VII. CSPA 2.0 Roles involved in the production of CSPA Services

173. Using CSPA will create new functional roles within a statistical organization. These roles may already exist in some statistical organizations and may have particular people (for example, Chief Information Officer, Enterprise Architect) assigned to them. CSPA makes no recommendation about who in an organization should play these roles or the composition (for example the 'investor role' could be an executive team, an investment board, or an individual business owner) – this is left to each statistical organization to decide how the role is undertaken – this is left to each statistical organization to decide who performs that role. This section and Figure 11 explain the functional roles in the form of one possible user story.

**Figure 11: Roles in CSPA**

**Investor**

174. The investor role is focused on directing and financing the investments in new statistical processes that are needed to produce new statistical products. Investors commission designing, building and running statistical processes. The challenge most executives of statistical organisations are facing is to manage the ever increasing demands for faster change with tight and often diminishing budgets.

175. This leads to a need for investments to be fit for purpose with the lowest possible Total Cost of Ownership (TCO) and a reasonably short time-to-market. Reusing already available solutions is a way to lower both costs and development time. But developing reusable components (services) incurs up-front development costs. The Investor needs to weigh up and decide on the alternatives. The Investor needs the Designer to design and present such alternatives.

176. International collaboration helps spread the costs and lower the overall TCO for the community and for the individual organisations. Investors therefore should pool their requirements and resources.

**Designer**

177. The Designer has been given a set of high level business requirements from the Investor, describing what data is needed, and some parameters of the process. The Designer identifies the data already available as input for the process, determines what data still needs to be collected and determines the steps needed to produce the required output from that collective input. The Designer then assesses what solutions (services) are available to actually realize this design. In order to determine what functionality is available, the Designer will consider internal and external capabilities. This will involve a search of the CSPA Catalogue(s), to determine whether there are internal or external candidates. The catalogue shows several levels of reuse: services already provided as a running service by a Service Provider (internal or external), services available as software from an internal or external source, solutions available only as a tool, not (yet) service ready, and services available as a conceptual specification only. When a possible internal candidate is found that is not yet a CSPA service, a decision is needed as to whether the existing functionality should be wrapped and exposed as a Statistical Service, or whether a new Statistical Service should be built.
178. Once it has been established that no suitable existing solution can be found, or in the case where existing functionality must be heavily modified, the Designer will specify the functionality for each Statistical Service needed to meet requirements. Together, the Investor and the Designer make the trade-off: generic vs. specific, development cost vs. long-term benefit. The Designer needs to help the Investor to make the right decisions.

179. Another parameter in this decision-making is the availability of other parties willing and able to participate in the investment of designing and building a new service. Potential collaborators should be identified and negotiated with the Investor.

180. Whatever the outcome of the trade-offs, the Designer will define any new Statistical Services on a conceptual and logical level. The conceptual level consists of a Service Definition using GSIM information objects, the logical level is a Service Specification using GSIM objects as presented in Section V. Figure 8.

181. Service design across these levels includes information design, work-flow, (service) interface design, and other relevant aspects. The Designer will address Service dependencies, Service interface 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 Service Builder.

182. Eventually, after all services needed for the process are either found to be pre-existing or specified and built, the process as such can be specified in terms of those services, connecting inputs, outputs and triggers. The task of building this process is given to the Assembler.

Service Builder

183. The Service Builder receives a Statistical Service Definition and a Statistical Service Specification from the Designer. The Service Builder then implements the Service, i.e. realizes functionality as specified in Definition and Specifications in software. The Service Builder will also implement features of the Service so that the Assembler and Configurer can customize the behaviour to fit the process or statistical domain. In addition, the Service Builder may provide features for the Service Provider to integrate it into the technology environment.

184. On an implementation level, decisions must be made about how to realize the functionality specified using various available technology approaches. The Service Builder will document the design decisions made, together with installation and operational instructions in a Service Implementation Description. The Service Builder tests the service to ensure that it supports the specified functionality. Test results may also be provided in the documentation. Finally, the Service Builder makes both software and documentation available through the Service Catalogue, locally and globally. The result is a software package that can be deployed onto some infrastructure and operated by a Service provider.

Service Provider

185. The Service Provider actually delivers the service so that it can be used in the execution of statistical processes. This means implementing the software as created by the Service Builder in a technical environment, drawing up Service Level Agreements and providing the service to Assemblers, Configurers and Users.

186. The Service Provider may limit the service to an IT-service only, providing the Software as a Service (SaaS). For a fully automatic service, this is the only scenario. But for a service that is automated only partially, the Service provider may decide to provide also the staffing, turning the SaaS into a full business service. However, if the Service Provider only provides the SaaS in this case, the Assembler (or ultimately the process owner) will need to find solutions for the staffing of the manual part of the service process.

187. There are two cases for the Service provider: one where the Statistical Service can be deployed within the local environment without any modifications, which provides a high degree of confidence in their compatibility. The second scenario involves the use of an external Statistical Service, which might require extension or modification to fit the local technical environment. In this latter case, issues would be communicated to the Service Builder and possibly Designer for adaptation or further development.

188. Note that Services and Service Providers can be both local and external to the organisation owning and operating the process that is using that service.

Environment Provider

189. The Environment Provider role is tasked with creating the communication infrastructure that drives the services during execution of the statistical process. The scope of this task is restricted to the local organization. Depending on the local requirements, the Environment should support one or even both patterns described in the chapter on Application Architecture. Also, the environment must support the Assembler and Configurer.

Assembler

190. The Assembler integrates the Statistical Services according to the requirements of the business process, as expressed in the (process) design documentation delivered by the Designer. The Assembler uses all levels of the interface specifications as provided by the Service Definition, the Service Specification and the Service Implementation Description. Depending on the pattern selected for the integration, the Assembler may build or configure a process orchestration, or set up mechanisms for an event-driven type of process. This might take place within a process management environment, as part of the (local) Communication Infrastructure, but could also be accomplished through a bespoke software layer.

Configurer
191. 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, Service Builder, or Assembler.

**User**

192. There is no single user involved with a particular Statistical Process. There are users both inside (executing manual parts of services, such as manual editing) and outside of (managing, controlling) the 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. The User should be able to perform his/her operations without experiencing a high degree of frustration or complexity.

**Example**

193. One example would be a case where the government legislates the requirement that some data collections must support people completing their reporting obligations on-line using eForms. Tasked by the Investor, the Designer reviews the impact of this change. To assess the impact the Designer needs to review what Statistical Services are required to implement the new Statistical collection, review the Statistical Service catalogue(s) for available solutions and identify any gaps.

194. The Designer may identify that there is a gap in the Services required to support rendering the design of an eForm for respondents to use: an eForm capability is missing from the current catalogue(s).

195. The Designer designs and presents alternative solutions, each with associated costs. The Investor makes the decision and once this decision as to how the data collection is to be run is made, the Designer will work out the details for the new process, including the specifications for any new services to be built.

196. The Service Builder will build the service solution, a Service provider will implement and provide the service and the Assembler will create the new process by integrating the new service into the (local) solution. Finally, the Configurer “tweaks” the process for a particular statistical survey.

197. With this, the machinery is in place and is “ready to roll”. Depending on the business environment for the process, the process owner (one of the types of users), or some preceding process triggers the process into action by providing the necessary input conditions. Users (“workers”) inside the process will be alerted as input arrives at the various process steps (services). For instance, once the survey has been designed and administered, the next User is alerted of the arrival of survey responses, including some data, which will be auto-coded, and other data, which needs manual coding. Once coded, the data would go through a series of edits, including data cleaning and validation, imputation, confidentialization, etc.