Distr.
GENERAL

WP.5
25 April 2012

ENGLISH ONLY

UNITED NATIONS ECONOMIC COMMISSION FOR EUROPE (UNECE)
CONFERENCE OF EUROPEAN STATISTICIANS

EUROPEAN COMMISSION
STATISTICAL OFFICE OF THE EUROPEAN UNION (EUROSTAT)

ORGANISATION FOR ECONOMIC COOPERATION AND DEVELOPMENT (OECD)
STATISTICS DIRECTORATE

Meeting on the Management of Statistical Information Systems (MSIS 2012)
(Washington, DC, 21-23 May 2012)

Topic (i): Architecture

Guidance for Statistical Services

Supporting Paper

Prepared by Jakob Engdahl Statistics Sweden, Sweden

I. Introduction

1. The *Generic Statistical Business Process Model (GSBPM)*[1] could be seen as the current building block for international collaboration regarding software for statistical production. The ongoing development of a *Generic Statistical Information Model (GSIM)*[2], as a corresponding information model, will result in the next building block to increase the possibilities of collaboration. The big question is: what should be built using these building blocks? The advancement of service oriented architecture is a clear indication that we should aim to build statistical services. This in turn raises the next big question: what should be encapsulated within a *Statistical Service*?

2. This paper summarizes Statistics Sweden's experience from the work of creating *Statistical Services* based on *Business Process Models* and *Business Information Models.* It describes how the transition from *Conceptual Models* to *System Logic* could be performed so that the result lives up to the principles of SOA in order to ensure that the *Statistical Services* benefits from the advantages of SOA.

II. Disclaimers and clarifications

A. Generic Statistical Information Model

3. The work of creating a GSIM-model is currently in its start-up phase and both the purpose and scope for GSIM could change, rendering some of the references to GSIM in this paper invalid. However, the need for modelling these *business information artefacts* remains valid, and the way in which this paper describes the usage of these *business information artefacts* also remains valid even though the development of GSIM could cause these to become out of scope for GSIM.

---

[1] http://www1.unece.org/stat/platform/display/metis/Generic+Statistical+Business+Process+Model
[2] http://www1.unece.org/stat/platform/display/metis/Generic+Statistical+Information+Model+(GSIM)

4. The way in which this paper describes the structure and usage of GSIM might be incorrect and based on assumptions more related to the author's previous discussion regarding the concept of a *Generic Statistical Business Information Model (GSBIM)*[3].

B. Data, Metadata and GSIM-artefacts

5. The concept of *metadata* could roughly be translated as "data about data". The traditional way of using the concept of *metadata* in the statistical production process is when referring to *information artefacts* such as *variables* and *classifications*. The scope for the metadata-concept has been debated time and time again and there are numerous attempts of classifying metadata objects. Distinguishing between *data* and *metadata* has been useful since it has helped us implement business principles like "*No data without metadata*".

6. In the current work being done to create a *Generic Statistical Information Model* the scope is much wider and includes both more traditional metadata objects like *Variable* and *Population* but also *Production-oriented* concepts like *Schedule and Rule*. Expanding the scope enough to describe the *information need* corresponding to the GSBPM will make the traditional classification of "*metadata* or *data"* less useful since everything related to the *information need* for a specific process which isn't *data* would become *metadata.* Referring to information objects like *Schedule* as being "data about data" is also a stretch since a *Schedule* is only a very indirect "data about data".

7. This paper describes the *information artefacts* needed for a specific GSBPM-process as GSIM-objects regardless if they would traditionally be classified as data, metadata or something else.

C. Survey, Product, Subject Area, Statistical Activity

8. There seems to be lacking a commonly used concept for describing the *context artefact* in which the GSBPM-process is performed. Performing the activities described in GSBPM generally results in one or more *statistical products*, but the result could also be a *register* which in turn is used as a data source in other *process instances* based on GSBPM.

9. Sometimes the concept of a *Statistical Activity* is used to describe something that performs the phases in GSBPM but this concept could easily be confused with the concept found when breaking down a specific GSBPM-process into smaller *activities*. In Statistics Sweden, we have adopted the concept of *Statistical Survey* for this in a broad sense. The GSBPM-process instance is in other words performed in the context of a *Statistical Survey.* The ongoing work with defining GSIM-objects might provide us with a better name for this concept, but this paper will use the concepts of *Statistical Survey* and *Survey Round* [4]

III. Expanding the Business Definition Models

A. Zachman Framework

10. The Zachman Framework[5] is an Enterprise Architecture framework which provides a structured way of viewing and defining an enterprise. Not a methodology in itself but instead a way of structuring the models that are used to describe an enterprise. The framework provides a number of *perspectives* or *levels* where the models on each level should provide insight to different roles in the organisation.

11. To define the ontology of an enterprise, yet another classification is necessary, corresponding to answering the interrogatives *who, where, when, why, how, and what.* This gives the other main dimension of Zachman's framework, which is concerned with organisational, spatial, temporal, motivational, process and

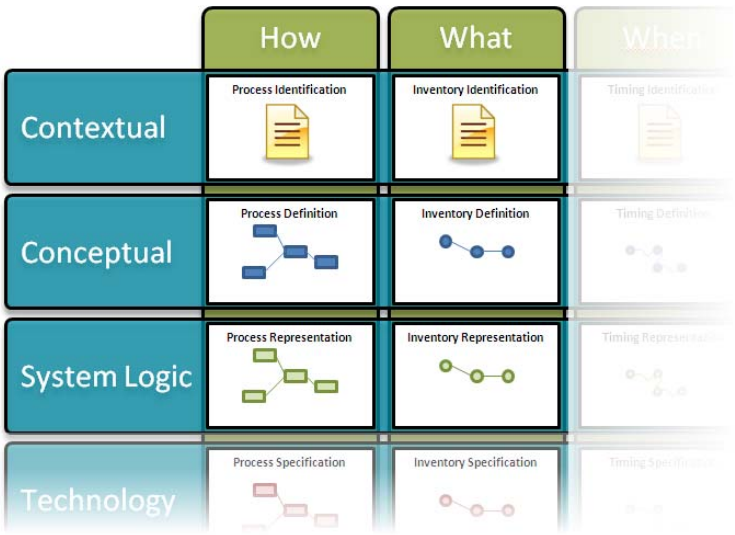
---

[3] Engdahl, J. An event-driven architecture for data collection http://www.unece.org/fileadmin/DAM/stats/documents/ece/ces/ge.50/2010/wp.9.e.pdf

[4] Fossan, Y., Holm, E. – Statistics Sweden - Business Information Model

content decompositions of an enterprise. By crossing the levels with responses to these interrogatives, one gets the rich structure of documents that, when in place, defines an Enterprise Architecture.[6]

12. Viewing the current international work being done to describe the *statistical business processes* in the *Generic Statistical Business Process Model (GSBPM)* and the *statistical information* in the *Generic Statistical Information Model (GSIM)* through the Zachman grid we gain insight into how this information can be used to reach an understanding of the structure of the models on *System Logic-level* that describe sharable software components.



[Picture 1: Fragment of the Zachman [7] Enterprise Architecture Grid]

B. Generic Statistical Business Process Model (GSBPM)

13. The GSBPM have been adopted by many *National Statistical Offices (NSO)* and describes the phases in a statistical production process and the activities that should be performed during a *Survey Round.* Viewed through the Zachman Framework, the GSBPM is fit to fill the *How*-column as it describes how different activities in the process should be performed. Since the GSBPM describes a list of sub-processes rather than describing the day to day flow of the production process, the GSBPM-model is more related to the *contextual* level in the Zachman grid than the *conceptual* level.

14. When adopting GSBPM it is important to view the model as an information bank, and it should only be used as a reference model. There is a gap between the GSBPM-model and the day to day activities being performed in the production process. The way to lessen this gap is not to continue the hierarchical breakdown structure to contain even more levels but instead to create a new set of process models which describes how the flow of the production process takes place. These models will describe the *How* on the *Conceptual* level in the Zachman grid and before we can begin international collaboration on *System Logic*-level and *Technology*-level we must first fill the gap on the *Conceptual*-level.

C. Generic Statistical Information Model (GSIM)

15. In June 2010, international collaboration was initialized with the goal to expand the current conceptual descriptions of the statistical production process with a corresponding conceptual description of

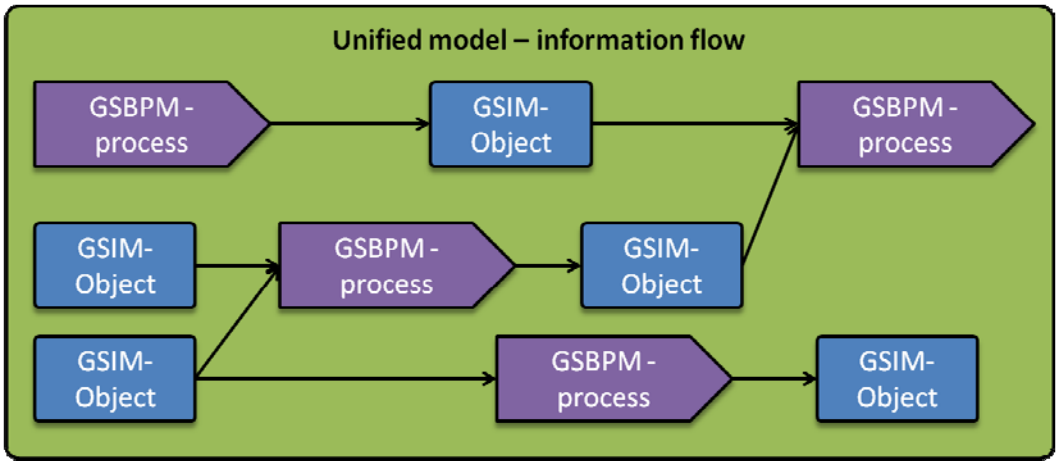
[5] http://en.wikipedia.org/wiki/Zachman_Framework
[6] Lorenc, B., Fossan, Y., Holm, E., Jansson, I. "Towards an Enterprise Architecture for an NSI"
[7] Zachman, J.A. (1987). "A Framework for Information Systems Architecture". IBM Systems Journal, Volume 26, Number 3.

the information objects required to drive the statistical production process. Viewed through the Zachman grid, the GSIM would answer the interrogative *What*.

16. Like GSBPM, GSIM is structured in a layered form, identifying a limited number of high-level information objects in the top levels, and adding more detailed objects in lower levels. The layered form of GSIM would put the high-level information models on a *Contextual* level and the lower levels on a Conceptual level.

D. Combining *How* and *What* – creating a *Unified Model*

17. The building blocks of *GSIM* and *GSBPM* could be used as separate models describing different aspects of the statistical production process, but by combining the information in these models we gain a *Unified Model* describing which processes create which type of information objects.



[Picture 2: Unified Model describing the information flow between GSBPM-processes]

18. The *Unified Model* gives us a better understanding of which of the *GSBPM-processes* that are directly related the *GSIM-objects* and this information gives us a better view of the value that each *GSBPM-process* adds in the *Value Chain* of a statistical production process.

E. Why, Who, Where, When - Zachman Framework

19. The Zachman framework also describes the need to model information regarding responsibility (*Who*), motivation (*Why*), timing (*When*) and distribution (*Where*) aspects on different levels. This could help each NSO to better understand their business but collaborating on these aspects on an international level will probably prove both difficult and meaningless. For example: the aspect of responsibility is highly connected to a NSO:s organisational structure and standardising on this aspect on an international level is not feasible.

20. The consequence when trying to make software sharable between NSO:s is to make the software components as independent as possible with regards to these aspects. In the example of the *responsibility* aspects, the effect on the software component would be to not require a too specific setup of roles for usage.

F. GSBPM-processes and Business Capabilities

21. Modelling *Business Capabilities*[8] differs somewhat from modelling *Business Processes*. A *Process* describes how a *Business Capability* is executed and a model describing a *Business Capability* is more focused on what the organization can accomplish. “At a high level, capabilities can appear identical to processes”[9], but in the *How* and *What* of a statistical production process, a *Business Capability* would

describe a set of activities in the *GSBPM-process* that are performed in order to either create, or add value to a *GSIM-object.*

22. In the context of international collaboration on creating sharable software components, this emphasises the value of the *Unified Model* since we only need to agree upon the structure of *Business Capabilities* - activities related to the creation of, or added value to, a *GSIM-object.*

G. The value of collaboration

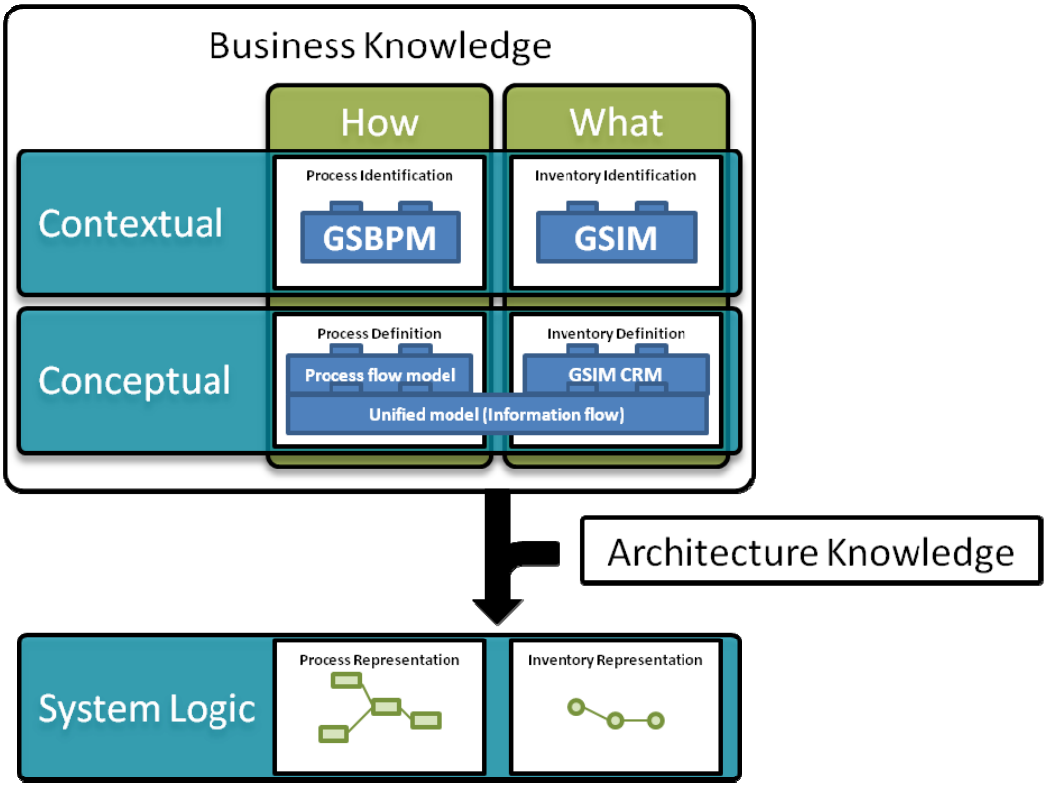
23. One value of collaborating internationally to describe different aspects of the statistical production process is that it helps us better understand our differences and similarities. Creating process- and information models on a conceptual level will provide a new level of insight, but the big value comes when we are able to create statistical software components which are sharable between NSO:s.

24. One of the experiences[10] that Statistics Sweden has drawn from implementing the GSBPM-process is the necessity of providing a usable set of tools to support the processes. Without supporting software, the implementation of GSBPM-processes will remain a *theoretical implementation*.

IV. Going from Business Concepts to System Logic

A. Business Knowledge and Architectural Knowledge

25. In order to achieve a *System Logic* level we first need to know how to make use of the information gathered in the models on the *conceptual* level and how this information can be translated into definitions of software components. In addition to knowledge about the business context these software components should support, we need to add knowledge about the target architecture for software components in a process oriented environment.



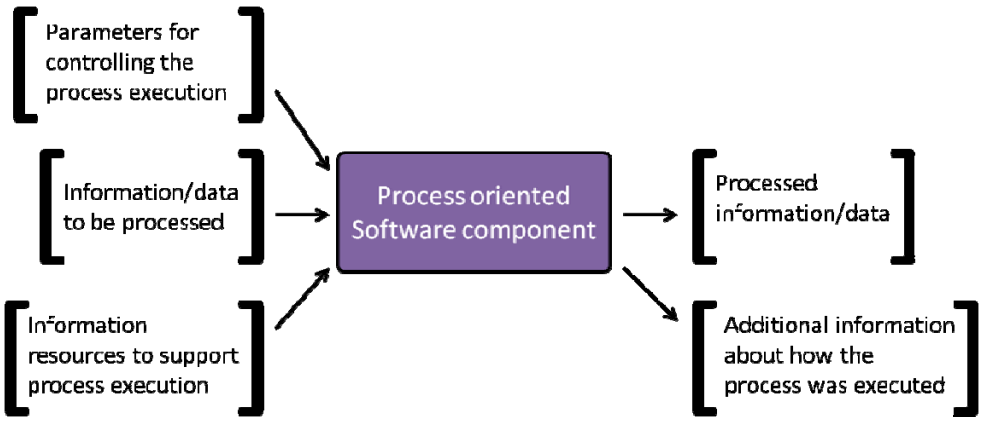
[Picture 3: Knowledge needed to reach System Logic-level in the Zachman grid]

26. Combining the knowledge of the business context with knowledge about the target architecture will enable us to make the transition to the *System Logic-level* where shareable software components can be modelled.

B. Information flow and software components

27. Without taking architectural knowledge into account, a highly simplified view of how the interaction between statistical software components and GSIM-objects can be created. The *information need* for a statistical software component, which is built to support a specific GSBPM-process, could be described using GSIM-objects. The input and output of each process oriented software component should therefore be described in terms of business information objects[11] and these can serve several purposes:

(a) Information controlling the behavior of the software component
(b) Information objects that the software module should add value to
(c) Other necessary information resources
(d) The result of process completion
(e) Information that further describes the result of the process execution



[Picture 4: Input and output of a process/process oriented software module]

28. In order to describe the mechanisms that could implement this information flow, we must first look at the principles related to the selected target architecture.

V. Service-Oriented Architecture

A. Architectural pattern

29. The architectural pattern of SOA builds on older concepts like *distributed computing* and *modular programming* and consists of "a set of principles and methodologies for designing and developing software in the form of interoperable services".[12]

30. A subject well written about but still sometimes forgotten is the confusion regarding SOA and Web Services. Now when the statistical community is looking into creating *Statistical Services* it is important that this venture isn't leading us into one of the big pitfalls of SOA. *Jason Bloomberg* summarized the confusion

---

[8] http://en.wikipedia.org/wiki/Capability_management_in_business
[9] Huenemann, T. http://www.ebizq.net/blogs/bpm_theory/2012/02/what-vs-how-part-2.php
[10] Bergdahl, M., and K. Blomqvist (2011). "National Implementation of the GSBPM – The Swedish Experience"

quite concise by writing: "Perhaps the most aggravating of misperceptions surrounding Service-Oriented Architecture (SOA) in the marketplace is that SOA and Web Services are the same thing. This point of confusion is unfortunately quite widespread, affecting architects and developers, consultants and vendors alike."[13]

31. If SOA is viewed as a technology evolution instead of an evolution in architecture, we risk maintaining concepts like *layers* from the architectural pattern of *distributed computing.* Adopting SOA by simply adding a *Service-Layer* will lead us into keeping the RPC-like pattern of *distributed computing* and we will end up just doing XML over HTTP which adds very little value and is terrible from a performance perspective.

B. Pitfall – Exposing existing tools as services

32. Another pitfall in the adoption of SOA involves encapsulating existing software into services in order to quickly provide the business with services. This "is in essence a technology first approach and is a recipe for disaster and/or serious over-engineering".[14] Encapsulating existing software should only be an option if the existing software is directly aligned to a specific sub-process in the GSBPM-process. Encapsulating generic statistical software like R, SAS or SPSS adds very little value since this is a technology first approach.

C. Pitfall – Translating GSIM-Objects into information services

33. The development of GSIM will bring along a new set of *nouns* and it might be tempting to begin to create *Services* corresponding with these and thereby encapsulating the *Business Information Objects* found in GSIM. "These are almost always the wrong place to start for identifying services in SOA"[15] and would ultimately create a dependency between services which would render the *Services* to become *hardcoded* to each NSO.

34. There are places in a NSO where the usage of *Information Services* could add value and data-warehouse solutions and archiving processes could be seen as *Information Services* which adds a *Temporal Capability* to the business as it provides the business with safe storage of the *Business Information Objects* until they are needed in the production process. This should however be seen as exceptions and not as the foundation for *Statistical Services.*

D. SOA Principles

35. Though, there are a number of pitfalls in adopting SOA which must be taken into account, there are also a number of principles in SOA which will give guidance when looking to identify *Statistical Services.* Thomas Erl published[16] the first list of eight specific service-oriented principles and even though other authors have added additional principles, the first list of eight principles gives us substantial information about the structure of a *Statistical Service.*

(a) Standardized Service Contract

36. "Services within the same service inventory are in compliance with the same contract design standards" – This principle preaches a "contract first" approach and in the context of reusable *Statistical Services* this means that the contracts of *Statistical Services* needs to build upon existing standards for communication. The best source of knowledge for creating the contracts for *Statistical Services* is our current knowledge of the *information need* for a specific process which is captured in GSIM. The conclusion is therefore that *Contracts* for *Statistical Services* should be based upon the business objects described in GSIM.

(b) Service Loose Coupling

---

[12] http://en.wikipedia.org/wiki/Service-oriented_architecture
[13] Bloomberg, J. http://www.zapthink.com/2007/06/20/divorcing-soa-and-web-services/
[14] Booch, G. (2006) "SOA Best Practices"

37. “Service contracts impose low consumer coupling requirements and are themselves decoupled from their surrounding environment” – This is one of the key principles for *Statistical Services* since any form of international reuse of a service is dependent on the service not requiring a specific set of other services in order to work. If a *Statistical Service* requires access to a number of *Information Services* in order to work, any exchange of that *Statistical Service* would require the receiving NSO to also implement the *Information Services* in the same structure as the NSO that created the *Statistical Service.*

38. To solve the *information need* for a *Statistical Service* the dependency should be reversed so that a *Statistical Service* instead of requesting required information instead is provided with the required information. This could be achieved either by using a *Publish/Subscribe* architectural pattern or by introducing a *communication platform* like an *Enterprise Service Bus.*

(c) Service Abstraction

39. “Service contracts only contain essential information and information about services is limited to what is published in service contracts” – The information reviled in a *Statistical Service* should only be derived from the information about the GSBPM-process it encapsulates (what capability the service provides) and the GSIM-objects (what information is needed and produced by the service). This principle helps maintainability of the *Statistical Service* since consumers of the *Service* doesn’t build dependencies based on the internal structure of the *Statistical Service.*

40. Following this principle also allows for the creator of the *Service* to create performance enhancing solutions without breaking *Service Contract.* The principle would allow a *Service* to implement an internal *Command-query responsibility segregation (CQRS)* or allow for the *Service* to store the GSIM-objects in whatever way best supports the execution of the *Service.*

(d) Service Reusability

41. “Services contain and express agnostic logic and can be positioned as reusable enterprise resources” – This principle promotes the reusability of a *Statistical Service* by requiring it to not be *hardcoded* for a specific *Survey*. For a *Statistical Service* to be reusable by more than one *Survey,* the specific configuration for process-execution must be applied in *run-time.*

42. How agnostic a specific *Statistical Service* should be is dependent on the possibility of describing the unique process execution context for the GSBPM-process that the *Statistical Service* supports, using GSIM-objects. An extreme level of abstraction could be found if a *Statistical Service* would encapsulate a commercial statistical software product like R and SAS, where the encapsulating *Service* would be so agnostic that an entire *Script* would have to be provided as a parameter. Encapsulating such a generic service adds little value. On the other side we would have a *Survey Specific Service* where all information would be hardcoded into the service. Since reuse of this service would be limited to a specific *Survey*, this *Statistical Service* would not meet the requirements for reusability.

(e) Service Autonomy

43. “Services exercise a high level of control over their underlying runtime execution environment” – This highly limits the possibility for a *Statistical Service* to access information stored in another database outside the scope of the S*ervice*. “A service composing another automatically loses autonomy”[17] means that a *Statistical Service* which require calls to *Information Services* breaks the principle of service Autonomy.

44. There is also the question about service autonomy with regards to physical execution environment. There is a common misconception that a *Statistical Service* that, for example, makes use of a database for storage must be agnostic to the database technology used. The technology for the database is part of the *Statistical Service* and the principle of service autonomy describes that a *Service* should not have dependencies outside the scope of the *Service.* The *Service* could in other words, require a specific technology for storage without breaking the principle of *Service Autonomy.*

45. Regardless of the principle, there could still be value found in creating some level of independence towards specific technologies to better support sharing of *Statistical Services.* The advancement of

Virtualisation Platforms will however increase the possibility of running *Statistical Services* that requires a specific physical execution platform.

(f) Service Statelessness

46. "Services minimize resource consumption by deferring the management of state information when necessary" – In order for a *Service* to meet the requirements of *Service Autonomy* and *Service Loose Coupling*, we introduced the solution that the S*ervice* is provided with necessary information instead of letting the *Service* make information requests to other *Services.* The different GSIM-objects that makes up the necessary information for performing a *Service Execution* isn't created at once. Instead each GSIM-object is created whenever the specific GSBPM-process responsible for creating the GSIM-object, is performed. When the *Statistical Service* is provided with these GSIM-objects, the *Statistical Service* must implement a capability of keeping these GSIM-objects until the information is required for the execution of the *Statistical Service.*

47. The principle doesn't require a *Statistical Service* to be stateless but instead requires the *Service* to implement the capability of storing away state information until it is needed. For *Statistical Services* this capability should be used to handle the temporal aspect of the creation of different types of GSIM-objects.

(g) Service Discoverability

48. "Services are supplemented with communicative meta data by which they can be effectively discovered and interpreted" – Borrowing from the principle "no data without metadata" this principle could simply be described as "no *Statistical Services* without metadata". Providing relevant metadata along with the *Statistical Service* will help the end user but also support the "*shareability"* of the *Statistical Service* between NSO:s. Relevant metadata for *Statistical Services* would at least include which GSBPM-process the *Statistical Service* supports together with information about which GSIM-objects are required for *Service Execution* and which type of GSIM-object the *Statistical Service* produces.

(h) Service Composability

49. "Services are effective composition participants, regardless of the size and complexity of the composition" – If *Statistical Services* were created that aligns based on another dimension than GSBPM, like *Organisational Structure*, there is a need for mapping *separation of concerns* to avoid creating overlapping *Statistical Services.* By aligning the *Statistical Services* to the GSBPM-processes, the work of *separation of concerns*, already implemented in the GSBPM-model can be inherited to the *System Logic-level*.

VI. Conclusion

50. The development of a Generic Statistical Information Model provides the statistical community with the next building block for international collaboration. The advancement of the architectural pattern of SOA makes this a good fit for creating sharable software components for the statistical production process. To ensure that the *Statistical Services* can benefit from the advantages of SOA, it is important that the structure of the *Statistical Services* follow the SOA-principles.

51. By viewing the *Business Context* described in GSBPM and GSIM through the principles for SOA, we see that *Statistical Services* should encapsulate fragments of the statistical production process and should communicate using GSIM-objects.

[17] Erl, T. SOA Principles and Patterns