_II. Information in a statistical organization

Note: To translate this paper into over 50 languages, please see the Automatic translation option at the top of the screen

II. Information in a statistical organization

A. Introduction

4. There is a widespread interest across statistical organizations in being able to trace how statistical information (for example, data and metadata) "flow" through statistical business processes (into processes and out of processes). Interested parties include broad statistical systems (like the European Statistical System), National Statistical Systems (both centralized and decentralized) and smaller task teams working inside National Statistical Offices.

5. In the description of the GSIM Business group, it is seen that GSIM covers the whole statistical process and is designed to support both current and new ways of producing statistics.

6. Achieving standards-based modernization of the production of official statistics places an emphasis on being able to share and reuse processes, methods, components and data repositories. Achieving reuse of processes, methods and components will require that process designers are readily able to discover what is available for reuse and whether it may be relevant to their particular purposes and needs. The case for reuse will be challenged if, in practice, discovering potentially reusable business resources, and assessing whether those resources are actually suitable for the designer's specific purpose, takes more time than creating new design elements.

7. GSIM was designed to enable an explicit separation between the design and execution of statistical processes. The description of the GSIM Production group shows how this has been modelled.

8. There is an increasing business need to record reliable, structured information about the processes used to produce specific statistical outputs. In order to maximize transparency and reproducibility of results, it is important for a statistical organization to understand the process and its inputs and outputs. The GSIM Concepts and Structures Groups contain the conceptual and structural metadata objects that are used as inputs and outputs in a statistical business process.

9. The GSIM Base Group consists of several objects that can be seen as the fundamental building blocks that support many of the other objects and relationships in the model. These objects form the nucleus for the application of GSIM objects. They provide features which are reusable by other objects to support horizontal functionality such as identity, versioning etc. For these reasons, many of these objects are rather abstract in nature.

10. Note: GSIM information objects have been given in italics in the descriptions that follow. The diagrams included in this section are stylized representations of the model. The colours of the boxes in diagrams represent which group the information object belongs to (Blue for Business Group, Red for Production Group, Green for Concepts Group, Yellow for Structures Group and Orange for the Base Group). In many cases there is more detail to be found in the UML. Detailed information on each information object in the model, including a glossary and UML class diagrams can be found in Annexes C and D of this document.

B. Business Group

11. The Business group is used to capture the designs and plans of *Statistical Programs*. This includes the identification of a *Statistical Need*, the *Acquisitio* , *Production* and *Dissemination Activities* that comprise the *Statistical Programs* and the evaluations of them.

12. An organization will react and change due to a variety of needs. In simple terms, these may be divided into at least two types of *Statistical Needs*: an *Inf ormation Request* and an *Environment Change*.

13. Where an organization receives an *Information Request* this will identify the information that a person or organization in the user community This community may include users within the organization as well as external to it. For example, a the team responsible for compiling National Accounts may need a new Statistical Activity to be initiated to produce new inputs to their compilation process. requires for a particular purpose. This request will commonly be defined in terms of a *Concept* or *Subject Field* that defines what the user wants to measure and the *Population* that the user wants data about.

14. When an *Information Request* is received it will be discussed and clarified with the user. This will be described by a *Process Step*. Once clarified, a search will be done to check if the data already exist. Discovering these *Data Sets* may be enabled by searching for *Concepts* and *Classifications*.

15. Where an organization identifies an *Environment Change* this indicates that there needs to be an externally motivated change. This may be specific to the organization in the form of reduced budget or new demands from stakeholders or may be a broader change such as the availability of new methodology or technology. A *Statistical Need* can be both internally and externally driven. For example, a statistical organization may realize that their existing *Products* and services must be improved. This may be in response to an *Assessment* of those *Products* and services.

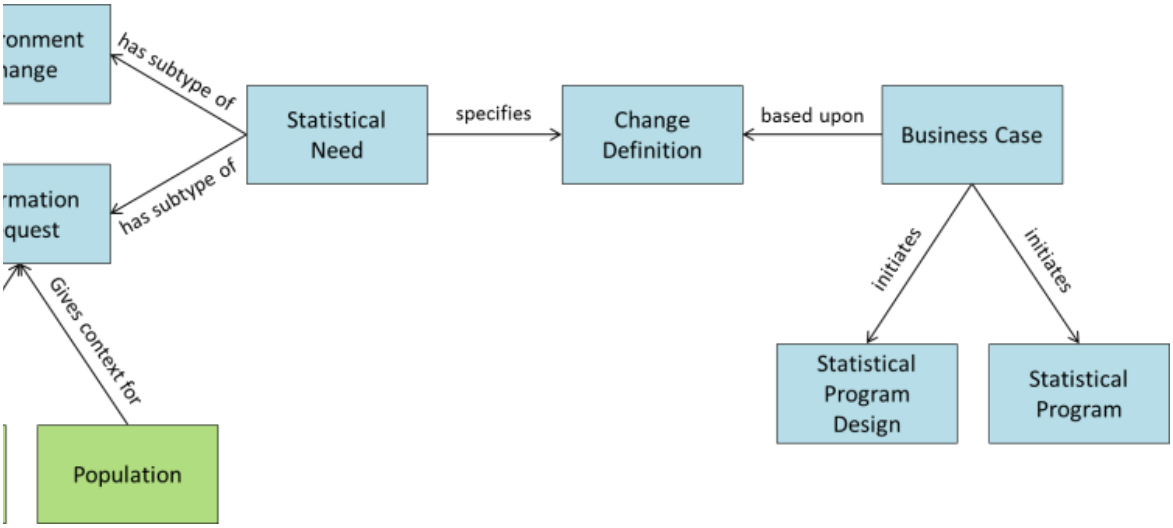

Figure 1. Statistical Need

16. As shown in Figure 1, once an organization has identified a *Statistical Need*, it will be further specified in the form of a *Change Definition*. This identifies the specific nature of the change in terms of its impacts on the organization or specific *Statistical Programs*. This *Change Definition* is used as an input into a *Business Case.* A successful outcome will either initiate a new *Statistical Program* or create a new *Statistical Program Design* that redefines the way an existing *Statistical Program* is carried out.

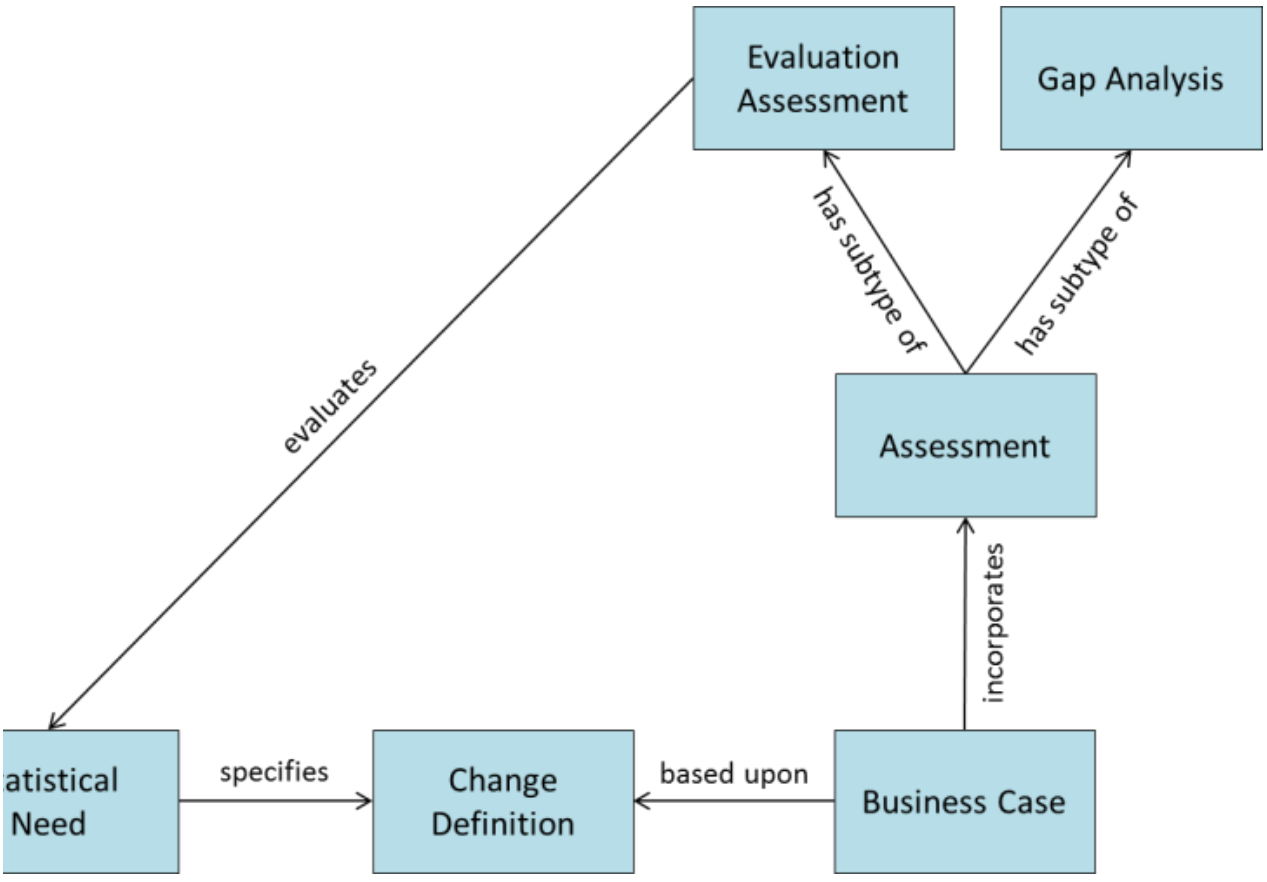

Figure 2. Evaluation

17. At any point in the statistical business process, an organization may undertake an evaluation to determine utility or effectiveness of the business process or its inputs and outputs. An *Assessment* will be undertaken to evaluate any resources, processes or outputs and may refer toany object described in the model.

18. An *Assessment* may be of several types depending on the purpose. A *Gap Analysis* may be undertaken often in the context of a *Business Case*. An *Ev aluation Assessment* is undertaken to determine whether a statistical output meets the need for which it was first created through analysis of:

(a) any information object that can be considered a *Process Output;* and

(b) in light of the original *Statistical Need*.
Statistical Program

19. A *Statistical Program* is the overarching, ongoing activity that an organization undertakes to produce statistics (for example, a retail trade survey). Each *Statistical Program* includes one or more *Statistical Program Cycles*. The *Statistical Program Cycle* is a repeating activity to produce statistics at a particular point in time (for example, the retail trade survey for March 2012).

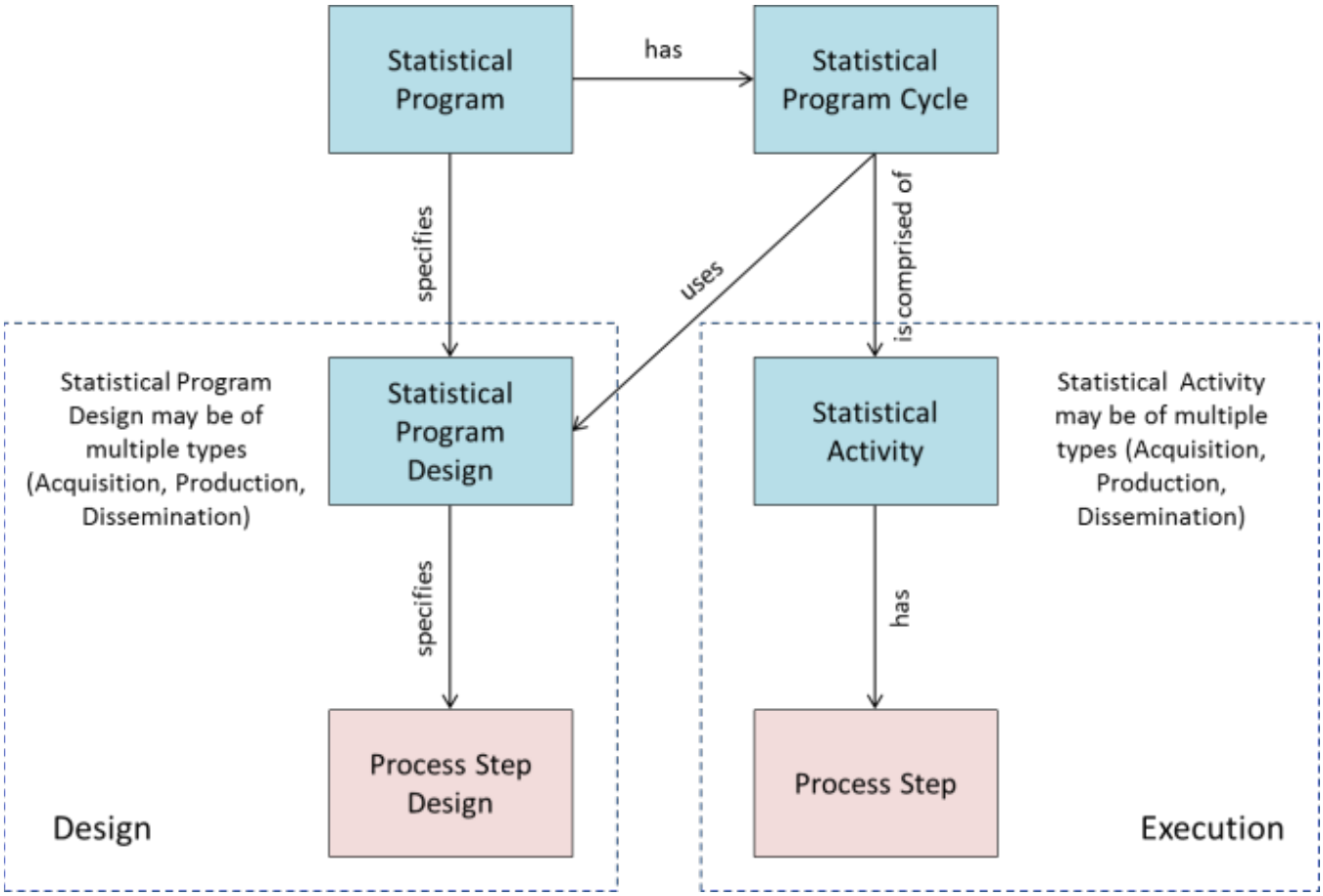

Figure 3. Statistical Programs

20. A *Statistical Program* (Figure 3) has an associated set of *Statistical Program Designs* that identify the methodology (the methods used to acquire, process and disseminate the data) used for the *Statistical Program*. Only one *Statistical Program Design* is valid for, and is identified as being used by, a particular *Statistical Program Cycle*. Changes to the methodology result in new *Statistical Program Designs* so over time each *Statistical Program* will have a series of designs that provide a history of changes to the *Statistical Program*. The *Statistical Program Design* identifies the set of processes that are intended to be used to undertake the activity (*Process Step Design*), the resources required for the processes and a description of the methodology and context.

21. Each *Statistical Program Cycle* consists of one or more *Statistical Activities*. A *Statistical Activity* is the set of executed processes and the actual resources required as inputs and produced as outputs. It is analogous to the *Statistical Program Design* but represents the execution rather than design. The same information that is identified in the *Statistical Program Design* and intended to be used to undertake an activity, is identified here as the actual information used. For example in the design, a dataset of a particular type may be identified as an input whereas in the *Statistical Activity* the filename and location of the actual input dataset would be identified.

22. The model identifies different types of activities that represent the major steps in the statistical production process (Figure 4). Three types have been specifically identified in the model but other types could be defined. The distinction between different types of activities and distinction of a *Statistical Activity* from a *Statistical Program Cycle* means that each iteration can be made up of multiple activities of the same or different types and these may or may not represent the sequence of collection through to dissemination. This model supports both the traditional approach of collecting data for a particular need, and the emerging and future approach of collecting data and producing new outputs based on existing data sources that are maintained and added to over time.

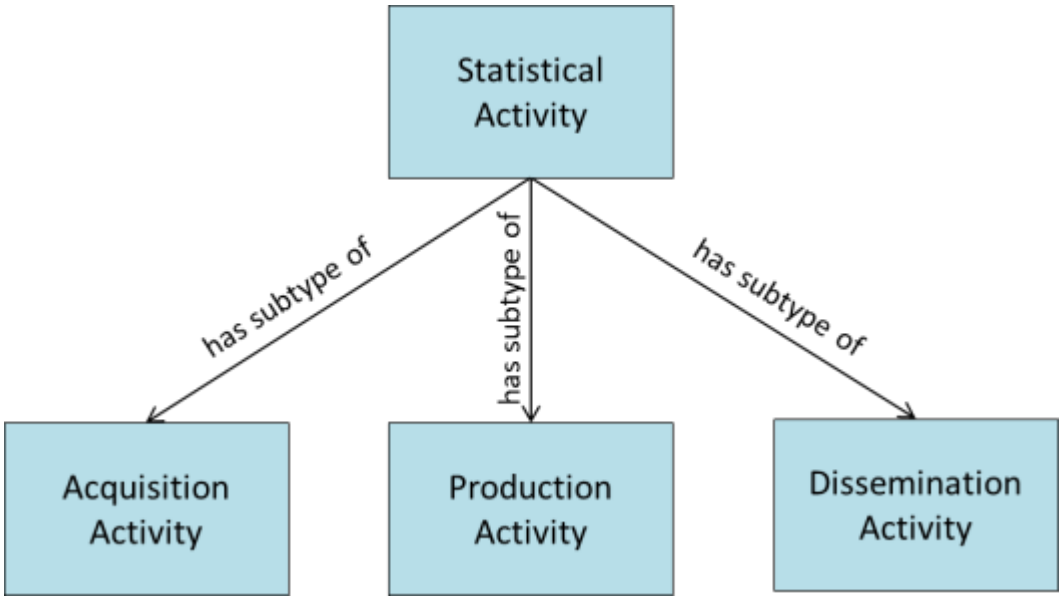

Figure 4. Statistical Activity

23. A possible future approach relates to a continuous collection process. In the age of 'big data', the cost of collecting and storing data (for example, a statistical register) is low. An organization can collect data on a continuous basis without a particular *Dissemination Activity, Product* or *Dissemination Service* in mind. In this case the organization has a *Statistical Program* with a *Statistical Program Cycle* that consists of an *Acquisition Activity* that gathers data and adds to a *Data Resource*. Any *Statistical Program* (consisting of only *Production* or *Dissemination Activities*) may then use this *Data Resource* in the future.

Acquisition Activity

24. For an activity where the purpose is to acquire data a *Collection Description* (Figure 5) provides a description of the activity and the associated contextual information. The *Acquisition Activity* identifies the means by which the data is collected and where it is collected from by identifying *a Data Channel*.

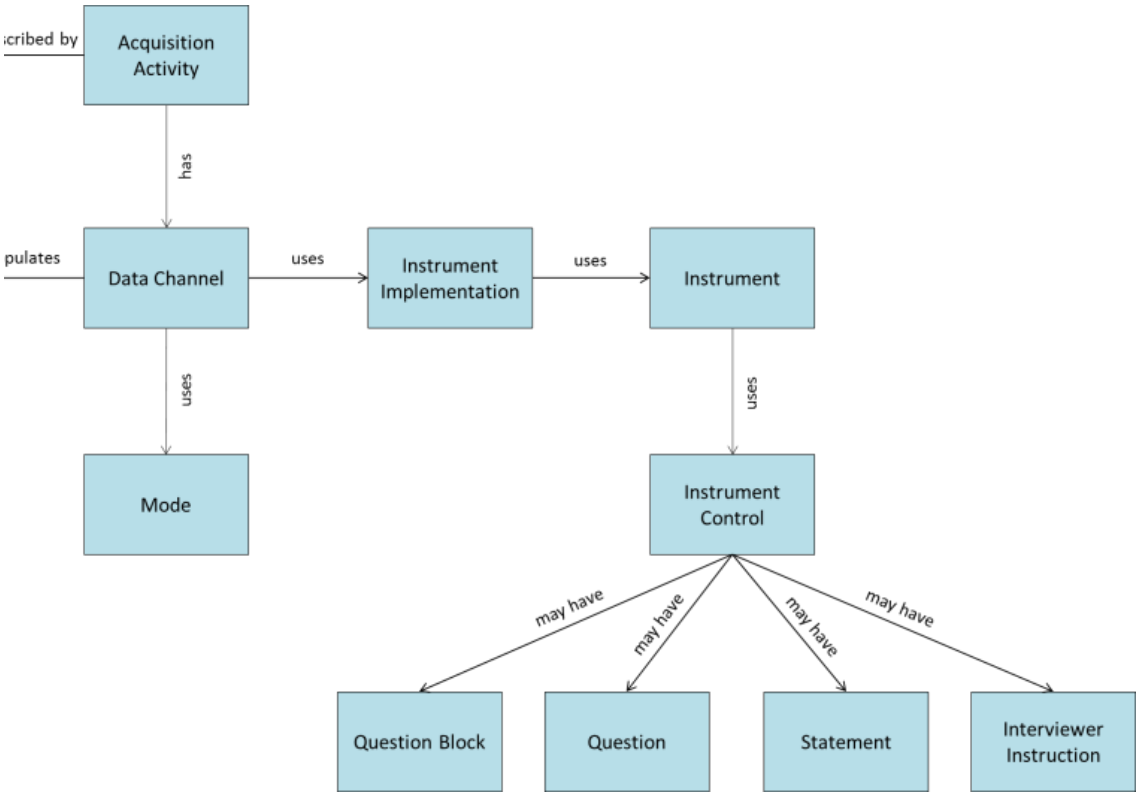

Figure 5. Acquisition Activity

25. A *Data Channel* identifies the *Instrument* used to collect data. An *Instrument* is the description of the tool that will be used to collect data. Examples of it may include a questionnaire or a set of requirements to develop software for gathering data. The *Instrument* includes an *Instrument Control* and may have *Question Blocks, Questions, Statements* and *Interviewer Instructions*.

26. Once the *Instrument* has been designed, it must be implemented in the form of one or more *Instrument Implementations*. These could be printed forms, software programs, etc. The *Data Channel* uses the *Instrument Implementation* to request data and describes the technique used to do it by means of a *Mode*. Once the *Data Channel* receives the data, it sends the data to an identified *Data Resource* (thus populating it with *Data Sets*).

27. The *Mode* represents the way the information collection process is going to be conducted and in this way, 'how' the *Data Channel* is going to be used, the following table (Table 1) represents some examples of *Data Channel*, *Instrument*, *Instrument Implementation* and *Mode*.

Table 1. Examples of Data Channel, Instrument, Instrument Implementation and Mode

Data Channel	*Instrument*	*Instrument Implementation*	*Mode*
Physical presence	Questionnaire	Paper Form	Traditional interview
Traditional mail			Self-administered
Direct deposit			
Computer		Software Program	CAPI interview
Phone			CATI interview
Internet			Self-administered
Data scanner device	Set of Requirements	Data Scanner Program	Data collector
Internet		Web Scraping Robot	Web queries
			Agents
Internet		Web Service Consumer Program	Applications interconnection
Secondary transfer of data		Data Transfer	Data Medium, File Transfer, Web Sphere Application

Production Activity

28. GSIM includes the notion of a *Production Activity*. More information about how GSIM expands on this activity can be found in the Production Group section.

Dissemination Activity

29. GSIM includes the notion of a *Dissemination Activity*. More information about how GSIM expands on this activity can be found in the Structures Group section.

C. Production Group

30. The Production group is used to describe each step in the statistical process, with a particular focus on describing the inputs and outputs of these steps. A business process can be specified in terms of:

- The *Process Steps* which need to be undertaken during that process, and
- The sequence in which *Process Steps* need to be undertaken during that process.

31. A *Statistical Activity* puts into effect a statistical business process which has been designed previously (it has a *Statistical Program Design*) and which spans one or more phases of the business process (for example, the Collect, Process, Analyze, and/or Disseminate phases of the GSBPM).

32. At the heart of the Production Group is the description of the *Process Steps* within the statistical business process and the use of statistical information as inputs to, and outputs from, each *Process Step*. Each P*rocess Step* can be as "large scale" or "small scale" as the designer of a particular business process chooses (see Figure 6). Steps can contain "sub-steps", those "sub-steps" can contain "sub-steps" within them and so on indefinitely.

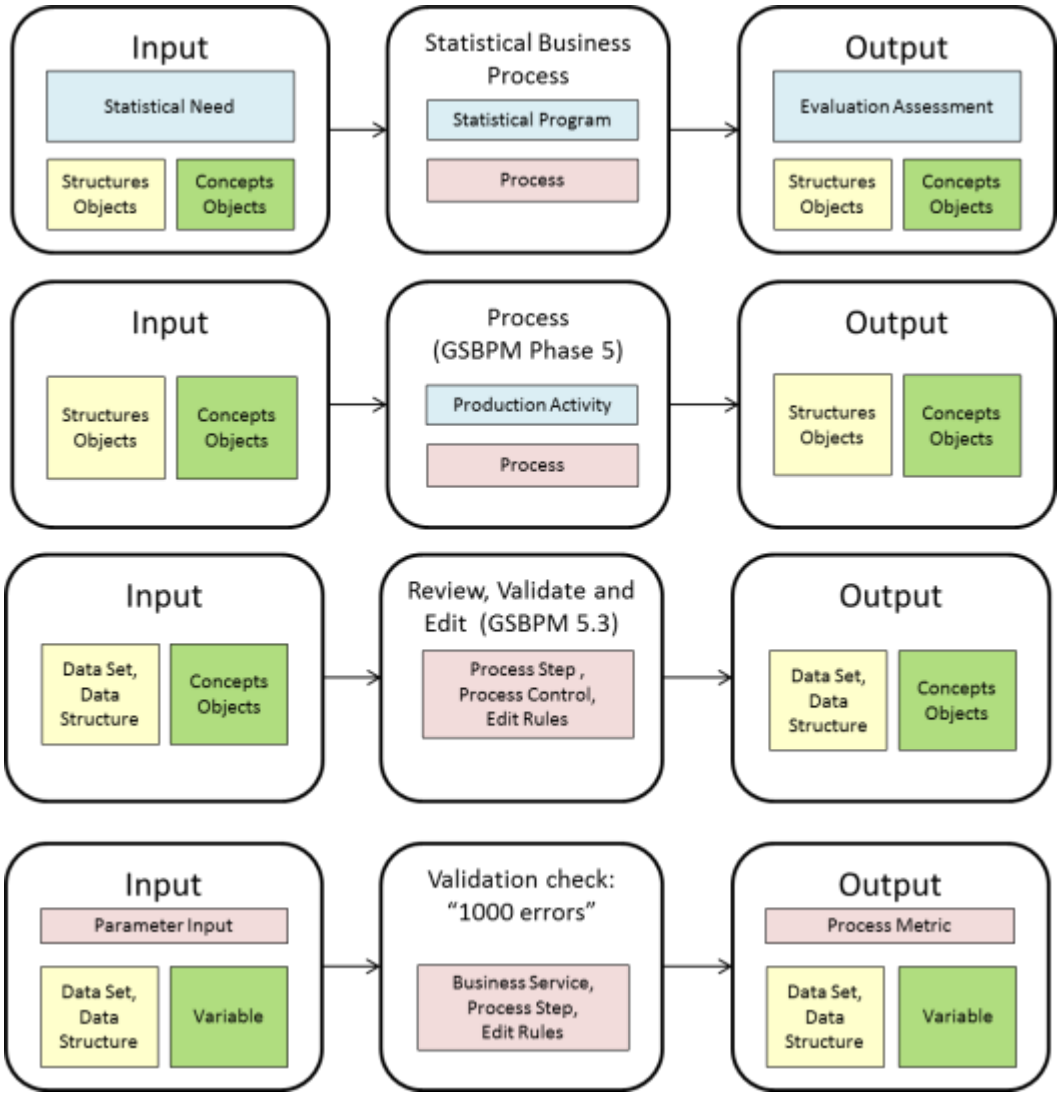

Figure 6. Process Steps can be as large or small as needed

33. In line with the GSIM design principle of separating design and production, the Production Group (see Figure 7) assumes that each *Process Step* will be designed during a design phase. Having divided a planned statistical business process into *Process Steps*, the next requirement is to specify a *Process Step Design* for each step. The *Process Step Design* identifies how each *Process Step* will be performed.

34. The sequencing of *Process Steps* within a business process is addressed through the concept of *Process Control.* When creating a *Process Step Design,* a *Process Control* that provides information on "what should happen next" is specified. Sometimes one *Process Step* will be followed by the same step under all circumstances. In such cases the *Process Control* simply records what *Process Step* comes next. However, sometimes there will be a choice of which *Process Step* will be executed next. In this case, the design of the *Process Control* will detail the set of possible "next steps" and the criteria to be applied in order to identify which *Process Step(s)* should be performed next.

35. During the production phase, as part of a *Statistical Activity, Process Steps* are executed in accordance with their design. An agent (person or system) initiates execution of the relevant *Process Steps* based on the following information:

- *Process Step Design* to determine how the current *Process Step* should be executed.
- *Process Control* to determine which *Process Step* to execute next.

36. A *Process Step Execution Record* should be recorded for each *Process Step* which is executed. The *Process Step Execution Record* is the information object which records the action. The action itself is a real world event, where *Process Step Execution Record* records that real world event.

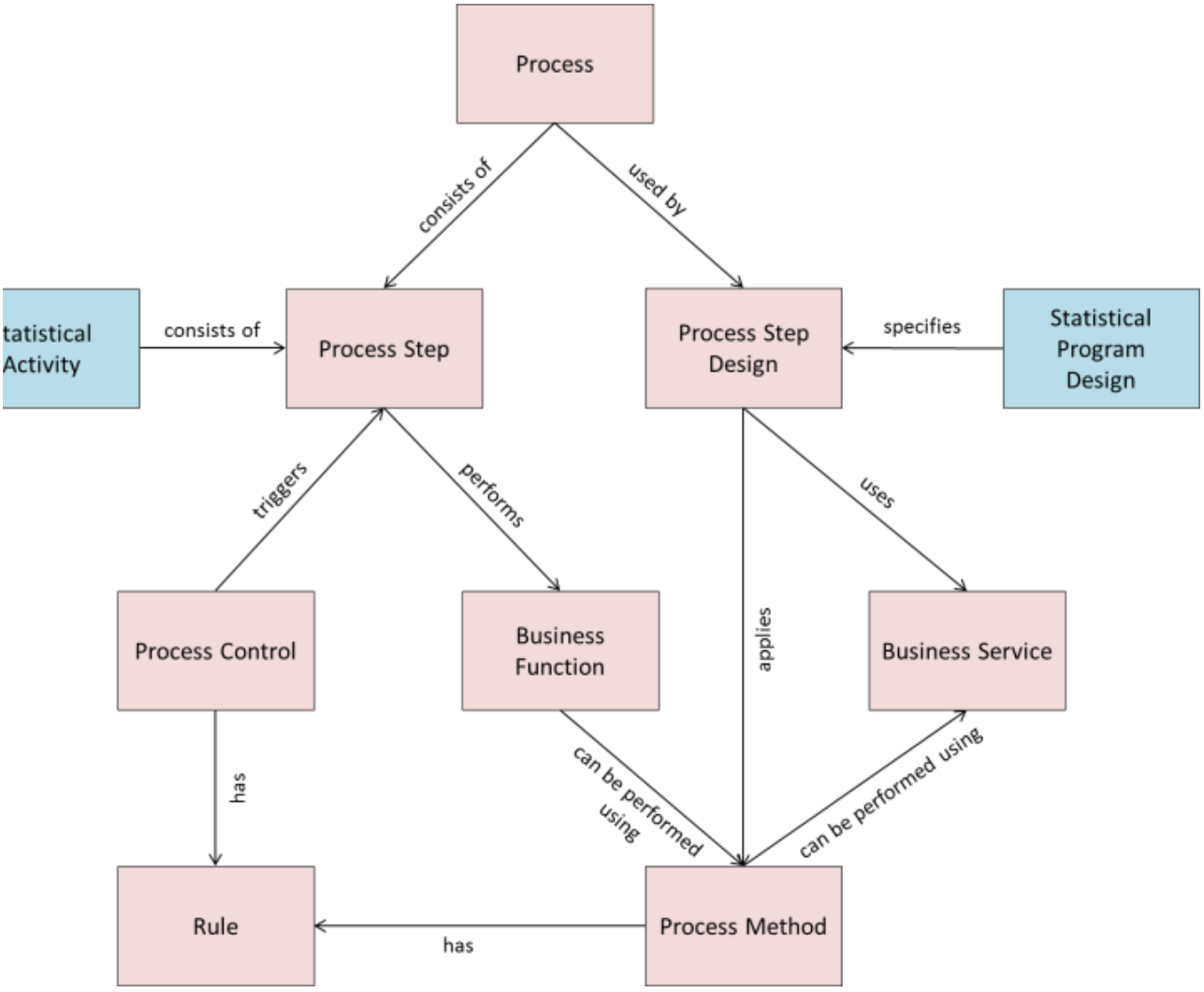

Figure 7. Simplified view of Production Group objects

37. As shown in Figure 7, a *Statistical Program Design* is associated with a top level *Process Step* whose *Process Step Design* contains all the sub-steps and process flows required to put that statistical program into effect. Each *Process Step* in a statistical business process has been included to serve some purpose. This is captured as the *Business Function* associated with the Process Step. The *Business Function*, for example, might be 'impute missing values in the data'.

38. The *Process Step Design* associated with that *Process Step* will then identify the *Process Method* that will be used to perform the *Business Function* associated with the *Process Step*. For example, if the *Business Function* is 'impute missing values in the data', the *Process Method* might be 'nearest neighbour imputation'.

39. A *Process Method* specifies the method to be used, and is associated with a set of *Rules* to be applied. For example, any use of the *Process Method* 'nearest neighbour imputation' will be associated with a (parameterized) *Rule* for determining the 'nearest neighbour'. In that example the *Rule* will be mathematical (for example, based on a formula). *Rules* can also be logical (for example, if Condition 1 is 'false' and Condition 2 is 'false' then set the 'requires imputation' flag to 'true', else set the 'requires imputation flag' to 'false').

40. At the time the *Process Step Design* is executed someone or something needs to apply the designated method and rules. The *Process Step Design* can designate the *Business Service* that will implement the *Process Method* at the time of execution. A *Business Service* represents a service delivered by a person or a piece of software. Putting a publication on the statistical institute's website or putting collected response forms in a shared data source for further processing are both examples of *Business Services*.

41. A *Process* consists of a set of *Process Steps,* including their associated process flow information. This enables the particular set of *Process Steps* to be named, and potentially catalogued and reused, as a *Process*. *Process Steps* need not be grouped into named *Processes* unless business benefits (for example, opportunities for reuse) are likely to result from doing so.

42. A *Statistical Activity* initiates the execution of a top level *Process Step* which will result in all sub-steps being executed which are relevant to that instance of the *Statistical Activity.* Executing the top level *Process Step* should start populating a *Process Step Execution Record* associated with that *Statistical Activity.*

43. The *Process Step Execution Record* (see Figure 9) will record the inputs provided when executing the top level *Process Step.* It will then record information which allows the actual flow of execution for that instance of the *Statistical Activity* to be traced. This includes recording the actual inputs to, and outputs from, each sub-step as well as the evaluation of each *Process Control* (which, in turn, determines the specific sequence of *Process Steps* performed during execution).

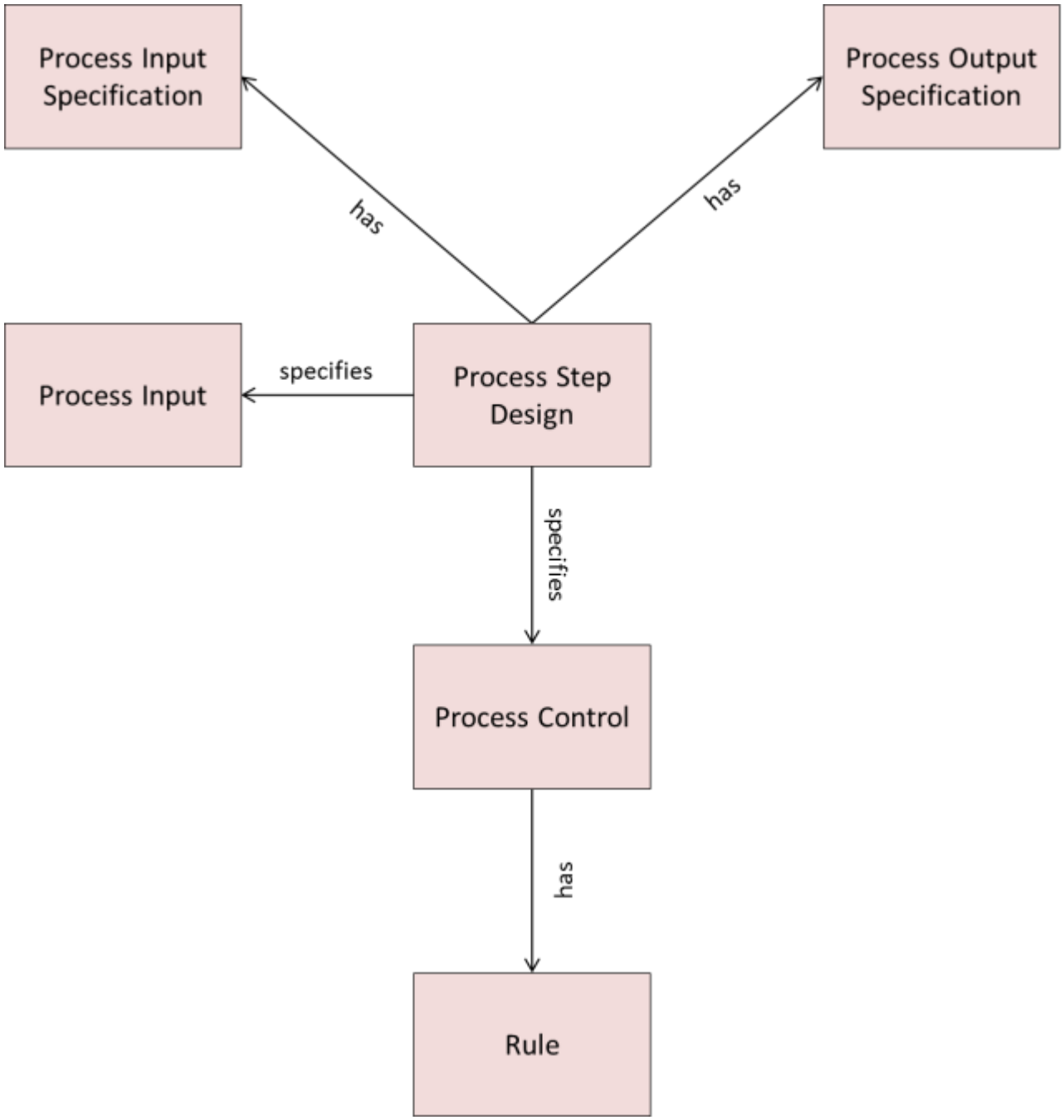

Figure 8. Process Step Design

Design

44. A *Process Step Design* (Figure 8) has a *Process Input Specification* that identifies the types of the *Process Inputs* required at the time of execution. An example might be a *Process Input Specification* that requires a *Dimensional Dataset* to be provided at the time of execution.

45. A *Process Step Design* may also identify *Process Inputs*. These refer to specific instances of inputs, rather than specifying a type of input. For example, a *Process Step Design* may specify that a particular *Code Set* will be used to provide a list of valid values.

46. *Process Input Specifications* and *Process Inputs* are often determined by the input requirements of the *Business Service*, *Process Method* and *Rules* associated with the *Process Step Design*.

47. *Process Output Specifications* play an analogous role to *Process Input Specifications* but describe the types of *Process Outputs* to be produced at the time of execution of the *Process Step*.

48. *Process Control* specifies what process flow should occur from one *Process Step* to the next at the time of execution. In some cases it may simply record the next *Process Step* to be executed on a fixed/constant basis. Alternatively, a *Process Control* may set out conditions to be evaluated at the time of execution to determine which *Process Step(s)* to execute next.

49. An example of the latter might be testing a *Process Output* against a quality criterion and initiating one course of action if the output meets the standard and another if it does not. It is not until the time of execution of the *Process Step* that it is possible to determine whether the standard has been met or not.

50. The specification and evaluation of conditional *Process Controls* refer to *Rules*. In the case of *Process Controls*, the *Rules* guide the process flow. (In the case of *Process Step Designs*, *Rules* guide the work done by the *Process Step* to produce *Process Outputs*).

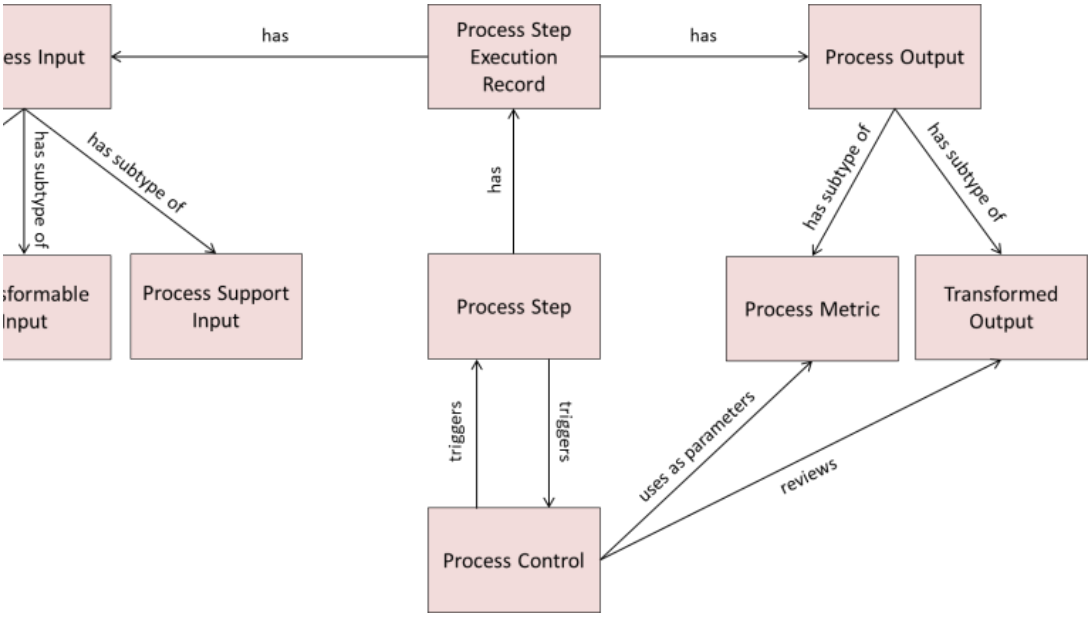

Figure 9. Process Step Execution

Execution

51. A *Process Step Execution Record* (Figure 9) records the execution of activities according to a *Process Step Design*.

52. Execution of a *Process Step* uses *Process Inputs* in accordance with the *Process Input Specification* specified in the *Process Step Design* (Figure 8).

53. When execution takes place a particular instance of a *Dimensional Dataset* (for example, "Turnover of retail trade establishments by employment size, industry class and state, for November 2012") will be provided as a *Process Input*. The identity (instance) of the particular *Dimensional Dataset* may be different for each example of execution. The specific *Process Inputs* associated with an instance of executing a *Process Step* are recorded in the *Process Step Execution Record*.

54. *Parameter Inputs* are a form of *Process Input* used to specify which configuration should be used for a specific execution of a *Process Step*. For example, a set of parameters like the statistical period concerned or a sample size.

55. A *Process Input* may be provided to a *Process Step* in order for the *Process Step* to 'add value' to that input by producing an output which represents a transformed version of the input. Such a *Process Input* is classed as a *Transformable Input*. Usually this represents the main dataflow within the statistical process (like microdata, aggregated data, and disseminated data). It is, in short, the data transformed by the statistical process.

56. A *Process Support Input* influences the work performed by the *Process Step*, and therefore influences its outcome, but does not correspond to a *Param eter Input* or a *Transformable Input.* Examples could include:

- A *Code List* which will be used to check whether the codes recorded in one dimension of a dataset are valid.
- An auxiliary *Data Set* which will influence imputation for, or editing of, a primary dataset which has been submitted to the process step as the *Tran sformable Input*.

57. A *Process Output* is any instance of an information object which is produced by a *Process Step* as a result of its execution. *Process Outputs* are subtyped as part of the *Process Output Specification*.

58. A *Transformed Output* is the result which provides the 'reason for existence' of the *Process Step*. If that output were no longer required then there would be no need for the *Process Step* in its current form. Typically, a *Transformed Output* produced by a particular *Process Step* will either be provided as a *Process Input* to a subsequent *Process Step* or it represents the final product from a statistical business process.

59. A *Process Metric* records information about the execution of a *Process Step*. For example, how long it took to complete execution of the *Process Step*; or what percentage of records in the *Transformable Input* were updated by the *Process Step* to produce the *Transformed Output.*

60. *Process Outputs* associated with execution of the current *Process Step* may be evaluated as part of *Process Control* in determining which process step to execute next.

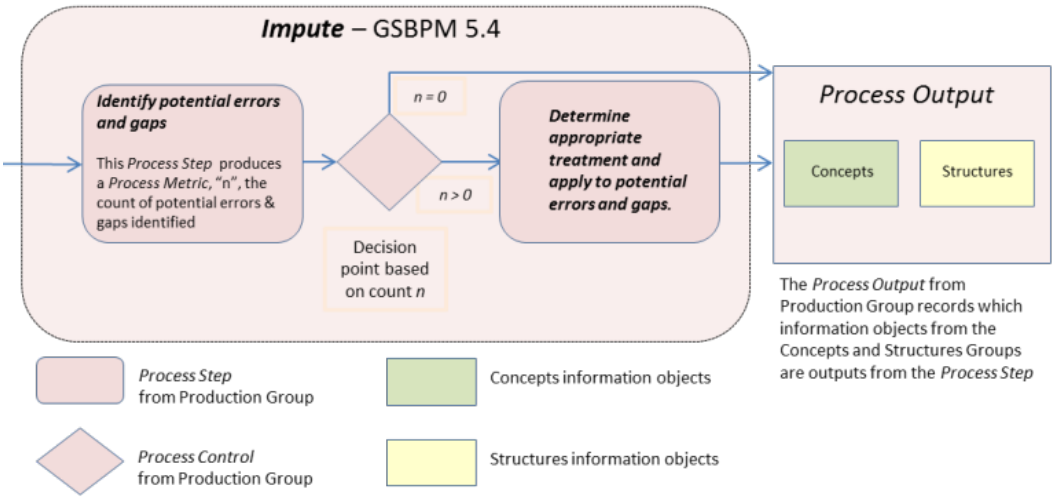

Figure 10. Conceptual and Structural information objects can be Process Inputs and Outputs

61. The execution of a *Process Step* will supply *Process Input*s and result in *Process Outputs*. The specific *Process Inputs* and *Process Outputs* associated with the particular execution will be recorded in the *Process Step Execution Record*. Through *Process Input Specification* and *Process Output Specification* the *Process Step Design* defines the types of *Process Inputs* to be supplied, and the types of *Process Output*s to be produced at the time of execution (See Figure 10). In many cases, these *Process Inputs* and *Outputs* are the conceptual and structural information objects that are described in the GSIM Concepts and Structures Groups (See Sections D and E). The same instance of an information object may perform different roles in different process steps.

D. Concepts Group

62. The GSIM Concepts Group contains sets of information objects that describe and define the terms used when talking about real-world phenomena that the statistics measure in their practical implementation.

63. The information objects in this group are used as *Process Inputs* and are often referred to in *Products* and *Representations* to provide information that helps users understand results.

64. At an abstract level, a *Concept* is defined in GSIM as 'unit of thought differentiated by characteristics'. *Concepts* are used in these situations:

(a) As a *Population.* To describe the set of objects it is wanted to obtain information about in a statistical survey. For example, the *Population* of adults in Netherlands.
(b) As a characteristic. A particular *Concept* about a *Population* is described by a *Variable*. The data are linked to a concept via a variable. For example, the *Concept* of gender in the *Population* of adults in Netherlands is collected by a *Variable*. At the representation level, there are data with *Codes*.
(c) As a *Category* to further define details about a *Concept*. For example, Male and Female for the *Concept* of Gender. *Codes* are linked to a *Category* via a *Classification Scheme*, for use within a *Classification*.

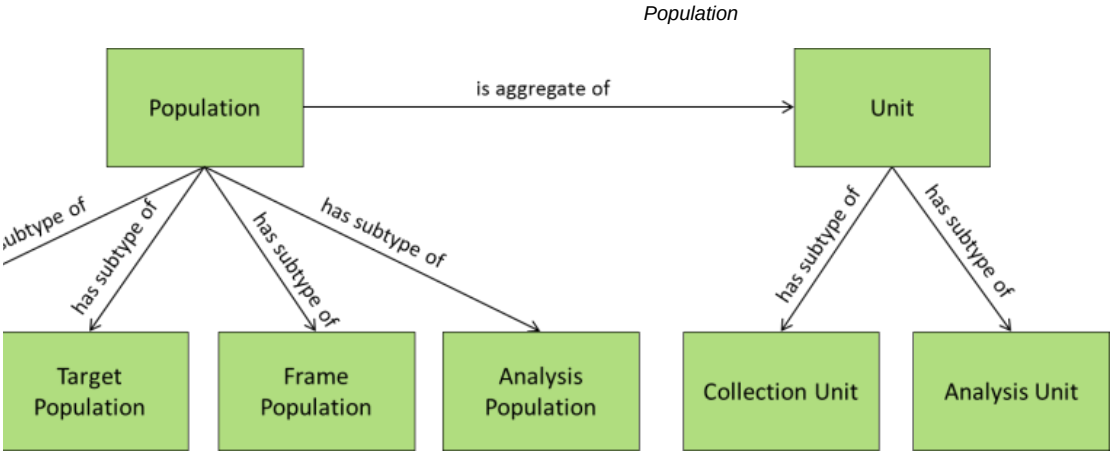

Figure 11. Populations and Units

65. As part of a *Statistical Activity* there is a *Population* (see Figure 11). There are several kinds of *Populations*: *Target*, *Survey*, *Frame*, and *Analysis*. The objects of interest are *Units* (for example, persons or businesses). Data are collected about *Units*. There are two kinds of *Unit* specified in the model. These are *Observation Unit* and *Analysis Unit*. A *Unit* is associated with a *Population*.

Variable

66. When used as part of a *Statistical Activity***,** a *Population* is associated with a characteristic. The association of *Population* and a *Concept* playing the role of a characteristic is called a *Variable* (see Figure 11). For example, if the *Population* is adults in Netherlands, then a relevant *Variable* might be educational attainment.

67. *Variable* (educational attainment of adults in Netherlands) does not include any information on how the resulting value may be represented. This information is in the *Represented Variable.* This distinction prevents the duplication of *Variable* information when what is being measured is the same but it is represented in a different manner. It promotes the reuse of a *Variable* definition.

68. A derived *Variable* is created by a *Process Step* that applies a *Process Method* to one or more *Transformable Inputs* (*Variables*). The *transformed* Output of the *Process Step* is the derived *Variable.* In GSIM, this is modelled in the Production Group (see Section C).

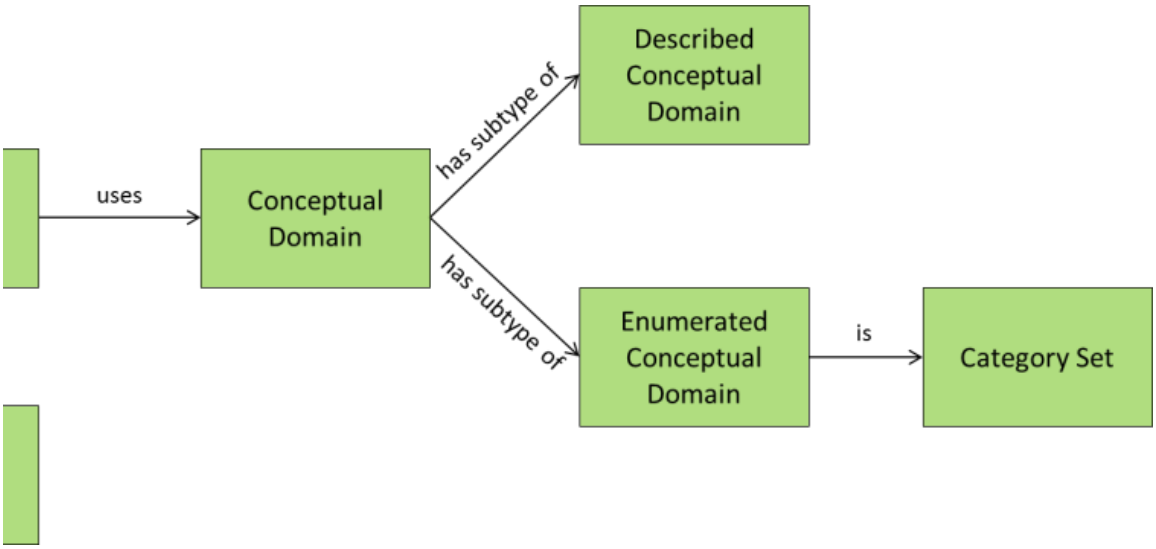

Figure 12. Variable

69. A *Conceptual Domain* is associated with *Variable*. It has two subtypes: *Described Conceptual Domain* and *Enumerated Conceptual Domain*. An *Enumerated Conceptual Domain*, in combination with a *Category Set* contains information on the semantics of the *Categories* used by the *Variable*.

Represented Variables

70.GSIM assists users in understanding both the meaning of the object and the concrete data-representation of the object. Accordingly, GSIM distinguishes between conceptual and representation levels in the model, to differentiate between the objects used to conceptually describe information, and those that are representational.

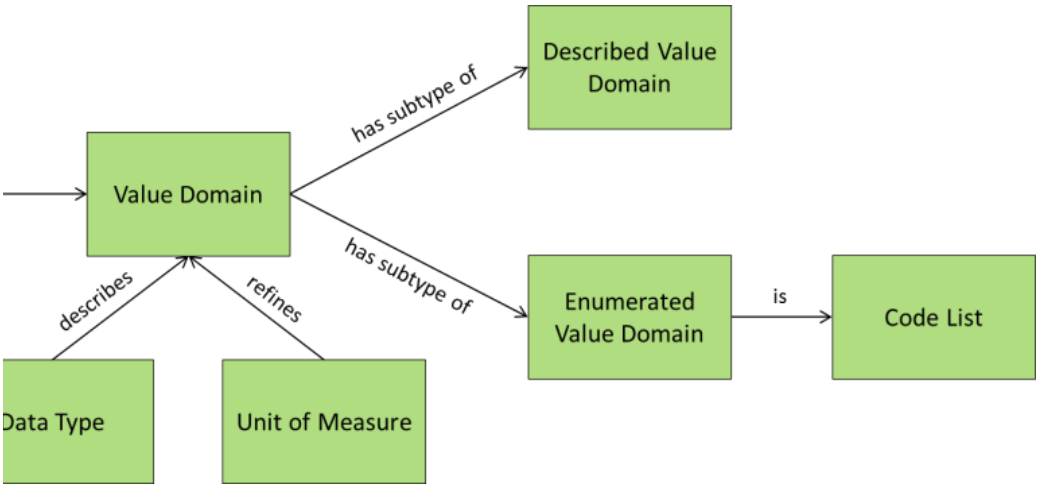

Figure 13. Represented Variable

71. The *Represented Variable* (see Figure 13) adds information that describes how the resulting values may be represented through association with a *Value Domain*. While *Conceptual Domains* are associated with a *Variable*, *Value Domains* are associated with a *Represented Variable*. These two domains are distinguished because GSIM wants to be able to talk about the semantic aspect (*Conceptual Domain*) separately to the representational aspect (*Value Domain*).

72. Both the *Enumerated Value Domain* and the *Described Value Domain* give information on how the *Represented Variable* is represented. The *Enumerated Value Domain* does this in combination with a *Code List*, while the *Described Value Domain* provides a definition of how to form the values, rather than explicitly listing them.

73.The *Value Domain* is defined by a *Data Type*. *Data Types* contain information on the allowed computations one may perform on the *Datum* (see Figure 15). For example, it is possible to distinguish between nominal-, ordinal-, interval-, and ratio-data as *Data Types*. Gender *Codes* lead to nominal statistical data, whereas age values lead to interval data.

74. A *Unit of Measure* refines the *Value Domain* . It is the entity by which some quantity is measured. Examples are Tonnes, Count of_, and Dollars.

Instance Variable

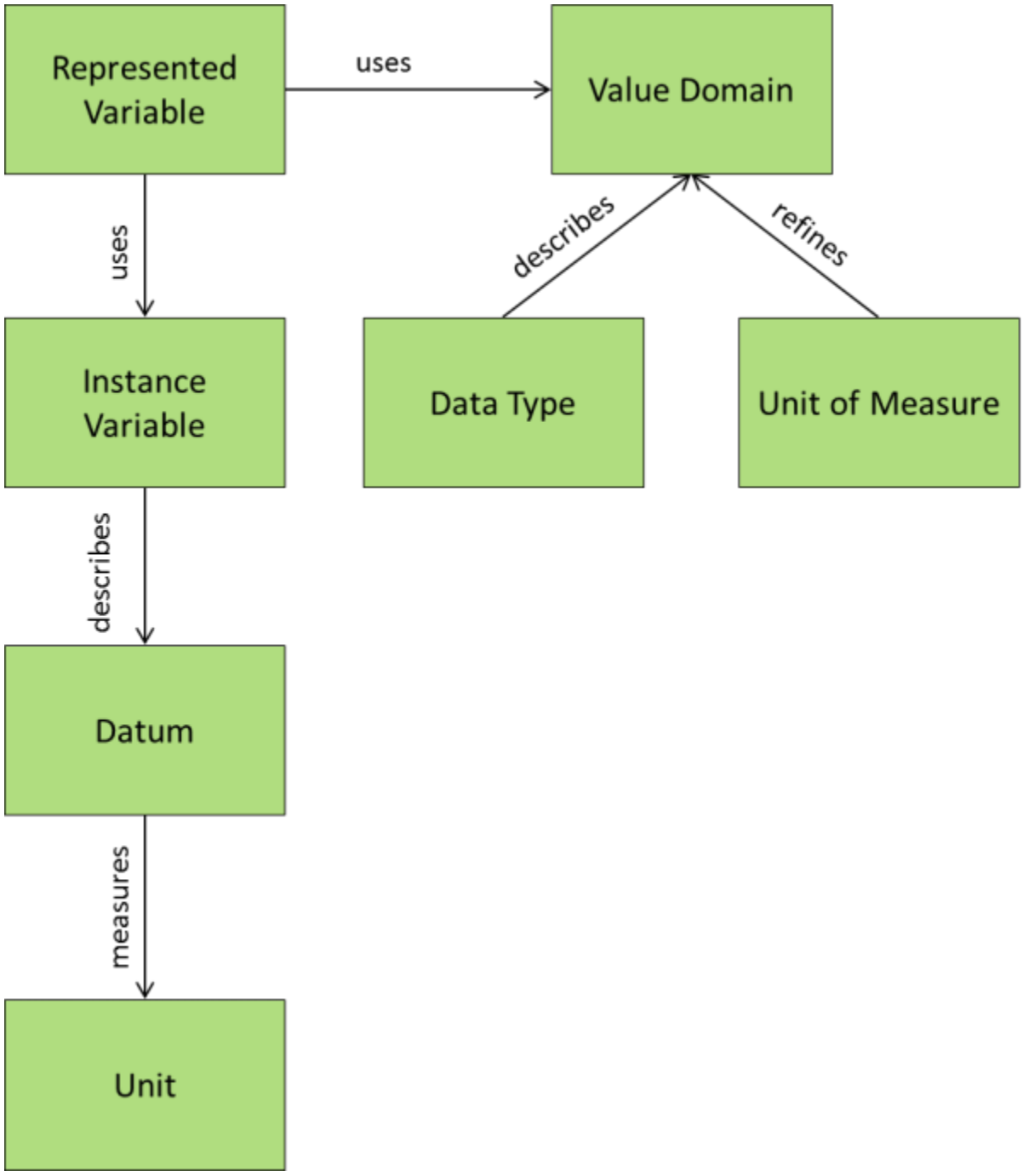

Figure 14. Instance Variable

75. An *Instance Variable* (see Figure 14) is a particular *Represented Variable* associated with a collection of data (*Datum*). This corresponds to a column of data in a database. More particularly, the age of all the US presidents either now (if they are alive) or the age at their deaths is a column of data described by an *Instance Variable*, which is a combination of the *Represented Variable* "Age" and the *Value Domain* of "decimal natural numbers (in years)".

76. A *Datum* is defined by the measure of a *Value Domain* combined with the link to a *Unit* (for example, persons or businesses). A *Datum* is also associated with a *Data Type* and a *Unit of Measure* through the *Value Domain*.

Classifications

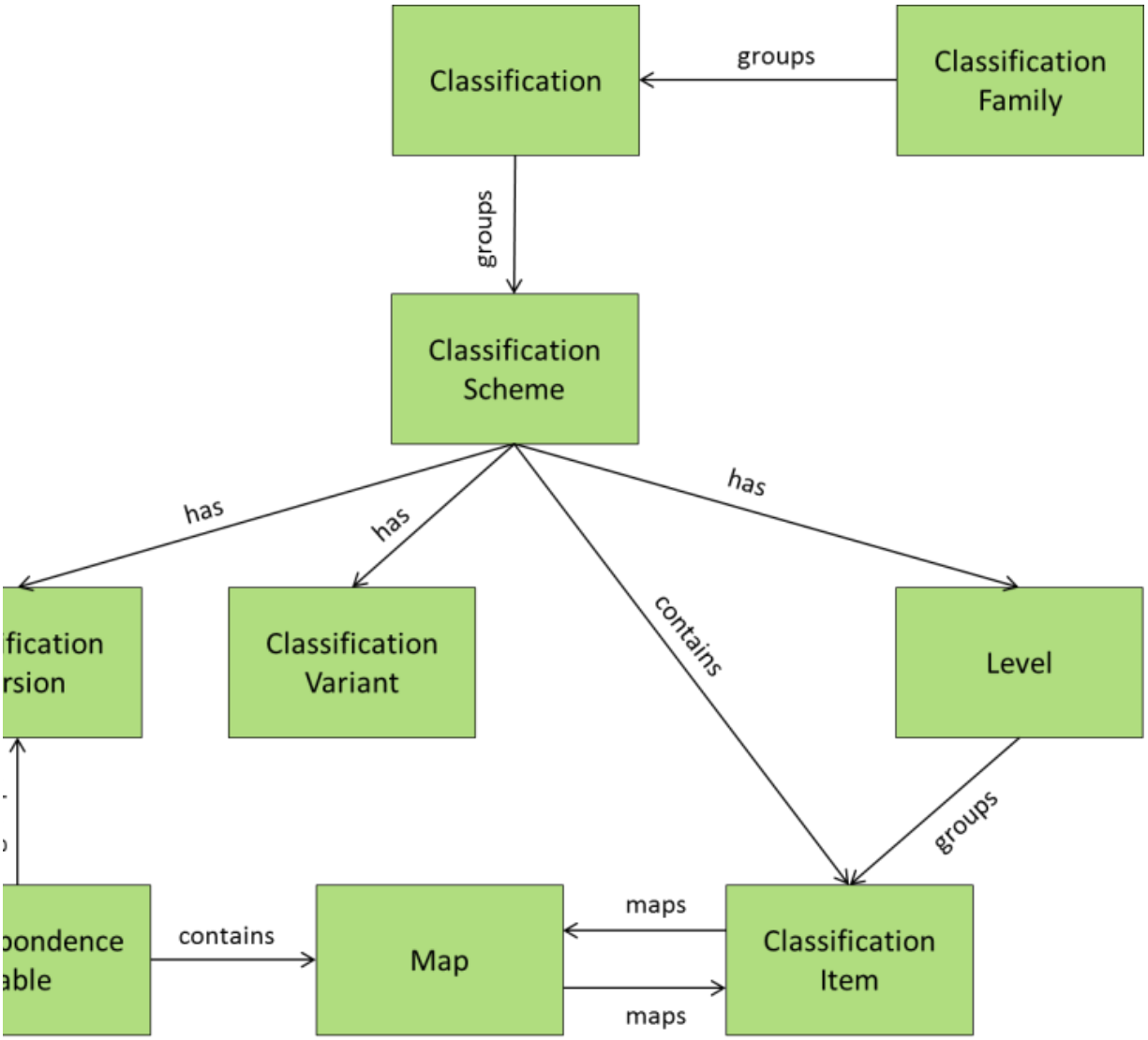

Figure 15. Over view of Classification

77. Figure 15 provides an overview of the objects relating to *Classifications*. *Classifications* describe the *Category* role of a *Concept*.

78. A *Classification* is a categorization of real world objects so that they may be grouped, by like characteristics, for the purposes of measurement, for example ISIC (International Standard Industrial Classification of All Economic Activities). *Classifications* can be grouped into a *Classification Family*, such as industrial activity.

79. A *Classification* such as ISIC is a set of related *Classification Schemes*. It relates *Classification Schemes* that differ as *Classification Versions* or *Classifi cation Variants*. A *Classification Variant* is based on a *Classification Version*. In a *Classification Variant*, the *Categories* of the *Classification Version* are split, aggregated or regrouped to provide additions or alternatives to the standard order and structure of the base *Classification Version*. A *Classification Sc heme* has *Categories* organized into *Levels* determined by the hierarchy. A *Level* is a set of *Concepts* that are mutually exclusive and exhaustive, for example, section, division, group and class in ISIC rev 4.

80. A *Classification Item* combines the meaning, representation and additional information in order to meet the *Classification* criteria, for example "A - agriculture, forestry and fishing" and accompanying explanatory text such as information about what is included and excluded.

81. A *Correspondence Table* can be created by a *Map* that links a *Classification Item* in a *Classification Scheme* with a corresponding *Classification Item* in another *Classification Scheme* via the *Category* corresponding to both *Classification Items*. For example, in a table displaying the relationship between ISIC Rev.4 and the North American Industry Classification System (NAICS 2007 (US)), 0111 in ISIC Rev.4 is related to 111110 in NAICS.

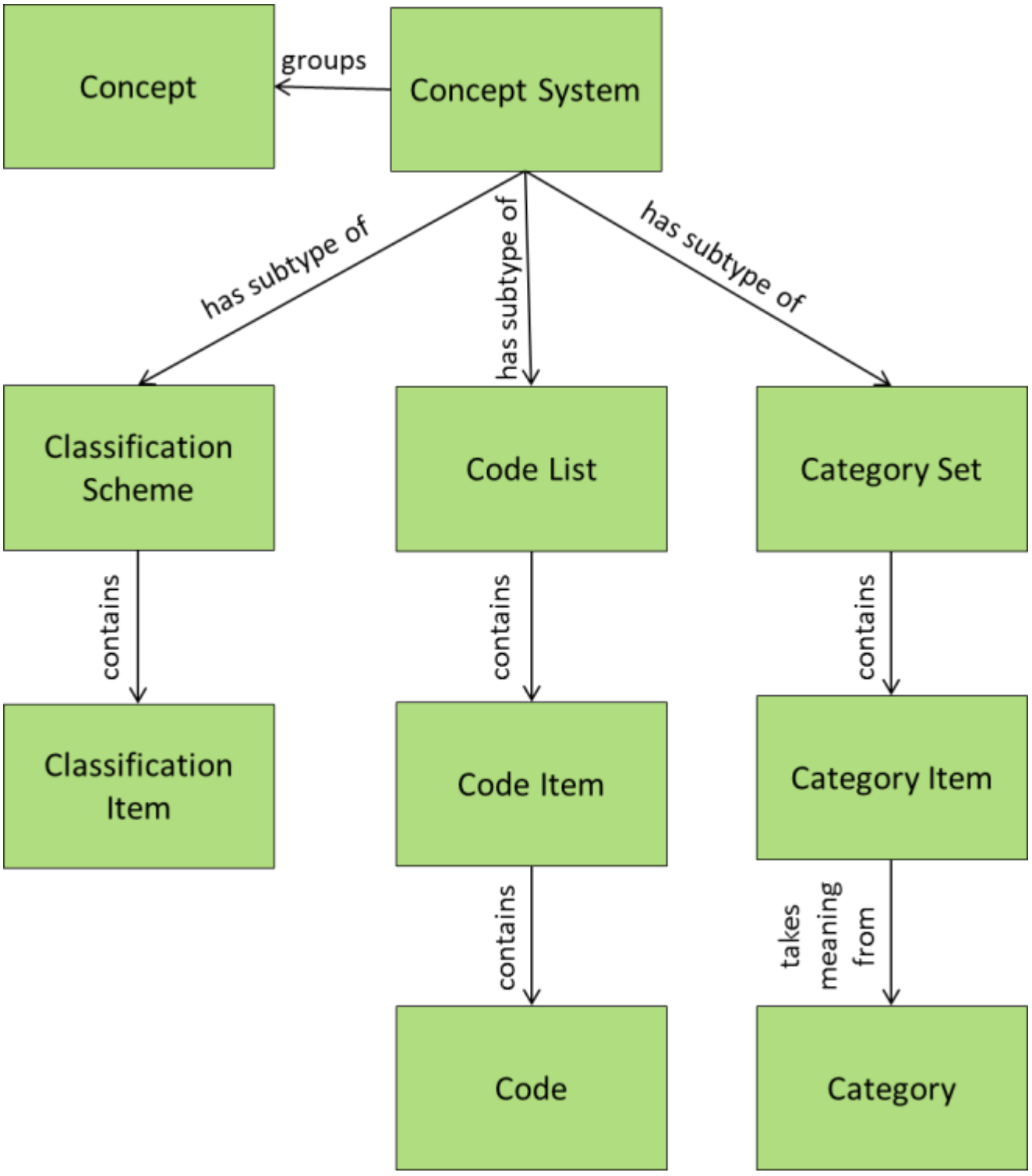

Figure 16. Concept Systems

82. A *Category* is typically part of a *Category Set*, which is a subtype of *Concept System*. A *Category Set* contains one or more *Category Items*. A *Category* can be represented in a *Category Set*, a *Code List* or a *Classification Scheme*. A *Category* provides meaning to these information objects, for example "agriculture, forestry and fishing" or "female".

83. A *Code List* is also a type of *Concept System.* It is used for creating a group of *Codes* and their associated *Categories*. It can consist of one or more *Code Items.* A *Code* designates a *Category* providing representation to the meaning from the *Category. F*or example in "F - female", the *Code* is F and the *Category* is Female.

E. Structures Group

84. The GSIM Structures Group contains sets of information objects that describe and define the terms used in relation to data and their structure. Like the information objects in the Concepts Group, the information objects in this group are used as *Process Inputs* and are often referred to in *Products* and *Repre sentations* to provide information that helps users understand the structure of the data.

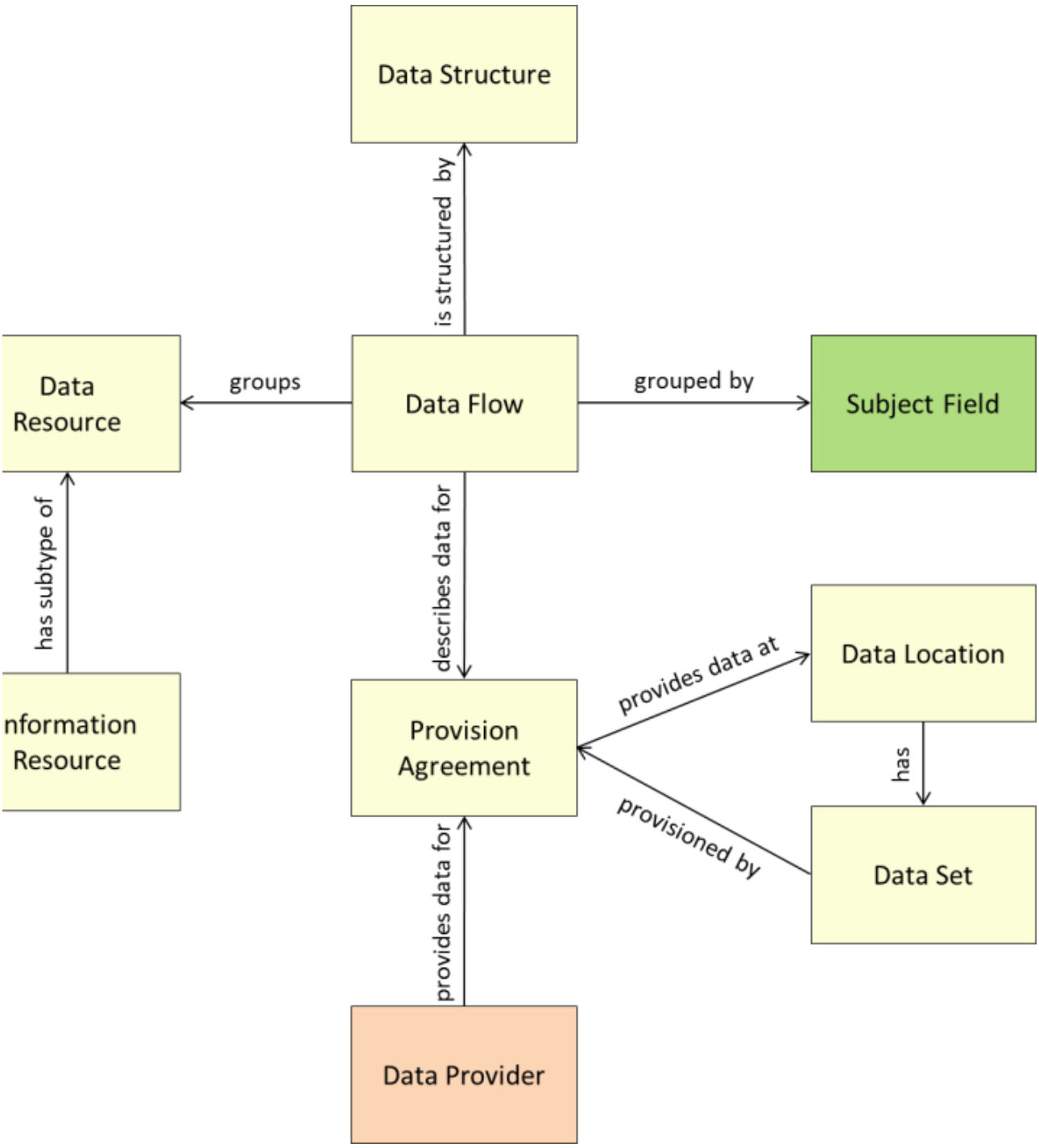

Figure 17. Data Resource

85. An *Acquisition Program* (see Figure 4) conducted by a statistical organization produces or supplies an *Information Resource* (Figure 17). In GSIM, one subtype of an *Information Resource* has been specified. This is the *Data Resource*.

86. A *Data Resource* is comprised of *Data Sets.* These *Data Sets* are made available as part of:

- an *Acquisition Activity* (that is, made available by the data providers for data acquisition or resulting from the *Acquisition Activity*); or
- a *Dissemination Activity*.

87. For a *Data Resource,* the *Data Set* is discovered and provided by means of the *Data Location*. The *Data Location* specifies from where the data can be retrieved. Either this can be a link to a specific file containing the data or to a *Dissemination Service* (see Figure 20) that will consume a query for the data and will return a *Data Set*. If the link is to a *Dissemination Service* then it is probable that the *Dissemination Service* is able to be queried for many types of data and so can provide many *Data Sets*. Each *Data Set* must be structured according to a known *Data Structure* (for example, a known structure for Balance of Payments, Demography, Tourism, Education etc.).

88. The *Data Location* is associated with a specific *Provision Agreement* which identifies the *Data Provider* and the *Data Flow.* Only one *Data Structure* can structure data relating to a *Data Flow*. A *Data Flow* can be grouped by *Subject Field*s (for example, National Accounts, Balance of Payments, Demography) which support data discovery.

89. It is mandatory that the *Data Set* is linked to a *Provision Agreement* to which it relates (that is, the union of the *Data Provider* and the *Data Flow*).

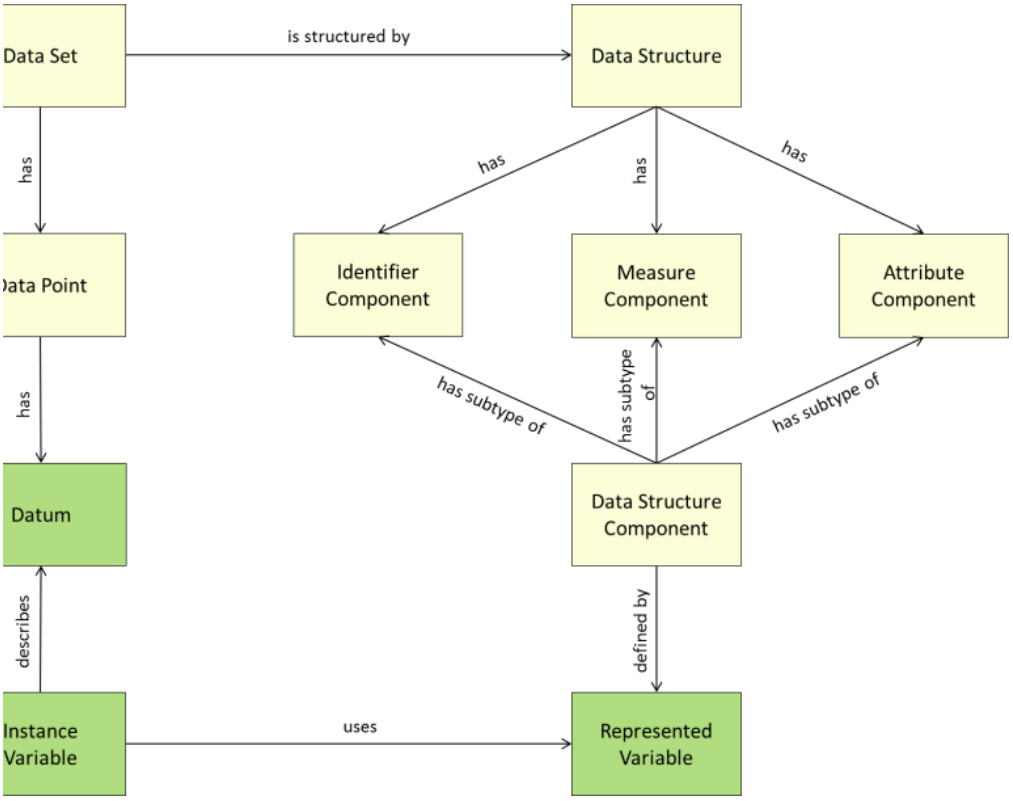

Figure 18. Data Set

90. A *Data Set* has *Data Points*. A *Data Point* is placeholder (for example, an empty cell in a table) in a *Data Set* for a *Datum*. The *Datum* is the value that populates that placeholder (for example, an item of factual information obtained by measurement or created by a production process). A *Data Structure* describes the structure of a *Data Set* by means of *Data Structure Components (Identifier Components, Measure Components* and *Attribute Components)*. These are all *Represented Variables* with specific roles.

91. *Data Sets* come in different forms, for example as Administrative Registers, Time Series, Panel Data, or Survival Data, just to name a few. The type of a *Data Set* determines the set of specific attributes to be defined, the type of *Data Structure* required (*Unit Data Structure* or *Dimensional Data Structure*), and the methods applicable to the data.

92. For instance, an administrative register is characterized by a *Unit Data Structure*, with attributes such as its original purpose or the last update date of each record. It contains a record identifying variable, and can be used to define a *Frame Population*, to replace or complement existing surveys, or as an auxiliary input to imputation. Record matching is an example of a method specifically relevant for registers.

93. An example for a type of *Data Set* defined by a *Dimensional Data Structure* is a time series. It has specific attributes such as frequency and type of temporal aggregation and specific methods, for example, seasonal adjustment, and must contain a temporal variable.

94. Unit data and dimensional data are perspectives on data. Although not typically the case, the same set of data could be described both ways. Sometimes what is considered dimensional data by one organization (for example, a national statistical office) might be considered unit data by another (for example, Eurostat where the unit is the member state). A particular collection of data need not be considered to be intrinsically one or the other. This matter of perspective is conceptual.

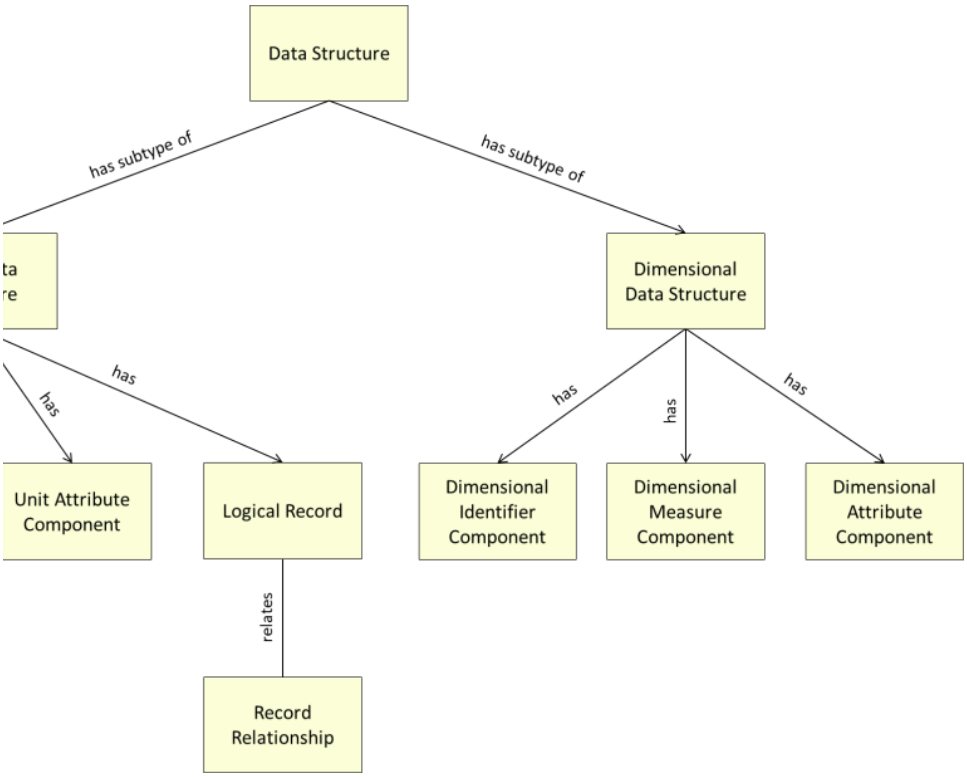

Figure 19. Dimensional and Unit Data Structures

95. A *Dimensional Data Structure* describes the structure of a *Dimensional Data Set* by means of *Dimensional Identifier Components, Dimensional Measure Components* and *Dimensional Attribute Components*. These are all *Represented Variables* with specific roles.

96. The combination of dimensions contained in a *Dimensional Data Structure* creates a key or identifier of the measured values. For instance, country, indicator, measurement unit, frequency, and time dimensions together identify the cells in a cross-country time series with multiple indicators (for example, gross domestic product, gross domestic debt) measured in different units (for example, various currencies, percent changes) and at different frequencies (for example, annual, quarterly). The cells in such a multi-dimensional table contain the observation values.

97.A measure is the variable that provides a container for these observation values. It takes its semantics from a subset of the dimensions of the *Dimensional Data Structure*. In the previous example, indicator and measurement unit can be considered as those semantics-providing dimensions, whereas frequency and time are the temporal dimensions and country the geographic dimension. An example for a measure in addition to the plain 'observation value' could be 'pre-break observation value' in the case of a time series. Dimensions typically refer to *Variables* with coded *Value Domains*, measures to *Variables* with uncoded *Value Domains*.

98. A *Unit Data Structure* describes the structure of a *Unit Data Set* by means of *Unit* Identifier *Components, Unit Measure Components* and *Unit Attribute Components*. These are all *Represented Variables* with specific roles.

99. A *Unit Data Structure* specifies the structure of unit data. It distinguishes between the logical and physical structure of a *Data Set*. A *Unit Data Set* may contain data on more than one type of Unit, each represented by its own record type.

100. *Logical Records* describe the structure of such record types, independent of physical features by referring to *Represented Variables* that may include a unit identification (for example, household number). A *Record Relationship* defines source-target relations between *Logical Records*.

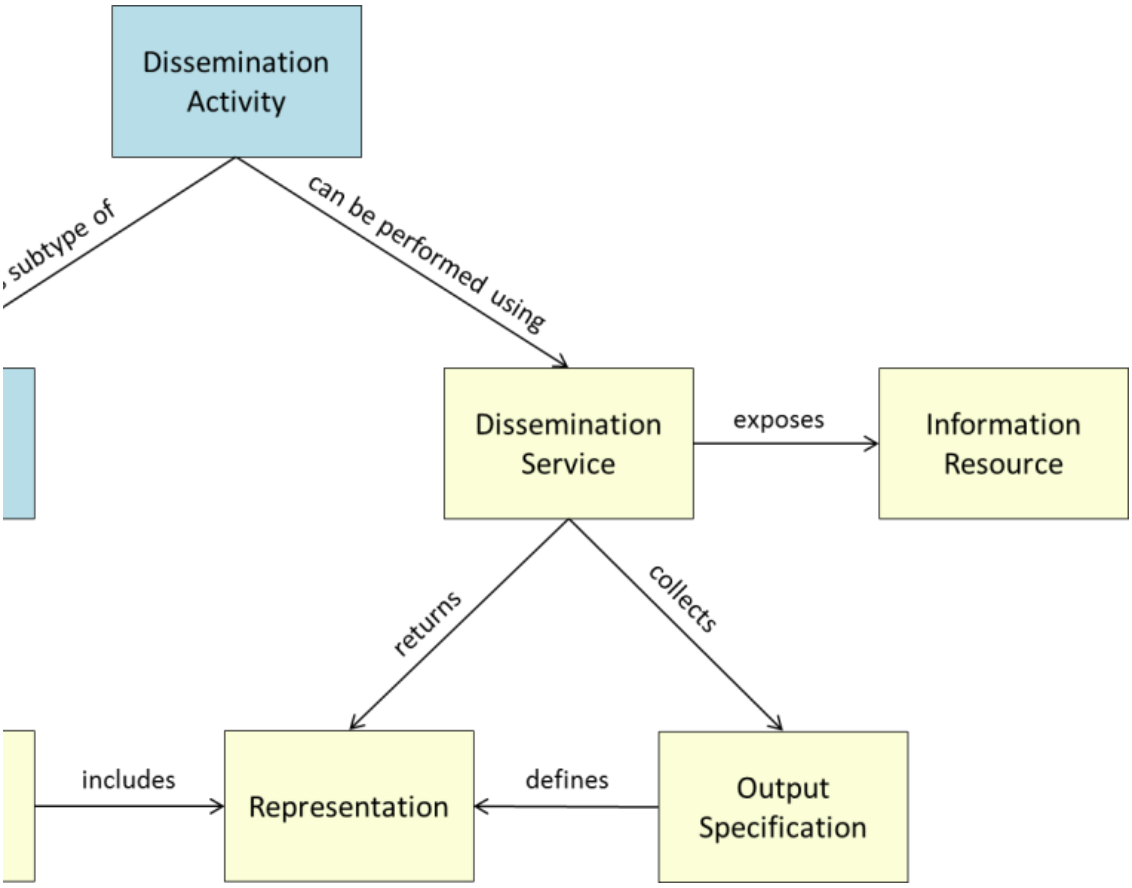

Figure 20. Dissemination Activity

101. A *Dissemination Service* exposes the *Data Sets* and other metadata that is contained in the *Information Resource.* It is the mechanism to create and disseminate *Representations* to consumers. These *Representations* are created dynamically on the specific request and according to the specific needs of the consumer (the *Output Specification*). *Representations* may contain any type of information, for instance statistical data (as a *Data Set* or visualization) or structural or conceptual metadata like a *Data Structure*, a *Code Set* or a description of a *Concept*.

102. A *Product* is the result of a *Publication Activity*. *Products* are stored for later dissemination through *Dissemination Services*. Examples of *Products* are publications, press releases, etc. *Representations* may be used as input to, and as components of, a *Product*.

F. Base Group

103.The GSIM Base Group consists of several information objects that can be seen as the fundamental building blocks that support many of the other information objects and relationships in the model. These information objects form the nucleus for the application of GSIM information objects. They provide features which are reusable by other information objects to support horizontal functionality such as identity, versioning etc. For these reasons, many of these information objects are rather abstract in nature.

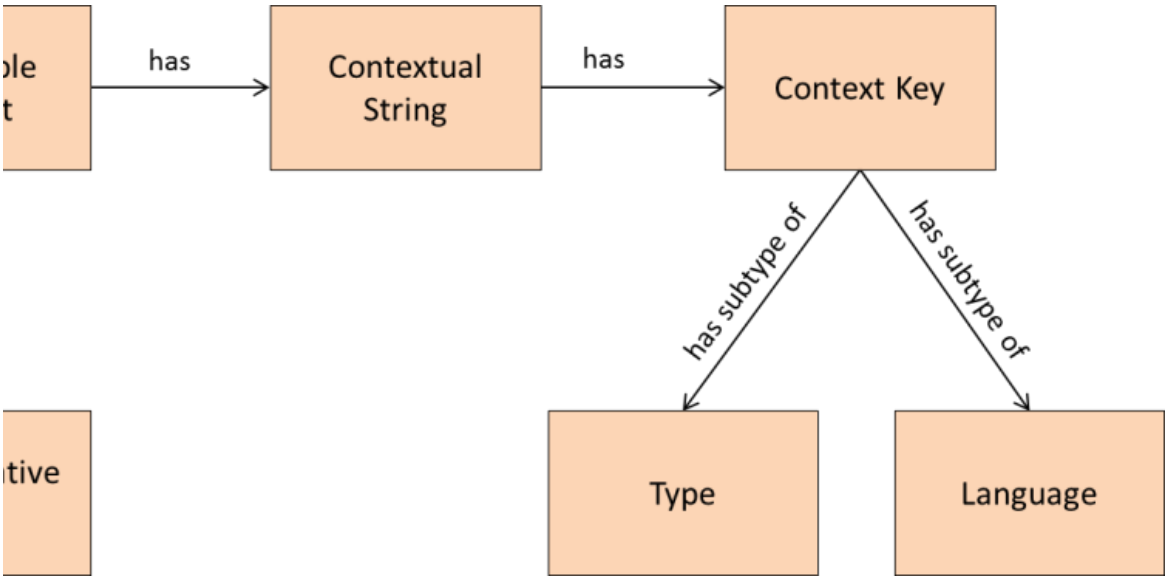

Figure 21. Base Artefacts

104. The only base artefact in GSIM that gives underlying identity and naming is the *Identifiable Artefact*. It can be inherited by any class in GSIM for which identity, name, description, and additional documentation is required.

105. The *Identifiable Artefact* has three associations to *Contextual String* – one for each of name, description, and documentation. The value in the *Contextual String* is given a context by the *Context Key* which can be *Type* or *Language*.

106. There is no attempt in GSIM to model the administration of items in repositories such as the maintenance agency, versioning, repository functions. However, the *Identifiable Artefact* does have a link to *Administrative Details* where such details can be added using the GSIM extension methodology.

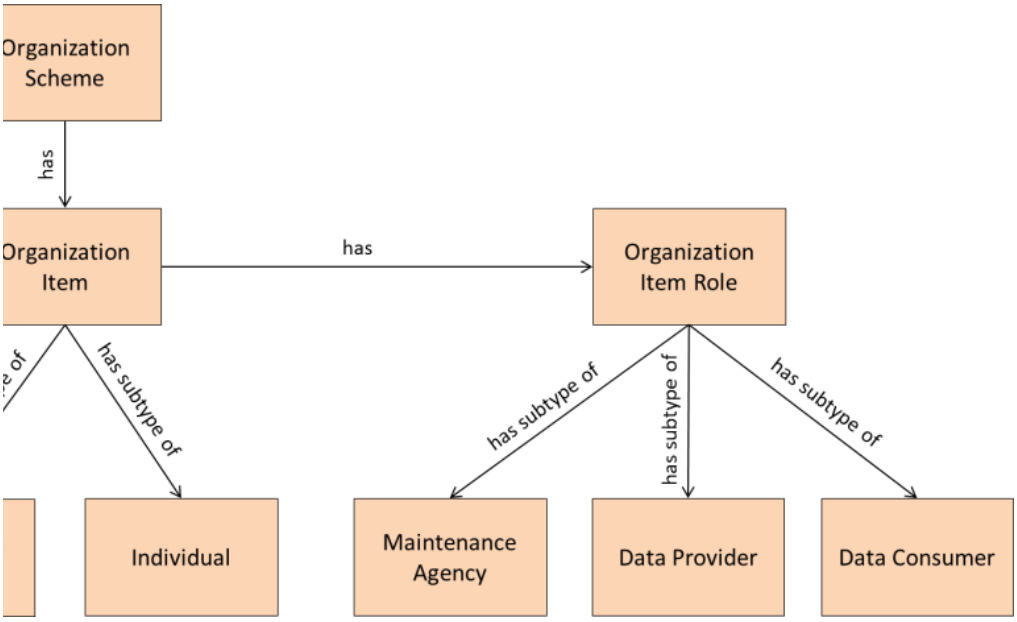

Figure 22. Organization

107. An *Organization Scheme* comprises *Organization Items*, each of which can be an *Organization Unit* or an *Individual*. The *Organization Unit* can be in a hierarchic scheme of *Organization Items*. An *Individual* or *Organization Unit* can have a number of different *Contact Details.*

108. The *Individual* or *Organization Unit* can play zero or more recognized roles (*Organization Item Role*) in the maintenance (*Maintenance Agency*) data collection (*Data Provider*) and dissemination (*Data Consumer*) processes.

Word version of this section only