statistical Service Specification are supported by this Service Implementation

Technical dependencies

List of technical requirements of the service in terms of:

- *Operating system(s) (specify version)*
- *Runtime platforms – any additional software that has to be installed on the machine the service is installed on (e.g. SAS, R, Java virtual machine, .net runtime, J2EE container, etc. – Specify version)*
- *Database(s)*
- *Other dependencies (libraries, packages etc.)*

Installation documentation

Installation guide for the Service Assembler

Additional information

Any additional information for a Service Assembler which is deemed relevant by the Service Builder

CSPA Roles

- The use of CSPA is based on functional roles, to be assigned within a statistical organization
  - **Investor**
  - **Designer**
  - **Service Builder**
  - **Service Assembler**
  - **Configurer**
  - **User**
- The functional roles are presented in the following in the form of user stories

Investor Story

- The Investor receives demand for a new data collection and needs to compare the cost of running a collection using traditional methods, with the cost of using a set of components as per CSPA.
- The Investor identifies existing Statistical Services to be used and identifies gaps where no Statistical Service already exists.
- The Investor weighs up creating a fully bespoke processing solution for the collection against having to build a new Statistical Service that fits into a set of existing Services. This would be done in consultation with other roles.

Designer Story

- The Designer has been given a set of business requirements at a high level from the Investor
  - what data is needed, what are the parameters of the process?
- In order to determine what functionality is available, the Designer will consider internal and external capabilities by searching in the Statistical Services Catalogue

Designer Story - 2

- When a possible internal candidate is found, there will be a decision made as to whether the existing functionality should be wrapped and exposed as a Statistical Service, or whether a new Statistical Service should be built.
- In the latter case, potential collaborators should be identified and negotiated with the Investor.

Designer Story - 3

- Once development has been decided, or in the case where existing functionality must be heavily modified, for each needed Statistical Service the Designer will specify the needed functionality to meet requirements.
- The Statistical Service is defined on a conceptual and logical level by a Service Definition using GSIM information objects and a Service Specification using GSIM implementation objects

Designer Story - 4

- On an implementation level, decisions must be made about how to realize needed functionality using technology approaches and these are documented in a Service implementation definition by the Builder.
- Service design across these levels includes information design, technology design, work-flow, interface design, and other relevant aspects.
- The Designer will address Service dependencies, Service contracts, extensibility, and all data, metadata and performance metrics.
- Capabilities for configuration by users will also be determined, as well as the degree of configuration to be implemented by the Builder.

Designer Story - 5

- An alternative scenario is one where Statistical Services are already available, having been found in the Statistical Services Catalogue, and meeting all identified requirements.
- In this case, the Designer specifies which Statistical Services are to be used, and specifies this for the Assembler to work with directly.

Builder Story

- The Service Builder receives Statistical Service definition and a Statistical Service specification from the Designer.
- The Statistical Service is then implemented by the Builder, by creating a Service implementation definition.
- The Builder will also implement features of the Service so that the Assembler can integrate it into the local environment so it can be deployed.
- The Builder tests the components to support the specified functionality.

Assembler Story

- The Assembler will take the Statistical Service and integrate it according to the understanding of the needed business process, as expressed in the design documentation.
- There are two cases for the Assembler
  - **Statistical Services entirely assembled within the local environment, which provides a high degree of confidence in their compatibility.**
  - **Use of external Statistical Services, which might require extension or modification. In this latter case, issues would be communicated to the Designer and Builder for further development.**

Configurer Story

- The Configurer takes the assembled process, and makes it suitable for use in the intended statistical domain.
- Parameters are specified according to the domain knowledge of the Configurer.
- Any issues with the assembled Service are communicated to the Designer, Builder, or Assembler.

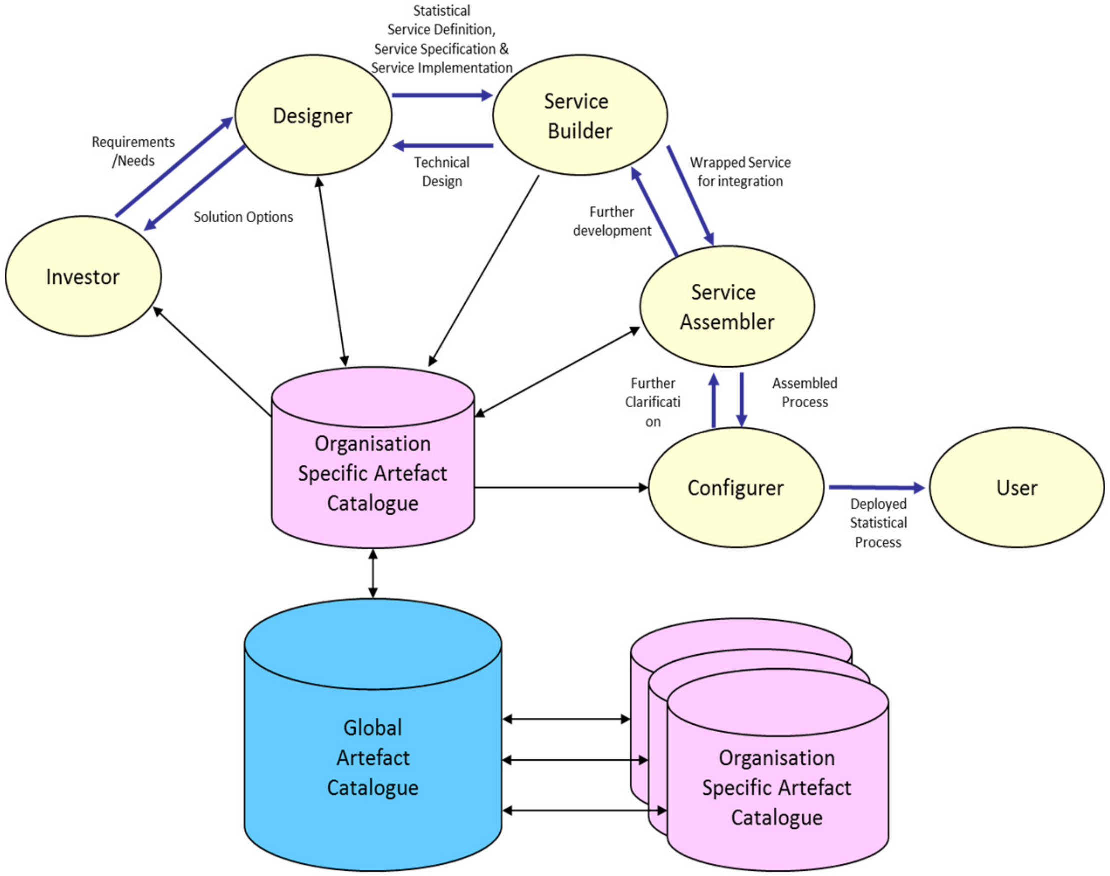
User Story

- There is no single user but a chain of users along the Statistical process.
- The user chain covers everyone from the designers of surveys, through the conduct of data collection operations, through to those who process the collected data.
- The User does not need to know where the data and metadata are stored - in particular the user does not need to actively manage how data flows between parts of the processing environment.

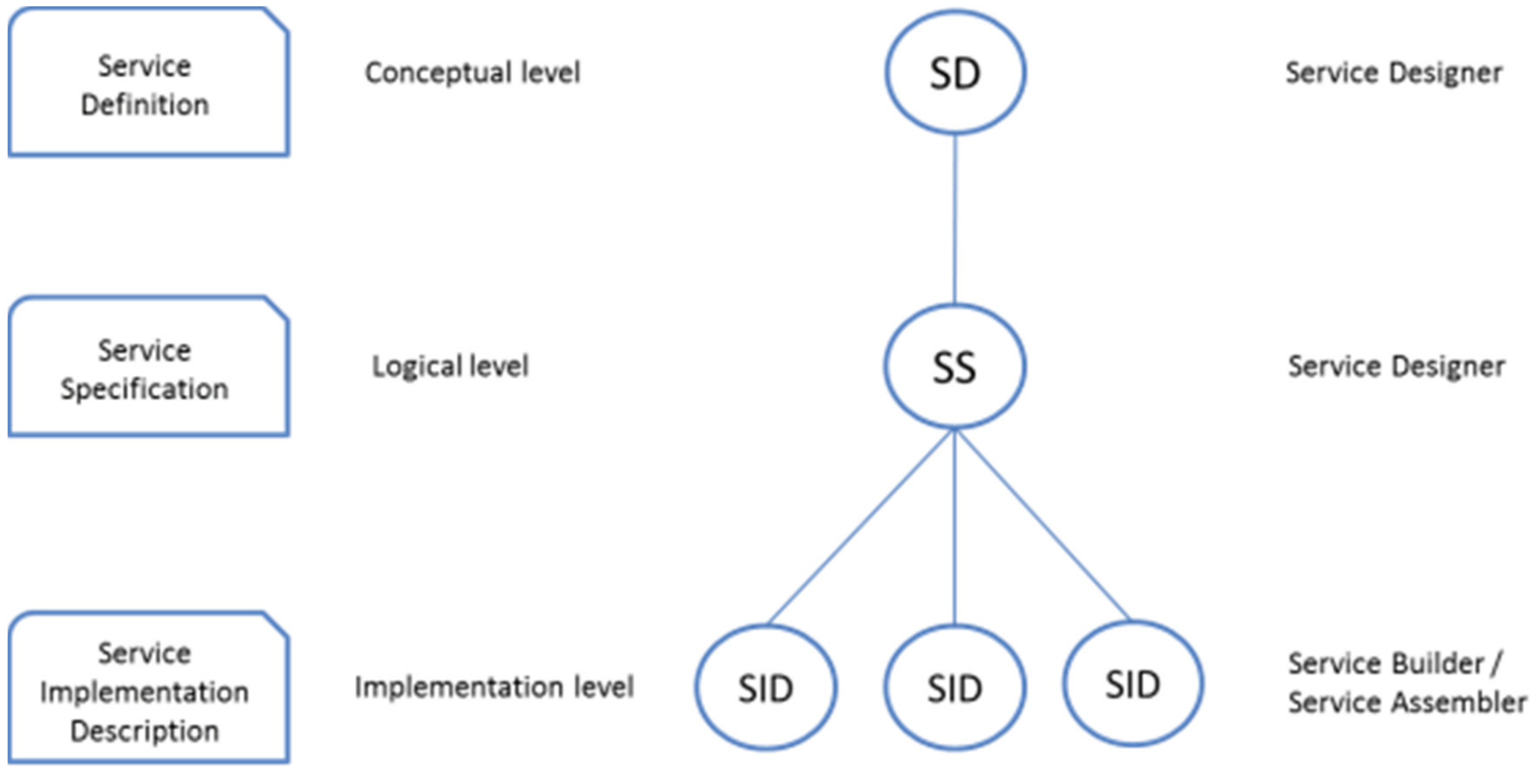
Catalogues

- Catalogues of reusable resources have a key role within CSPA.
- They provide lists and descriptions of standardized artefacts, and, where relevant, information on how to obtain and use them.
- The catalogues can be at many levels, from global to local. For example, it is envisaged that each statistical organization will have catalogues of processes, information objects and Statistical Services.

CSPA Roles Review



CSPA Roles Review



Service Definition — Conceptual level — SD — Service Designer

Service Specification — Logical level — SS — Service Designer

Service Implementation Description — Implementation level — SID SID SID — Service Builder / Service Assembler

European Commission

Architecture Patterns

- Describe a re-usable solution to certain classes of problems.
- They explain how, when and why Statistical Services can be used, as well as the impact of using them in that way.
- They help a Service Assembler to identify combinations that have been used successfully in the past.

Architecture Patterns: Request/Response

- The Request/Response pattern for activating services implies a rather fixed routing of messages between services.
- The integration infrastructural platform implementing the process “orchestrates” the routing and executing of services. This pattern leads to less flexibility and tighter coupling between services than the Publish/Subscribe pattern described below.

Architecture Patterns: Request/Response

- Example of how this pattern could be applied in collection.
- Each questionnaire is stored in an entity service.
- The entity service exposes the operation to get the questionnaire through a service call.
- The indicators are computed and stored and made available using an entity service call.

Architecture Patterns: Publish/Subscribe

- Could be considered an asynchronous version of the Request/Response pattern.
- As an event is generated by an event source and is sent to the processing middleware. It is not known which functionality is triggered next.
- In the Request/Response pattern, the concrete service call would have been made, but this is not the case for the Publish/Subscribe pattern. For this reason, the Publish/Subscribe pattern talks about "decoupling" rather than loose coupling.

Architecture Patterns: Publish/Subscribe

- Example of how this pattern could be applied in relation to collection
- Each questionnaire completed publishes an event that is available for subscribers downstream.
- Early indicators can be produced by processing collection events straight through aggregation.

Request/Response vs. Publish/Subscribe

- The Request/Response pattern can be used if
  - A functional style and sequential flow is required
  - It is known precisely which service interface should be called
- The Publish/Subscribe pattern can be used if:
  - All recipients that may be interested in the event should be notified
  - It is not exactly known which and how many recipients are interested in the event
  - It is not known how recipients respond to this event
  - Different recipients respond differently to the same event
  - Only one-way communication from the sender to the recipient is possible

Non-Functional Requirements

- In the context of CSPA, a non-functional requirement is a requirement that relates to the operation of a system.
- While functional requirements define what the services does (for example, error localization), the non-functional requirements describe a performance characteristic of a system (for example, authorization of who can access the resources and functions of the service).
- That is, non-functional requirements determine how a service behaves rather than what it should do.

Non-Functional Requirements

- Non-Functional Requirements are important to be captured in the design of the services.
  - They have a significant influence on the software architecture of a service.
- The Designer of a Statistical Service should identify the non-functional requirements that are relevant to that service when they are designing it.
- The implementation of a Statistical Service provides some functional value when assembled into a value chain within an organization.
- The non-functional requirements of a Statistical Service address other concerns or behaviors of the service such as performance, security, process metrics and error handling.

Non-Functional Requirements

- Multilingual Support
- Security
  - **Authentication and Authorization**
- Performance
- Process Metrics
- Error Handling

Implementing Protocols in a Statistical Service

- Stateless vs Stateful Services
  - **Statelessness: an individual service should be called with all the information it needs to complete - the service shouldn’t rely on previous execution.**
  - **Stateful: the Statistical Service needs to defer or persist some form of state.**
- Trade off flexibility of the service to fit into a particular process with scalability of the service.

Deferring State Information

- When designing and building a Statistical Service, the capability of deferring state information is important in two specific situations:
  - When the Statistical Service needs to be used in a publish\subscribe design pattern.
  - When the Statistical Service involves human-interaction and can therefore be considered long-running.
- Statistical Services with the capability of deferring state needs to provide an endpoint to support enquiries about the deferred state.

Invoking a Service

- A protocol is the technical implementation of a communication mechanism. It is used to invoke the Statistical Services deployed in a CSPA implementation.
- A Statistical Service Implementation Description must specify one or more protocols. These protocols are associated with the following aspects of a Statistical Service:
  1. making the Statistical Service reachable as an endpoint for invocation
  2. accessing data that is declared to be passed by reference in the Statistical Service Specification

Invoking a Service: Protocols

- The protocols for invoking service endpoints which are recommended by CSPA are:
  - SOAP Web Services – Service exposes a WSDL interface and is addressed by a http URI
  - REST Web Services - Service exposes a REST interface and is addressed by a http URI

Invoking a Service: Protocols

- There are also a number of other protocols which are acceptable. These are:
  - Microsoft Message Queue - Service is a MSMQ consumer
  - Java Messaging Service - Service is a JMS consumer
  - File-based invocation – the service is “invoked” when a file is placed at a known location which results in an OS-level trigger to the service; alternatively, the service can poll the location for arrival of “message” files and treat them as service invocations
  - Command line interface - Service is invoked by specifying a command line to be executed on operating system runtime accessible by the platform.

Invoking a Service: Protocols

- In some instances, existing tools support database access. If the database is involved in transfer (and not merely as a local state storage for the service), we recommend that the database access be mediated through the http: protocol access above.
- In general, the use of an “out of band” data transfer mechanism should be avoided wherever possible, and used only in circumstances involving the need to transfer large volumes of data.
  - Its addition adds increased coupling between the architecture and services, so its use must be managed carefully.

Transferring Large Data Volumes

- As a general rule, service invocation will involve the Statistical Service receiving a message via the organization-provided communication platform.
  - This message will contain the necessary information objects as well as the requested service.
- In certain circumstances, the service requires large data sets as inputs.
  - Examples of this could include administrative data files or large survey response files.
- The problem is similar to a “pass by value” situation in that the input data is passed to the service via in-message approaches.

Transferring Large Data Volumes

- There are a number of problems that can arise if we attempt to send these data sets via the messaging interface:
  - Dataset transfer time can be slow due to messaging overhead (packing / unpacking of data, message segmentation and reassembly, etc.)
  - Communications platform performance may degrade due to the load of transporting the messages between services
  - Service memory requirements can increase before required use (see State Deferral discussion
- In order to address this problem, we provide a “pass by reference” mechanism that avoids the need to use the communication platform messaging layer to transport these large data sets.

Transferring Large Data Volumes

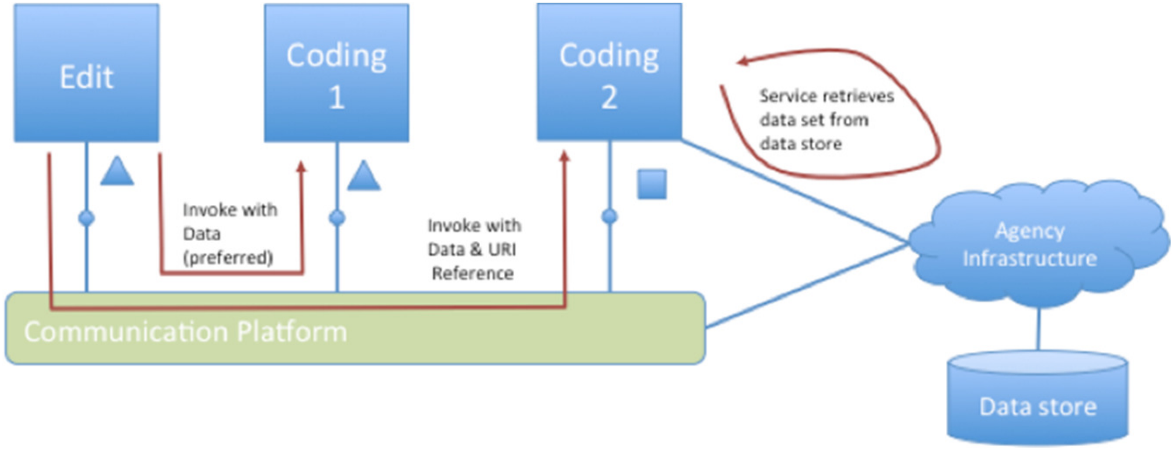
The pass-by-reference approach is as follows:

The data set being sent to the service is stored in a source location in a manner local to each organization – the location name is associated with a Uniform Resource Identifier (URI)

The service consumer invokes the requested service by sending it a message containing the URI for the dataset

The service provider receives the URI reference and when ready attempts to retrieve the dataset from repository or cache. If successful, it executes its service actions

Upon completion, it may update or place a resulting dataset (if relevant) in the repository or cache


Edit
Coding 1
Coding 2
Service retrieves data set from data store
Invoke with Data (preferred)
Invoke with Data & URI Reference
Agency Infrastructure
Communication Platform
Data store

Transferring Large Data Volumes

- The implementation of a data source is local to each organization and may be implemented as part of the communications platform. Organizations may choose to implement a utility service, a repository, a file cache, or some other mechanism. URI management is also a part of local operation.

- CSPA provides the following guidance for service input dataset retrieval protocols.

- Recommended protocols:
  - Simple http: file transfer from data source to the service logic (without additional protocols such as REST)

- Acceptable protocols:
  - ftp: file transfer from the data source to the service logic
  - Use of network file system services (such as SMB, NFS) with appropriate file reference

- Not Recommended:
  - Database retrieval using queries

TECHNOLOGY ARCHITECTURE

Technology Architecture

- *Technology Architecture (TA) describes the IT infrastructure required to support the deployment of business services, data services and applications services, including hardware, middleware, networks, platforms, etc*
- Within each statistical organization, there needs to be an infrastructural environment in which the generic services can be combined and configured to run as element of organization specific processes.
- This environment is not part of CSPA. CSPA assumes that each statistical organization has such an environment and makes statements about the characteristics and capabilities that such a platform must have in order to be able to accept and run Statistical Services that comply with CSPA.

Technology Architecture

- Platform for Service Communication
- Platform for Configuring and Controlling Services and Processes
- Platform for Reporting on Services and Processes

Platform for Service Communication

- A communication platform provides the capability for communication between Statistical Services.
- It enables inter-service communication while allowing Statistical Services to remain autonomous and adds additional capabilities for monitoring and orchestrating the information flow.

- There are multiple ways of establishing a communication platform. Examples of architectural components could be BPMS, ESB, Workflow Engines, Orchestration Engines, Message Queuing and Routing.

Platform for Configuring and Controlling Services and Processes

- Encompasses the functionalities and tools to support the management and maintenance of services metadata, artifacts and policies.
- Examples of how this mechanism could be achieved include Business Process Modelling System, Lifecycle Management, Service Monitoring and Management.

Platform for Reporting on Services and Processes

- Responsible for enabling real-time monitoring and near-real-time presentation of user defined business key performance indicators (KPIs).
- Examples of how this mechanism could be achieved are Static Dashboard or Business Activity Monitoring (also generates alerts and notifications to user when these KPIs cross specified thresholds).

Communication Platform

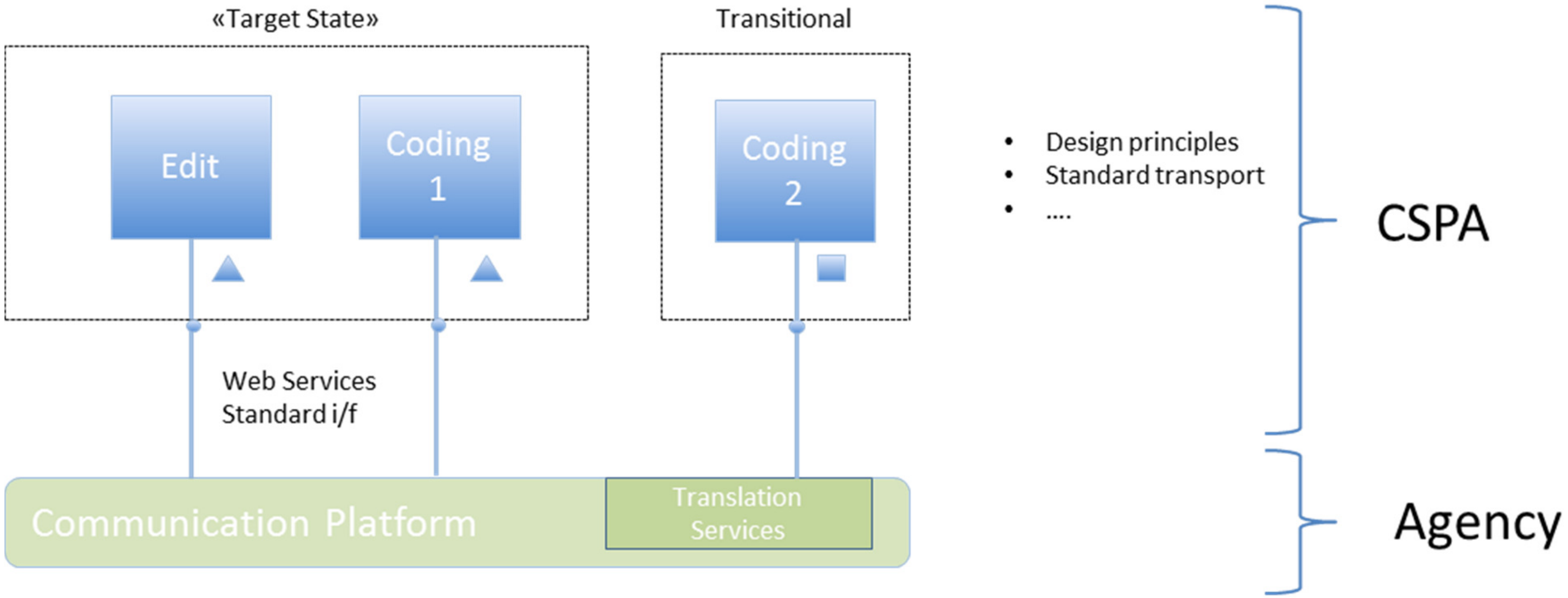
- CSPA does not specify how organizations will coordinate the use of Statistical Services to implement a wider business process.
- Organizations will need a technology solution to support communication between Statistical Services since the Statistical Services are not to talk directly to each other.
- Where the integration of Statistical Services is non-trivial a communications platform of some sort will usually be required.

Communication Platform Functions

- Orchestration
  - managing the sequence of flow of invocations of the Statistical Services;
- Error handling
  - where Statistical Service fail or where the output of services contain erroneous cases that require a different treatment;
- Message payload translation

- Auditing, Logging, Activity Monitoring;
- Performance Management;
- Security

Statistical Service Components and Communication Platforms



Statistical Service Components and Communication Platforms

- It is important to state that CSPA does not prescribe the capabilities and architecture of the underlying Communications Platform
- It instead assumes that an organization’s Assemblers and Configurers will be responsible for addressing how the platform supports the use of CSPA-compliant Statistical Services.
- This allows CSPA and its Statistical Services to be used by the widest possible community amongst statistical organizations, all of who may be in different stages of development and modernization.