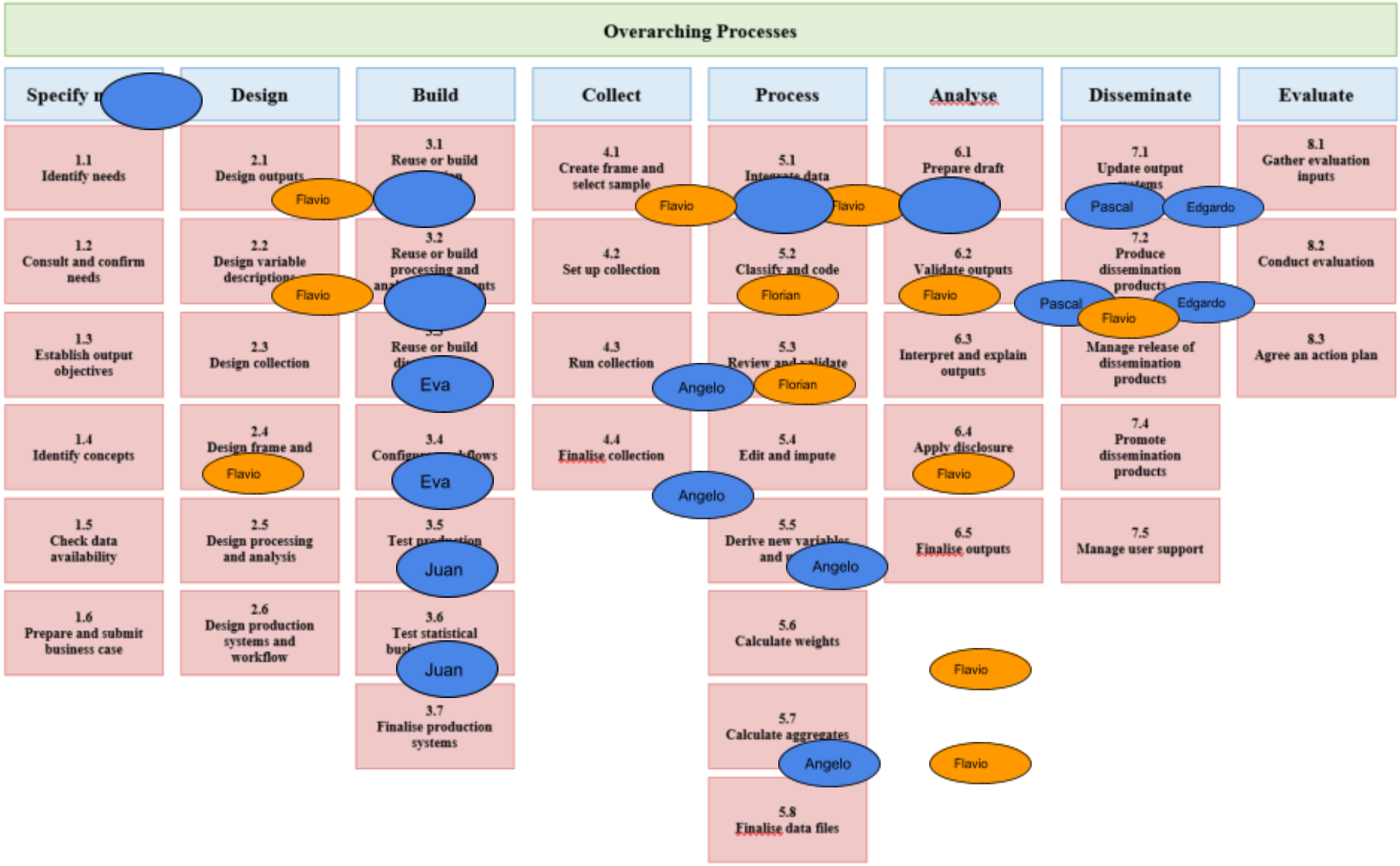
List

	SDMX	DDI
eeds	1.1 Identify needs 1.2 Consult and confirm needs: 1.3 Establish output objectives 1.4. Identify Concepts 1.5 Check data availability 1.6 Prepare and submit business case	1.4. Identify Concepts Explore DevelopmentActivity in DDI-L (Christophe), maybe an interesting object (nevertheless not sure of his scope.. if it is only for Design phase or if it could be used also in Need phase)
	2.1 Design outputs 2.2 Design variables description 2.3 Design collection (Eva; SDMX) 2.4 Design frame and sample (Eva; SDMX) 2.5 Design processing and analysis (Juan; SDMX) 2.6 Design production system and workflow (Juan; SDMX)	2.1 Design outputs (Flavio; DDI; Nov 15th) 2.2 Design variables description 2.3 Design collection (Flavio; DDI) 2.4 maybe cognitiveExpertReviewActivity (Christophe) 2.6 Design production system and workflow (Flavio)
	3.1. Reuse or build collection instruments 3.2. Reuse or build processing and analysis components 3.3. Reuse or build dissemination components	3.1 Build and reuse collection instrument Feasible in DDI L, to be described (Christophe)
	4.1 Create frame and sample	4.1 Create frame and sample

	4.3 Run collection (Angelo; SDMX) 4.4 Finalise collection (Angelo; SDMX)	
	5.1 Integrate data 5.2 Classify and Code (Eva; SDMX) 5.5 Derive new variables and units (Angelo; SDMX) **5.7 Finalise data files (Angelo; SDMX)**	**5.1 Integrate data** 5.2 Classify and Code (Flavio) 5.4 (Flavio) 5.6 Calculate weights (Flavio; DDI) **5.7 Finalise data files (Flavio; DDI)**
	6.2 Validate outputs (Pascal; SDMX) **6.3 Interpret and explain outputs (Gabriele; SDMX)** **?? 6.4 Apply disclosure control (Edgardo; SDMX)**	**6.2 Validate outputs (Flavio; DDI; Nov 15th)** **6.3 Interpret and explain outputs (Christophe, Flavio; DDI)** **6.4 Apply disclosure control (Christophe, Flavio; DDI)**
te	7.1 Update output systems(Edgardo; SDMX) 7.2 Produce dissemination products (Edgardo; SDMX) 7.3 Manage release of dissemination products (Edgardo; SDMX) 7.4 Promote dissemination products (Edgardo; SDMX)	
	8.2. Conduct evaluation **8.3 Agree an action plan**	

Homework

- Juan: 2.5, 2.6 + phase 3?
- Pascal: 6.2
- Gabriele: 6.3
- Flavio: 5.2 and 5.4
- Florian & Christophe: (DDI) sub-processes involved with collection instrument., 2.3 [Done], TBD: 4.2, 3.1, 4.3
- [Done] Edgardo: Dissemination phase
- [Done] Eva: 2.3 and 2.4, 5.2
- [Done] Angelo: 4.3, 4.4, 5.5, 5.7


Overarching Processes
Specify n
Design
Build
Collect
Process
Analyse
Disseminate
Evaluate
1.1 Identify needs
1.2 Consult and confirm needs
1.3 Establish output objectives
1.4 Identify concepts
1.5 Check data availability
1.6 Prepare and submit business case
2.1 Design outputs
2.2 Design variable descriptions
2.3 Design collection
2.4 Design frame and
2.5 Design processing and analysis
2.6 Design production systems and workflow
3.1 Reuse or build
3.2 Reuse or build processing and
Reuse or build
3.4 Configu flows
3.5 Test production
3.6 Test statistical busi
3.7 Finalise production systems
4.1 Create frame and select sample
4.2 Set up collection
4.3 Run collection
4.4 Finalise collection
5.1 Integrate data
5.2 Classify and code
5.3 Review and validate
5.4 Edit and impute
5.5 Derive new variables and
5.6 Calculate weights
5.7 Calculate aggregates
5.8 Finalise data files
6.1 Prepare draft
6.2 Validate outputs
6.3 Interpret and explain outputs
6.4 Apply disclosure
6.5 Finalise outputs
7.1 Update output
7.2 Produce dissemination products
Manage release of dissemination products
7.4 Promote dissemination products
7.5 Manage user support
8.1 Gather evaluation inputs
8.2 Conduct evaluation
8.3 Agree an action plan
Flavio
Pascal
Edgardo
Florian
Eva
Angelo
Juan

Objective

1. Short description why/how SDMX/DDI helps as an entry point
2. List relevant SDMX/DDI artifacts under each sub-process
3. Map relevant SDMX/DDI artifacts under each sub-process to GSIM class

Introduction

- add an introduction explaining the intent of this activity, i.e. (i) clarify/emphasize that SDMX and DDI are complementary rather than competing standards; (ii) describe how each standard can be used to implement each sub-process, considering that GSBPM operates as a template or jig on which the two standards interoperate; and (iii) elucidate and provide guidance on which constructs to use to stablish a “handshake” between standards, when necessary.
- we need separating the discussion in terms of constructs from the respective information models, DDI and SDMX, and other components/supporting elements each standard provides, e.g. registries, guidelines, harmonized content at different levels (organizational, national, international), etc.

Common statements

SDMX

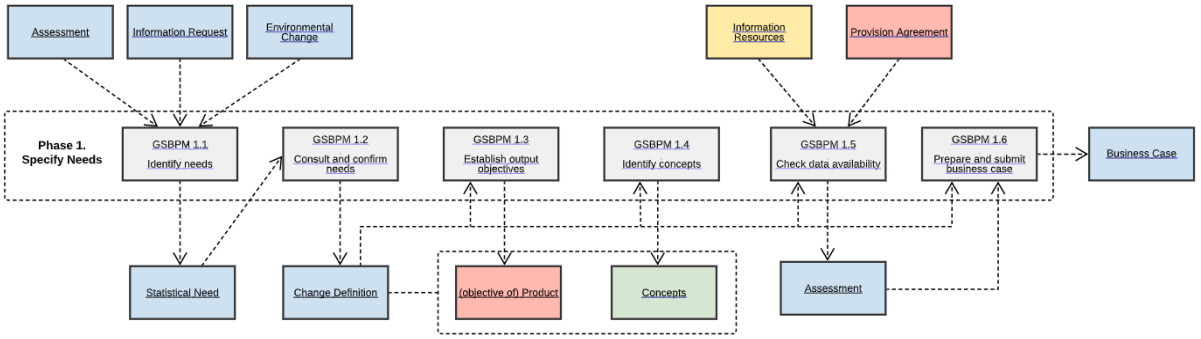
(under discussion)
GSBPM ovearching process: Statistical data handling
According to the Statistical Subject-Matter Domains of SDMX could be handled the access rights, maintainance. We could write down our statistical data production process by usage of Information model. It identifies objects and their relationships, data storage, data protection, planning and working of data accessability and it allows central management and standard access. In other words, statistical data, metadata and the data exchange process are all modelled.

GSBPM ovearching process: Metadata-management

The SDMX Glossary, SDMX Global Registry are special instance Registry of metainformation, that are used by the SDMX, Implementation of these could promote and document the whole statistical data production process and serves the purpose of storing and making openly available some global-level artefacts (which is implemented using Fusion

Registry, indeed). The metadata reusing serves the comperability, interoperability and understandability.

Phase “Specify needs”



Design phase (and specify need phase?)

In DDI there are a set of object to describe expert activities during developpement phase, so in particulare 2.3 design collection

the ddi object **Developmentactivity,** that is an Extensible structure for development activity elements used in describing the development of a questionnaire. For example: ContentReviewActivity, TranslationActivity, and PretestActivity.

We can describe who is in charge, what is the output , what is the process of this activity.

Substitutions for **Developmentactivity** exists that **add a type. Compared with Developmentactivity There is no additional information other than** the specification of the type of cognitive expert review taking place for development purposes.

The different type of CognitiveExpertReviewActivityType, CognitiveInterviewActivityType, ContentReviewActivityType, FocusGroupActivityType, PretestActivityType, TranslationActivityType

My question about that is it seems to me that this kind of activity are around the development of a questionnaire, in what extent can we use also this object in other tasks ?

Sub-process 1.1 Identify needs:

SDMX (Eva)

The needs of official statistics could come from inside from the nation and NSI, and from outside like Eurostat, OECD, international organizations of SDMX Community. The trigger can be a new legal act, users (decision maker organization, researcher...) or statistical domain (as new data , or maintainer data) needs.

Sub-process 1.2 Consult and confirm needs:

SDMX (Eva)

The NSI get a request for the compulsory data transmission as it is mentioned in 1.1. The details of the needs are included into the Provision Agreement.
(Organizations of the SDMX give the opportunity to take part in a pilot before starting a compulsory, repeating and regular data and metadata exchange.)
If the need comes from outside the SDMX community (for example: government or NSI) the need can be specified like into the Provision Agreement.

Sub-process 1.3 Establish output objectives:

SDMX (Eva)

One side, the objectives of the compulsory SDMX data exchange are defined into MSD and DSD as it arrives parallelly with the request. (many statistician should to know it, because they have used that during the pilot).
Other side, output of a voluntary data publishing in SDMX structure can be defined filling the headline or relevant parts of an MSD or/and DSD.

Sub-process 1.4 Identify Concepts

SDMX

- SDMX provides the means of capturing **Concepts** that can then be organized (and managed) in **Concept Schemes**.
- Relevant SDMX artifacts/instrument: Concepts, ConceptScheme; (optional) SDMX modeling guideline, SDMX Glossary

DDI

- DDI provides the means of capturing **Concepts** that can then be organized (and managed) in **ConceptSchemes**.
- Relevant DDI artifacts: Concept, ConceptScheme, DDI-C (optional)

GSIM Mapping
Concept (SDMX) - Concept (DDI) - Concept (GSIM)
? ConceptSchemes (SDMX) - ConceptSchemes (DDI) - Concept System (GSIM)

Sub-process 1.5 Check data availability

SDMX (Eva)

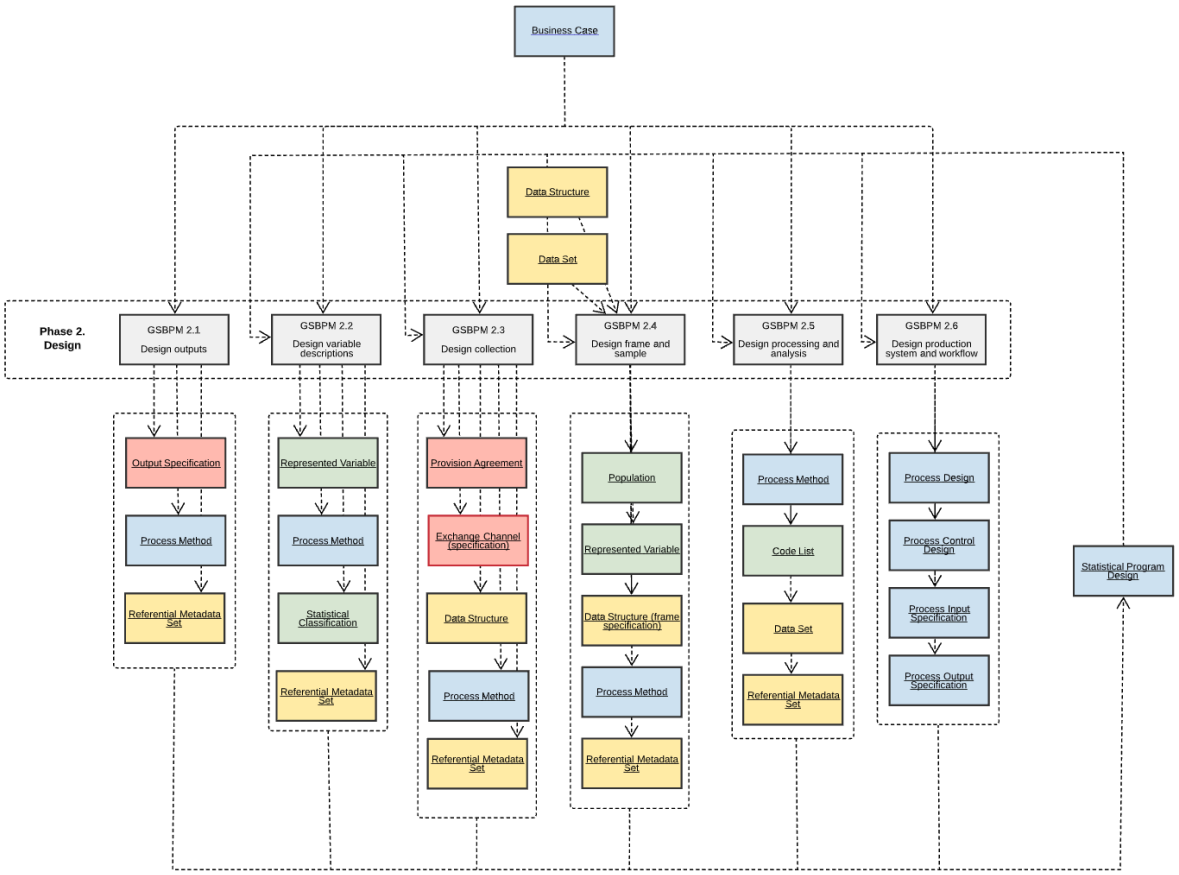
From the previous sub-processes could be known the expectation and details of the needs. Knowing the needed measures, classifications, frequency etc from the Provision Agreement, DSD or MSD, it can be compared with data, that is owned by our institute, by another National Official Statistical organization. If the needed data has not owned yet, there is the tasks to get it by data collection.

Sub-process 1.6 Prepare and submit business case

SDMX (Eva)

After the summarizing the information of I.1-I.5, our NSI can decide:
- Does the NSI have the needed data, or not?
- Is there need for a new data collection, or not.

Phase “Design”



Sub-process 2.1 Design outputs

SDMX

- The “outputs” are supposed to be datasets or statistical tables, which in SDMX are described by a **Dataflow/DSD**.
- When “outputs” are metadata sets or reference metadata, which in SDMX are described by an **MSD** (vastly open, global metadata artefact - e.g., version)
- [to re-write this part] In addition, the **Provision Agreements** and the **DSDs** are instruments that establish requirements that must be taken into account when designing the outputs. We need to be sure that all the information that we have agreed to report/publish it is going to be created.
- Relevant SDMX artefacts: DSD/dataflow, MSD, ??Provision Agreement??
- + DSD guidelines

DDI

I think the core is the variables and datasets used in the products to be disseminated. DDI can provide the usual constructs **Data Sets** and their **Data Structures** together with the **Represented Variables** these **Data Structures** capture.

All these can be encapsulated in **Study Unit**, together with additional dissemination information, like **Quality Statements**, **Coverage**, **Purpose**, **Embargo**, **Universe** and **Methodology.**

As far as I know there is no specific construct to capture policies, handbooks and information about prior cycles or studies, but they can be included with the generic **Related Other Material** and **Other Material Scheme**.

GSIM Mapping
StudyUnit (DDI) - StatisticalProgrammeDesign (GSIM)?
Data Sets (DDI) - Data Set (GSIM)
Data Structure (DDI) - Data Structure
RepresentedVariable (DDI) - Represented Variable (GSIM)

Sub-process 2.2 Design variable descriptions

SDMX

- Define a description and representation (data type and/or enumeration by a **Codelist**) for each **Concept** in the **ConceptScheme**. Should additional descriptive metadata be included, specific metadata attributes linked to the concept may be added.
- Write sth about standardization (or harmonization - depending on how u see :))?
- Relevant SDMX artifacts: ??Codelist, Concept, Concept Scheme?? Glossary

DDI

- Both DDI-LC and DDI-CDI provide the means to design variables at the conceptual and representation levels. A **ConceptualVariable** provides the link between a **Concept**, e.g. marital status, income, and a **UnitType**, e.g. person, business. A **RepresentedVariable** adds to a **ConceptualVariable** representation, in the form of a **ValueDomain** (called **RepresentationType**, in DDI-LC). Categorical **ValueDomains** can be either **StatisticalClassifications** or **CodeLists**. **Variables** can be organized into **VariableCollections** (called VariableSchemes and VariableGroups in DDI-LC).
- Relevant DDI artifacts: (highlighted ones above)

GSIM Mapping
UnitType (DDI) - Unit Type (GSIM)
ConceptualVariable (DDI) - Variable (GSIM)?

RepresentedVariable (DDI) - Represented Variable (GSIM)
ValueDomains (DDI) - Value Domain (GSIM)
CodeLists (DDI) - Code List (GSIM)
StatisticalClassifictions - Statistical Classifiction (GSIM)

Sub-process 2.3 Design collection

SDMX (Eva)

- If there is data transmission, the data exchanging is promoted by the DSD,MSD. It consists of data structure, classifications, process method, which is handled by the data owner.
  We can use the DSD and MSD of collections from the sub-process 2.2 (classification, codes, structure).
- Relevant SDMX artefacts: DSD, MSD, ConceptSheme.

DDI (Flavio)

DDI provides elements to describe a data collection through one to many **DataCollection** attached to a **StudyUnit**.

- the design of running the collection can be detailed with **CollectionEvent** and **ProcessingEvent**
- the design of the collection **Instrument** can be detailed with **QuestionItem**, **Instruction** and **ControlConstruct** (flow logic within the instrument)
- the provision agreements information are partially mapped with several **StudyUnit (or group)** objects, namely : **InformationClassification , Embargo, AuthorizationSources**

A **QuestionItem** can be linked to a **RepresentedVariable** as designed in the sub-process 2.2, a **CollectionEvent** to a **Sample** as designed in the sub-process 2.4

To help share and reuse, collection instruments or their components can be stored and organised in resource packages : **InstrumentScheme**, **QuestionScheme, InterviewerInstructionScheme** and **ControlConstructScheme**.

Some controlled vocabularies are also provided for the mode of collection, the type of instrument, the response unit…

The process of designing a collection can also be described with a **DataCaptureDevelopment**.

Sub-process 2.4 Design frame and sample

SDMX (Eva)

- That data collection, which based on register(s) or censuses, could helpful the frame and sample designing, if there is available the DSD(s) of the register(s) and censuses. DSD (classifications, codelists), MSD, the ConceptSheme can used from sub-process 2.2.
- VTL can be used to plan frame and sample
- Relevant SDMX artefacts: DSD, MSD, ConceptSheme

Sub-process 2.5 Design processing and analysis

SDMX (Juan)

- When designing the processing and analysis, it is possible to use **Concept Schemes** to support the reuse of elements that have been used in other previous cycles of statistical projects, which also helps to achieve conceptual harmonisation within a domain.
- Unique identifiers can be assigned to each concept contained in a **Concept Scheme** along with associated **Code Lists** and used to strengthen the design of coding routines and rules.
- **Data Structure Definitions (DSD)** can serve as tools to facilitate the conceptual, syntactic and structural integration of data, especially when these may come from different sources.
- Likewise, the design of adequate **Metadata Structure Definitions (MSD)** may serve as a guide to facilitate the subsequent analysis of the data that will be produced in the process.
- To facilitate the integration of information from different sources, the design of **Data Structure Definitions (DSD)** can be used.
- Seen from the point of view of an organisation that must report information in SDMX, the design must ensure that all the needed information to be reported in attention to a **provision agreement** will be produced in the process

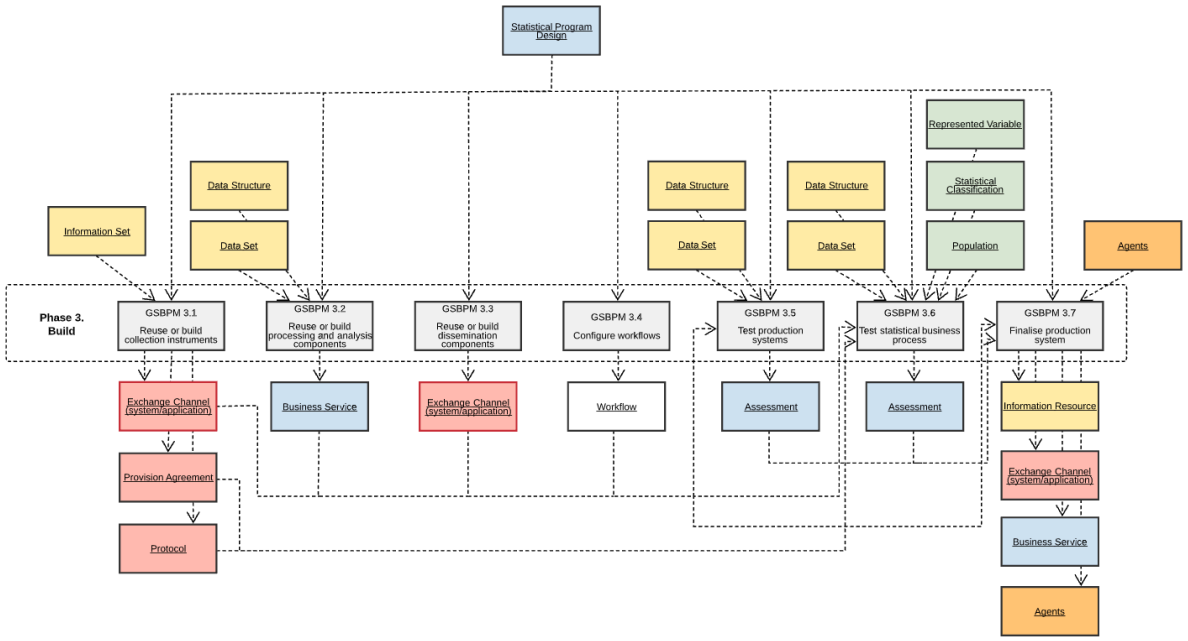
Sub-process 2.6 Design production systems and workflow

SDMX (Juan)

- The SDMX **Concept Schemas**, **DSD** and **MSD** can be taken as tools to help to avoid redundancies and inefficiencies during the production of information.

- If the production systems consider in an early design stage the information that will be exchanged or reported there will be no need to make mappings or transformations to different structures to fulfil provision agreements.
- An additional advantage when considering these elements will be an increase in the interoperability of the databases containing the statistical information from different projects, as they can share concepts, classifications and even some structures.
- Additionally, as SDMX provides tools to manage the data in a generalised way, using its **Information Model** as a reference to create the production systems will be useful to build reusable solutions.
- Just to remember, SDMX does not have a fixed set of **Concepts**, it provides rules and technical specifications that can be used to create the standardised information structures that would be needed to integrate, analyse, disseminate and exchange the statistical information products. The **SDMX API Technical Specifications** can be useful to build the connections between the different modules, systems and software tools needed to support the different phases of the GSBPM.

Phase “Build”


Statistical Program Design
Represented Variable
Data Structure
Data Structure
Data Structure
Statistical Classification
Information Set
Data Set
Data Set
Data Set
Population
Agents
Phase 3. Build
GSBPM 3.1 Reuse or build collection instruments
GSBPM 3.2 Reuse or build processing and analysis components
GSBPM 3.3 Reuse or build dissemination components
GSBPM 3.4 Configure workflows
GSBPM 3.5 Test production systems
GSBPM 3.6 Test statistical business process
GSBPM 3.7 Finalise production system
Exchange Channel (system/application)
Business Service
Exchange Channel (system/application)
Workflow
Assessment
Assessment
Information Resource
Provision Agreement
Exchange Channel (system/application)
Protocol
Business Service
Agents

Sub-process 3.1 Reuse or build collection instruments

SDMX (Juan)

- **Concept Schemes** have associations of concepts and code lists that can be reused when designing the collection instruments as they will constitute the basis to ensure that the information will be collected in a way it can fulfil the **provision agreements**.
- A more precise granularity level for the codes can be used but it is recommended to maintain them aligned to those that will be reported or exchanged later on.
- The use of a single set of Concept Schemes across the organisation fosters data harmonisation and facilitates data interoperability.
- It is important to maintain a record of the decisions made during this process as it will be needed when developing the quality reports which are often part of the reference metadata reports.
- Questionnaires should be built making use of the concepts and its associated representation (e.g. codelists) in the particular collection tools' format, as well as in formatting the output records, including data collection related metadata (process, quality, paradata) stored in attributes.
- If we think in collecting information beyond traditional questionnaires, like the information integrated by international organisations, the information collected in a national information system, or the information sourced from administrative registries; then we can think that a SDMX **DSD** will be an instrument that can be used to collect the information. The reuse of published DSD (like the Global DSD published in the SDMX Global Registry) may be considered as a way to reuse collection instruments providing an additional advantage for the informants as they can report information to more than one organisation using the same dataset. This strategy has the benefit that it reduces the reporting burdening faced by an statistical institution that must provide information from the same statistical program to different organisations.
- An electronic information channel instrumented with tools following the SDMX technical specifications may be reused reducing the need to develop different tools to collect and to integrate the statistical data and metadata.

Sub-process 3.2 Reuse or build processing and analysis components

SDMX (Juan)

- A well designed **DSD** will provide enough information to correctly interpret and analyse the statistical variables. Although SDMX doesn´t provides specifications about how to build processing components, it is recommended that the information required to fill a **DataFlow** or a **MetadataFlow** is considered when building these kind of software tools.

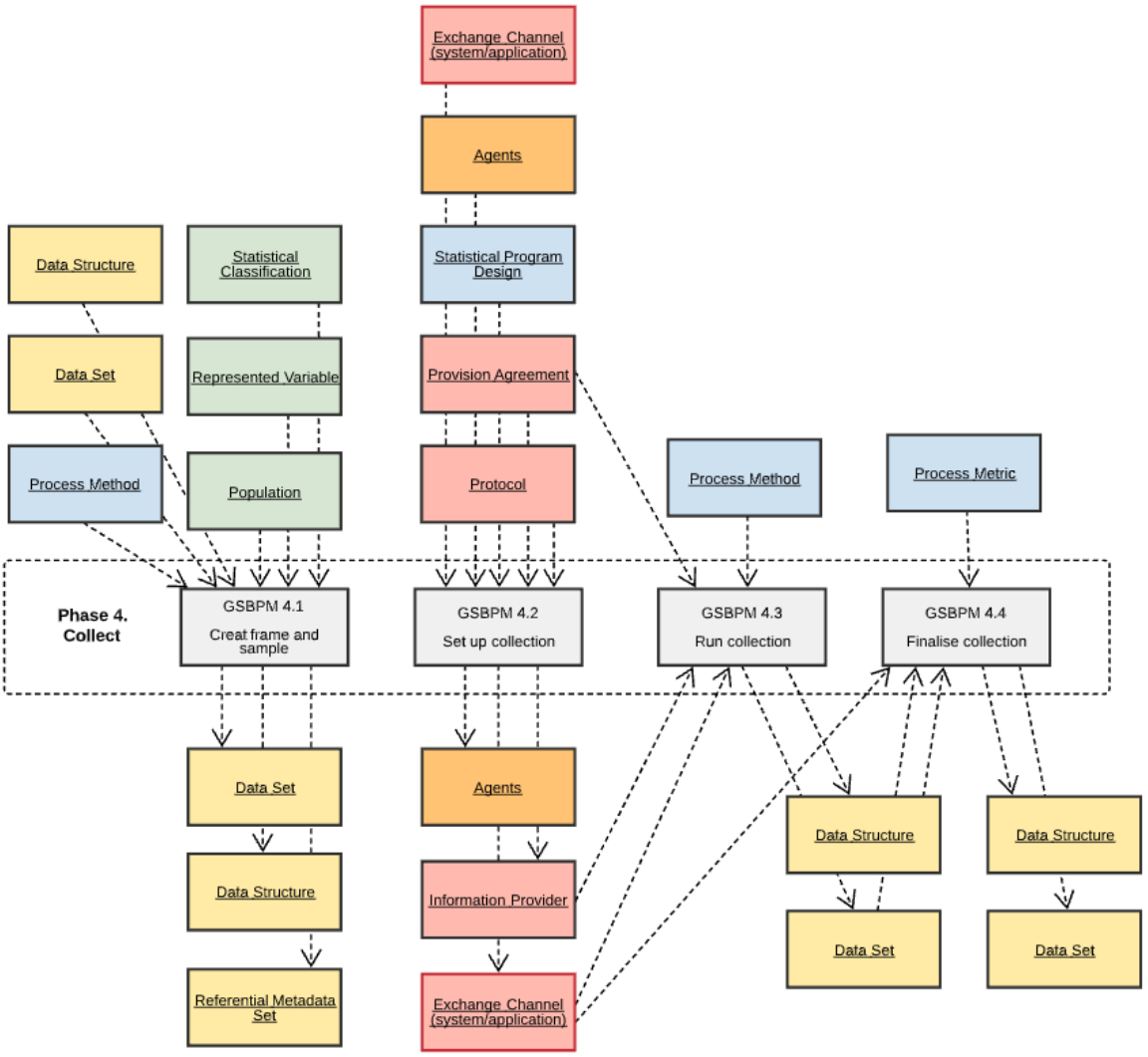
- *SDMX API* provides the technical specifications to create web services for querying the datasets and create tools to analyse the information using different filters.
- If we think in a reference metadata report as an instrument that can provide information to better interpret and analyse the quality of the produced information, then it makes sense to develop the tools that will be helping to automatically document the process and filling the **MSD** defined in a previous designing phase.
- Likewise, analysis results or quality information can be stored in attributes previously defined at different attachment levels, either in the DSD (travelling together with the data) or in a MSD, to be part of a metadata report.
- Besides, “process” metadata can also be stored in SDMX artefacts like metadata sets (as defined by a MSD) and structure or representation maps. This information can afterwards be processed by analysis programs developed in, for example but not limited to, VTL.

Sub-process 3.3 Reuse or build dissemination components

SDMX (Juan)

- SDMX provides technical specifications that can be used as the basis to build web services to deliver statistical information. Indeed, there are tools implemented using the SDMX specifications that can be used to provide this kind of services.
- By following the technical specifications generic applications able to deal with statistical information from different domains can be implemented. This is an advantage as the tools can be reused.
- The implementation of DataFlows and MetadataFlows based on Internet Web Services allows the publication and updating of information in different organisations at the same time. The subscription notification mechanisms described as part of the SDMX Registries can be used to make this kind of implementation. Tools that have been built following the SDMX standards can be reused by different organisations to deploy this kind of information services.

Phase “Collect”



Sub-process 4.1 Create frame and sample

SDMX

- SDMX **MSD** can help with management of metadata reports
- SDMX **DSD** provides standard structure and logical descriptions of the datasets that are involved (register, frame, sample) which is helpful for organisations that do not have standard naming conventions or structures for data sets
- Relevant SDMX artefacts: MSD, DSD, Content Oriented Guideleines?

DDI

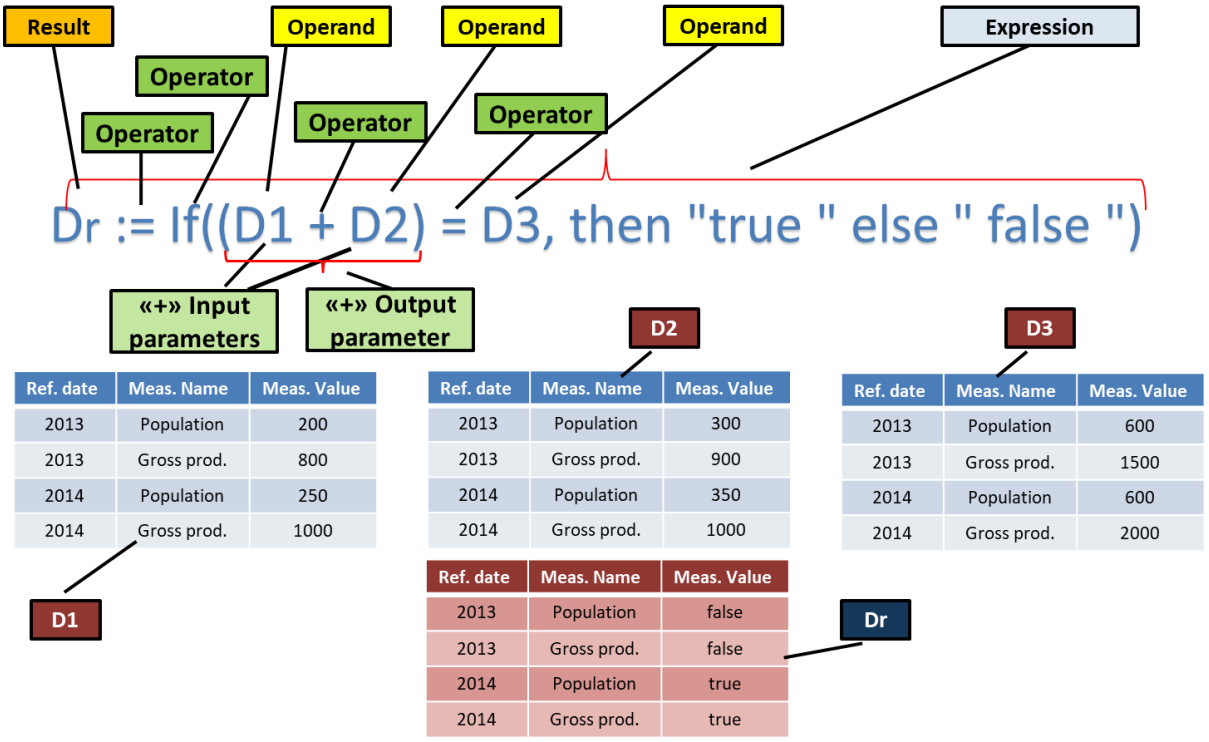
- DDI can capture some of the key elements to define and populate frames and samples. **Universe** and **Population** define the set of statistical units from which to draw a frame and samples.
- Sample information can be maintained in **Sampling Information Schemes**, which not only contain **Universe**/**Population** but also the **Sampling Plan** (i.e. how the sample is drawn), the **Sample Frame** (i.e. the source for the sample) and the **Samples** themselves.
- Alternatively, the method used for creating the frame and sample (e.g. stratification) may be further specified via **Control Logic** (or **Control Constructs** in DDI-LC) and/or parameterized **Steps**. This method often uses **Data Sets** from previous iterations of the same study or from related relevant studies and relevant **Represented Variables** and **Statistical Classifications**.
- Relevant DDI artefacts highlighted above.

Sub-process 4.3 Run collection

SDMX (Angelo)

- SDMX can provide all the information needed to ensure that all expected data has been collected, both for fixed-frequency processes (e.g. monthly or quarterly) and ad-hoc collected data (e.g event-based). These metadata are aimed to define who-has-to-send-what-and when, in terms of which portion of data has to be sent by a respondent (e.g. some respondents will not be obliged to send all data defined in a DSD) or with different frequency (e.g. some reporting agent could send data less frequently than others) or within various time spans (e.g. one or two months after a given reference date). Base artefacts, linking data structures to data provider are already defined in the SDMX Information Model (https://sdmx.org/wp-content/uploads/SDMX_3-0-0_SECTION_2_FINAL-1_0.pdf); some more structures and attributes (e.g. related to frequency) can be added in order to define a complete set of provisional agreement metadata. VTL can be used to check if a given data flow sent by a reporting agent is expected, is on time and/or is complete according to the agreement metadata previously defined.
- Besides possible technical checks, for example related to security (antivirus, decryption, signature verification) and format (unzipping, file type), all verifications against DSD (field format, valid codes, constraints, mandatory values) intrinsic in SDMX could be also be implemented in VTL (in some organisations these type of checks are named “formal”).
- After executing formal ones, so-called “plausibility” checks can be applied; these can include cross validations within the received data (e.g. coexistence/mutual exclusion of two information items, aggregated data comparisons) or with other available data (e.g. time-series or behavioural checks). Also these categories can be implemented

using VTL, provided that metadata are available to describe all referenced data. Below is an example of a VTL check with aggregated data:



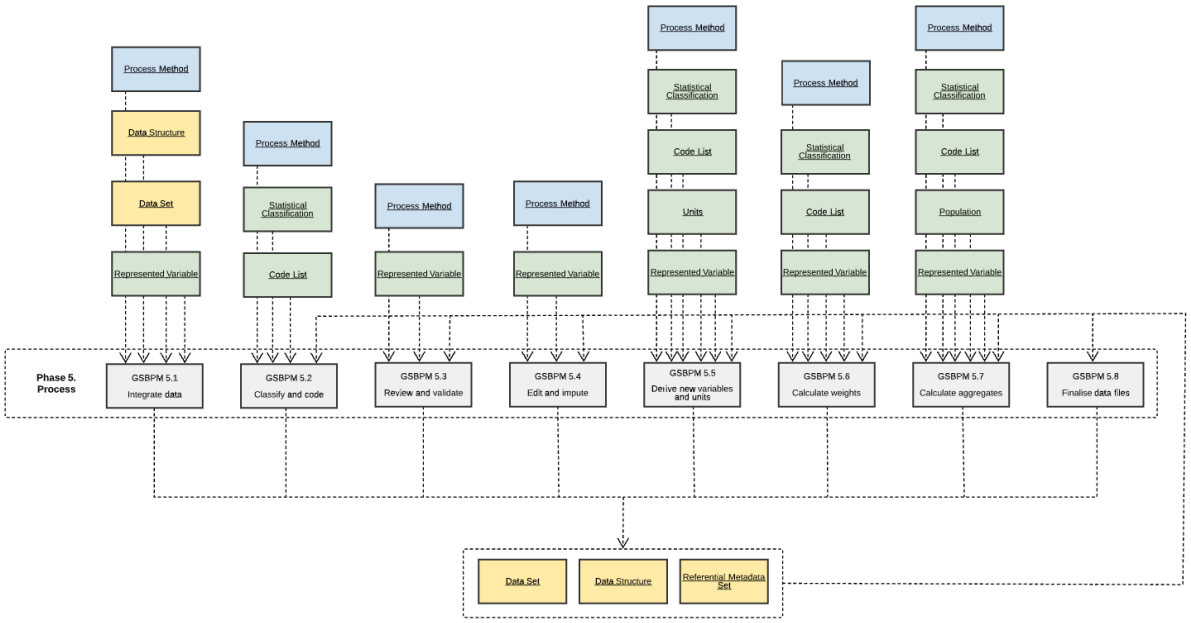
- Some data quality management metadata are recommended to define behaviours to apply in case of errors. For example it is important to have a seriousness attribute for each check (e.g. warning, error, fatal) in order to define thresholds to decide the level of rejection (no action, erroneous information item or whole received file) and to determine the quality of collected data (eventually used to automatically calculate an attribute to attach to it). Further DQM metadata could be defined in order to drive the process of interacting with a reporting agent (e.g. sending feedback and/or requests for re-submissions), applying automatic corrections on wrong data, releasing collected data to the following steps (e.g integration). As in most cases in SDMX, these types of structures, with the described semantics, can be defined with the needed dimensions/measures/attributes.
- In case data from one or more reporting agents is missing, a completeness check on provisioning metadata (agreements), implemented in VTL, could trigger a reminder sending process in order to urge late respondents to send data.

Sub-process 4.4 Finalise collection

SDMX (Angelo)

- At the end of the collection process, the outcome of the above described data quality management process can determine the release of collected data to the following steps (e.g integration) or the interaction with reporting agents to provide corrections and/or re-submissions.
- Based on retention policy, metadata can be defined to drive an archiving process to move older data on different devices in order to optimise storage. Also in these case, these type of structures, with the described semantics, can be defined in SDMX with the needed dimensions/measures/attributes.

Phase “Process”



Sub-process 5.1 Integrate data

SDMX

- SDMX provides uniform data formats, cross-domain structural metadata and content-oriented guidelines regardless of the data source (including geospatial), ensuring enabling data harmonization “by design”. StructureSets can be used to define concepts’ mappings for combining, recoding or transforming incoming data. Attributes attached to dimensions or measures can be used to “Prioritise” – they could be used but it's not their purpose, therefore would need a bespoke implementation
- IT infrastructure based on Web Services, the SDMX Registry and the mapping mechanisms used in SDMX can be very useful to support the integration. Now a day, various international organizations integrate information from statistical indicators in dissemination databases and with version 3.0 we expect to extend this practice to the level of microdata inside the statistical offices. The integration of data (and structural metadata) is complemented with the management of metadata reports which deals with Reference Metadata that helps to ensure metadata quality while performing the integration process.
- SDMX Dataflow is helpful in this sub-process as it can help to integrate data from different sources
- Relevant SDMX artefacts: DSD,MSD, Glossary, Code lists, Content Orineted Guidleines
- Note from Eva: Data Structure +Represented Variable DSD
- Data Set Glossary+ Code list+ Prov.Agreement
- Process Method MSD

DDI

- Data integration span several aspects that can benefit from DDI.
- First, data needs to be organized in some way, usually in **Data Sets**, and described in detail, usually with **Data Structures**.
- Second, data needs to be described consistently at three levels of detail: conceptual, representational and physical (formats). These three levels are captured in DDI by the variable cascade: **Conceptual Variable**, **Represented Variable** and **Instance Variable** (or simply **Variable** in DDI-LC**)**. The first two types of variables have been introduced elsewhere (e.g. see GSBPM 2.2). To the **Conceptual Domains** (e.g. category sets), **Value Domains** (e.g. classifications and code sets) and **Units of Measure** (e.g. distance, currency, weight, etc.) provided by the **Conceptual** and **Represented Variables**, the **Instance Variables** add the notions of **Physical Datatypes** (e.g. ISO or XSD date, VARCHAR strings, int numbers, etc.) and **Sentinel Value Domains** (e.g. platform-specific codes for notions like “null”, “unspecified”, “not applicable”, etc.). All these elements need to be captured to perform data integration successfully.
- Third, mappings may have to be established between the variables used to integrate the **Data Sets/Structures**, often via an **Instance Variable Map** (or **Comparison** in DDI-LC).
- Finally, the data integration method (e.g. probabilistic/deterministic linkage, entity resolution, matching approaches, etc.) may have to be specified, usually with some form of **Control Logic** (or **Control Constructs** in DDI-LC) and/or parameterized **Steps**.
- (Relevant DDI artefacts highlighted above.)

Sub-process 5.2 Classify and code

SDMX (eva)

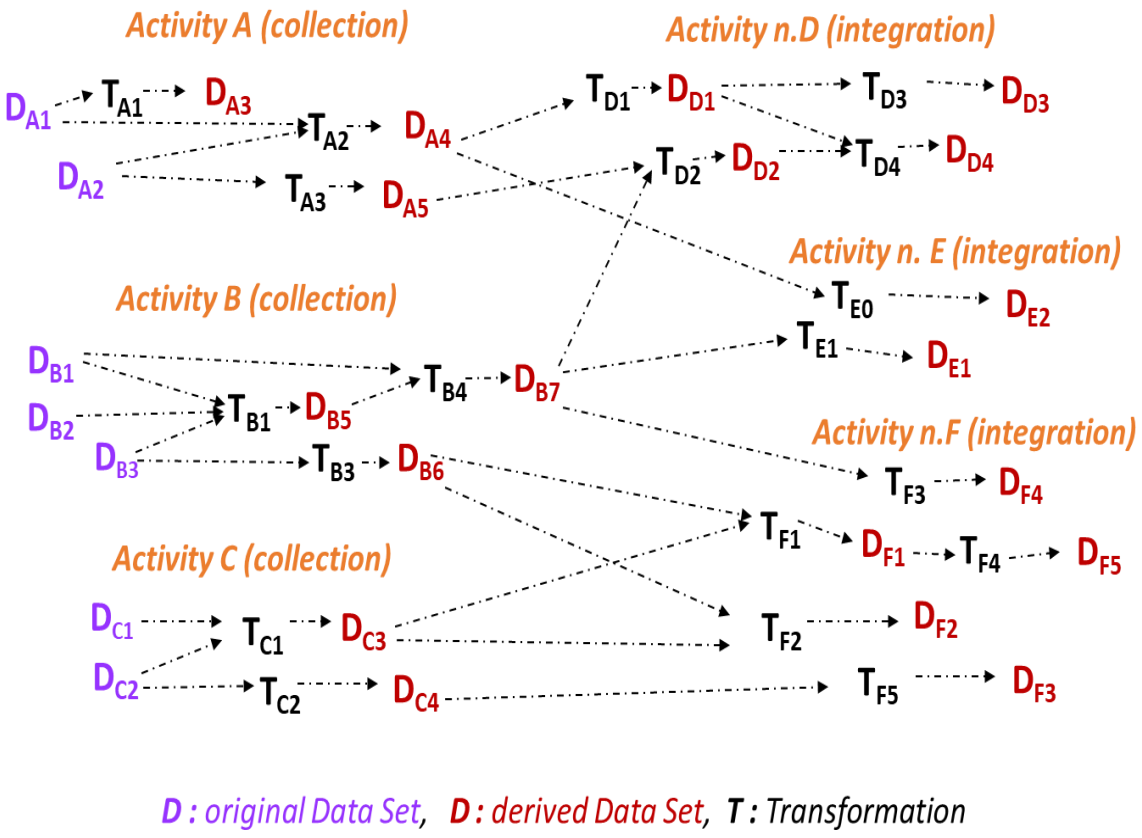
- Mapping is an organized coding mechanism. The DSD includes the ID, the codes, the elements of the code lists. We can link to the sub-process 2.2, where the codelists are planned.
- Related Artefacts: DSD, ConceptSheme, Codelist

Sub-process 5.5 Derive new variables and units

SDMX (Angelo)

- One of the main purposes of the integration phase is to calculate new statistical information from collected data. For this goal VTL can be used to build a Direct

Acyclic Graph of transformations, in order to get also data lineage of the derived data. Here is an example of such DAG:



Sub-process 5.6 Calculate Weights

DDI (Flavio)

Weights can be represented in DDI when describing both a dataset and its associated provenance information.

A DDI **Variable** provides an i**sWeight** Boolean attribute that can be set to true when the variable functions as a weight.

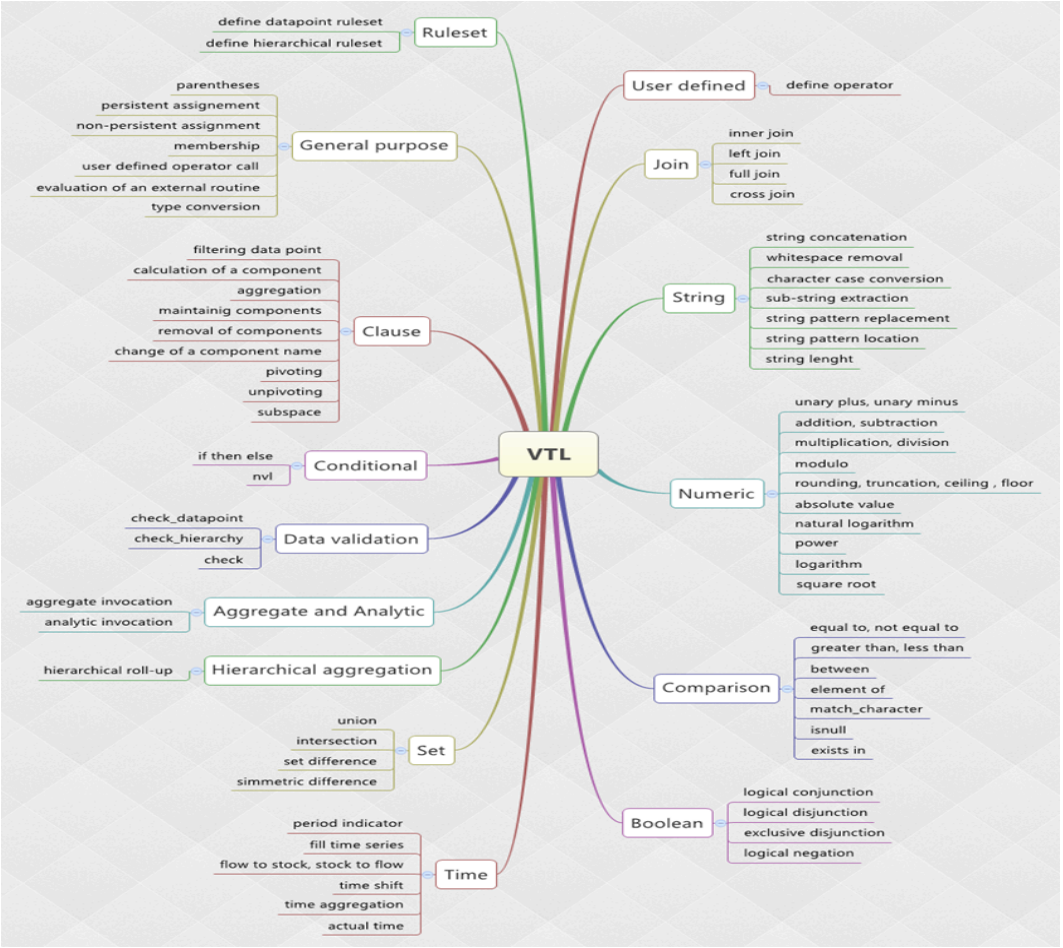
In addition, details about the weighting mechanism and associated methodology can be captured in the associated **Weighting Process Reference**, both in the **Variable** itself and in the associated **Processing Event** describing the provenance information. This **Weighting Process Reference** includes the **Type of Weighting**, the **Weighting Methodology**, the **Analysis Unit** (e.g. individuals, families, institutions, etc.), the **Sample** that was the basis for

the weighting methodology, and a **Standard Weigh** factor used by all or a subset of variables in a dataset, when appropriate.

Sub-process 5.7 Calculate aggregates

SDMX (Angelo)

- VTL offers a comprehensive set of operators in order to calculate aggregates (a list of them by categories is available below). All multidimensional functions are defined in the current version of the language (selection, projection, join..).
- Aggregated measure values can be obtained using all available statistical operators; given the specific composition rules, also derived attributes can be calculated (e.g. status/quality indicators like provisional, estimated, ultimate, revised…)
- Furthermore, given the possibility to introduce user defined operators, the language is “open” in order to allow any transformation.



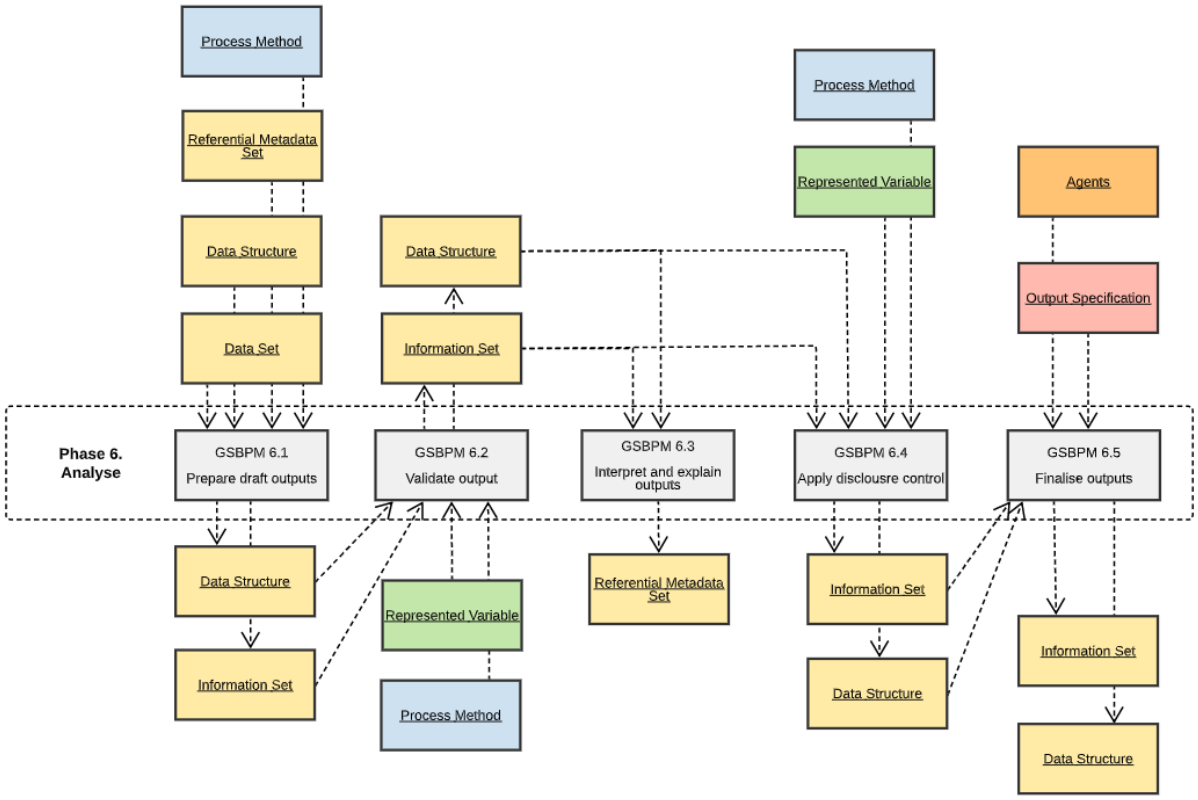
DDI (Flavio)

- Both unit and dimensional data sets and their associated data structures can be captured in DDI. Unit data can be captured in DDI in **Datasets** and **Physical Data**

Products. Dimensional data can be captured in DDI in **Ncube Instances** and **NCubes**: the former are dimensional datasets whereas the latter are their structures. **Ncube Instances** are essentially matrixes where each cell is associated with a measure or dimension in the **NCube**. Data and data structures, both unit and dimensional, can be stored together via the use of **Data Items** and **RecordLayouts**. The **Variables** and **Concepts** describing the data are captured in **Logical Products**. All these provides the essential machinery to describe data at different levels of granularity.

- DDI also provides a description of provenance and how the data associated to each **Variable** can be created or transformed. Each Variable can have a **Source Parameter Reference** that connects with the DDI process model where workflows can be defined via **Control Constructs** and **Commands** to express all sort of transformations and aggregation functions present in usual ETL (extract, transform, load) processes.

Phase “Analyse”



Sub-process 6.2 Validate outputs

SDMX 6.2 (Pascal)

The quality of produced outputs can be determined at the following levels :

Structural : do output data files fit in the designed structures and do codelist items occur in the data that is under inspection. For these purposes mostly **Data Structure Definitions**, **Codelists** and **Content Constraints** are used.

Data : Validate the metadata to ensure that it is complete, accurate, and consistent with the data. This mostly will be done at observation level. For each observation (**OBS_VALUE**), defined in the **Data Structure Definition**, data type and restrictions can be applied.

Domain knowledge : produced datafiles can have a dependency to certain domain knowledge needed for the specific data which is bound to be published. This knowledge is mostly captured in rules or rulesets. SDMX does not specify any rules, but supports **Annotations** to add specifications or even rules that can be checked by external tooling. The annotations can be added on **Data Structure Definitions, Data Flows** or **even Code Lists** and **Content Constraints** if needed**.**

The syntax of these rules depends on the tooling available. Automated tooling can be used, like implementations of VTL or commonly used libraries available for programming languages like R or python. Some domain knowledge cannot be captured in rules and therefore some quality checks remain a manual activity. Manual quality checks and validations are difficult to measure, and therefore not really usable to fit in a metadata model.

Other validations like t-1 checks or measure-interval checks are much more suitable for automated processes with large amounts of measures.

Validation results resulting from performed validations can be used in workflows to accept or decline the SDMX-data to the following step(s) in the process.

Discussion needed : does SDMX support a feature to store these validation results ?

DDI (Flavio)

6.2 Validate outputs, again DDI constructs like **Data Sets** and their **Data Structures** together with the **Represented Variables** of the outputs to be validated seem to apply.

Related Other Material and **Other Material Scheme** can be used as reference for prior cycles or related studies for validation.

The design of the validation process itself could be captured in **Content Review Activity**, which encapsulates things like **Desired Outcome** (metrics, etc.) and **Process Summary** (the actual review process), which can contain a high-level description of the validation process method. More details can appear in **Methodology**.

Sub-process 6.3 Interpret and explain outputs

SDMX (Gabriele)

Interpretation of the outputs concerns the clarity dimension and there are two main ways to approach this matter: reference metadata and data visualizations. Both are supported by SDMX.

- As with data flows and DSDs, metadata flows can be linked to Metadata Structure Definitions (MSD) to organize a wide range of metadata concepts in a structured format. The MSD defines the selection of concepts to be reported, their format and their role.
- Reference metadata are usually high-level structures that can refer to a single or multiple datasets and can be acquired separately from them.

- As an example, the current ESS standard for quality reporting, SIMS, include metadata concepts and sub-concepts from the SDMX glossary, mostly cross-domain concepts.
- Examples of relevant concepts: ACCURACY, CONTACT, QUALITY_MGMNT, STAT_CONC_DEF. Relevant artefacts: MSD, Guidelines, Glossary.
- Data visualizations can accompany the outputs and offer insights for interpretation even for internal use to the statistical agency itself and not just the public. SDMX allows to combine information from multiple sources to build appropriate visualization tools.

DDI

DDI (Christophe)
This sub-process is where the in-depth understanding of the outputs is gained by statisticians. They use that understanding to interpret and explain the statistics by assessing how well the statistics reflect their initial expectations, viewing the statistics from all perspectives using different tools and media, and carrying out in-depth statistical analyses such as time-series analysis, consistency and comparability analysis, revision analysis (analysis of the differences between preliminary and revised estimates), analysis of asymmetries (discrepancies in mirror statistics), etc.

In DDI It is possible to define a **LifeCycleEvent,** associated with the **Archive** module**,** that documents an event in the life cycle of a study. It is any event that is judged to be significant enough to document by the agency. A lifecycleEvent can be described by an **EventType** that supports a controlled vocabulary. A interesting controlled is the following for the type of lifecycleEvent DDI Controlled Vocabulary for Lifecycle Event Type, which includes a large portion of the GSBPM, e.g. study and instrument design, data collection, various stages of data processing, analysis and evaluation, and overarching processes like metadata management.

In particular in this controlled vocabulary we could use the category **DataAnalysisReports** to describe this subprocess.

From the lifecycleEvent we can add a **Description** of the task and make a reference to an **Agent** that can be a **person or organisation that makes the report**.

a **UserAttributPair** could be also specified (as a key value reference) to point to the readable document.

At the same time, it is possible to make a **Relationship** to **QualityStatement** to precise the quality standard that is associated with this task. Here the quality standard referenced could be the code of practice principle number 12.

6.4. Apply disclosure control

This sub-process ensures that the data (and metadata) to be disseminated do not breach the appropriate rules on confidentiality according to either organisation policies and rules, or to the process-specific methodology created in sub-process 2.5 (Design processing and analysis). This may include checks for primary and secondary disclosure, as well as the application of data suppression or perturbation techniques and output checking. The degree and method of statistical disclosure control may vary for different types of outputs. For example, the approach used for microdata sets for research purposes will be different to that for published tables, finalised outputs of geospatial statistics or visualisations on maps.

SDMX (Edgardo)

The *Guidelines for Confidentiality and Embargo in SDMX* explain how to manage the disclosure timing and access authorisation for confidential products.

DDI (Christophe)

In DDI, **InformationClassification** object contains several ways for managing confidentiality. Objects like StudyUnit (or group of studyUnit), the **physicalInstance** of a dataset, or ResourcePackage can include a reference to an informationClassification. Here are some of them :

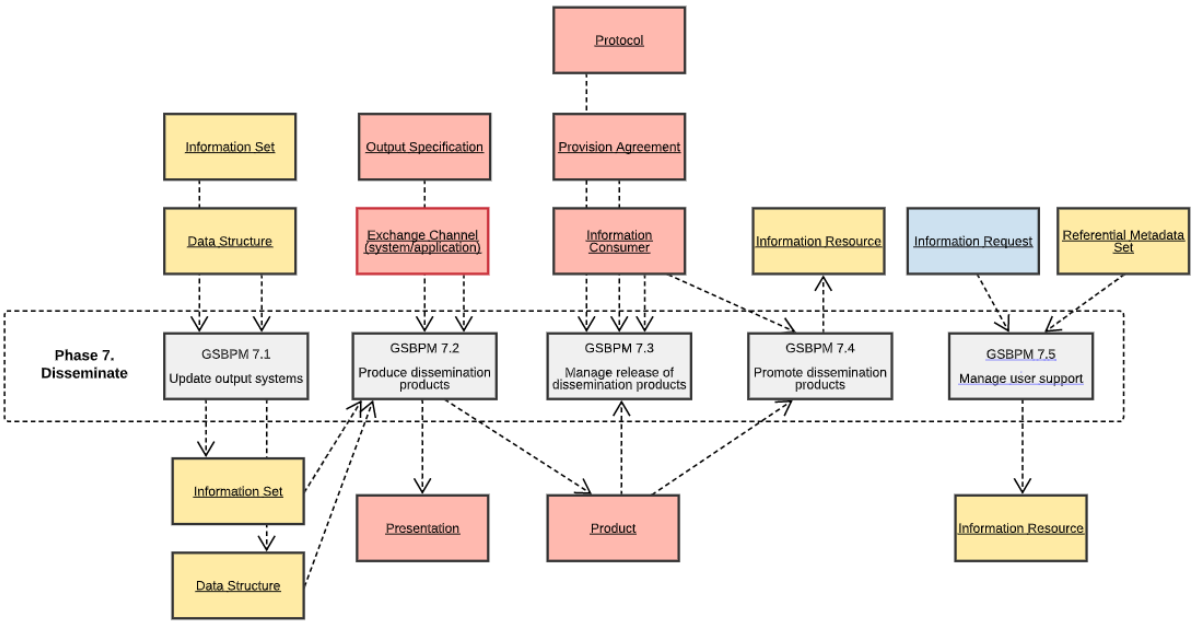
TypeOfInformationClassification and **LevelOfInformationClassification** correspond to the type and level of classification of the data, determined by an assessment of the need for confidentiality of the data. The use of an external controlled vocabulary is supported and strongly recommended.

With **DataHandlingPersonalRules**, it is possible to give a description of the rules applied to anyone accessing the data, for example security clearance, confidentiality agreements or authentication. They can be expressed in several languages and allow the use of structured content.

DataEncryptionRules provide a description of the rules regarding the level of encryption required for the data. Can be expressed in multiple languages and allows the use of structured content.

AuthorizedPolicySource refers more generally to the current policy for the confidentiality treatments.

Phase “Disseminate”



Sub-process 7.1 Update output systems

SDMX (Edgardo)

In case recoding and or reformatting of the data or metadata is required, *StructureMaps, RepresentationMaps* or *ItemSchemeMaps* should be used to establish the rules, which may also be specified using VTL.
(Structure - DSD, Representation - codelist, ItemSchemeMaps-any other)

ContentConstraints provide a way of ensuring the consistency of codes used in the data and metadata being updated.

Linking of metadata to the data is intrinsic to the SDMX Information Model, and no special considerations are required in this regard if a proper data model has been designed and implemented.

The actual loading and update of the dissemination databases can be implemented through the SDMX API by executing VTL scripts, or submitting data or metadata sets including *SDMX Actions* for data updates. → **Soon to be released**

SDMX (Pascal, Under construction)

Adding, merging, deletion or updates of data to SDMX based databases can be done by either SDMX-ML or SDMX-CSV formatted datafiles, which both support the required actions.

Releasedate (aka Point-in-Time release) can be applied, but is not part or enforced by the

SDMX 2.1 datamodel. Data will be released and embargo is lifted only after releasedate has expired.

Sub-process 7.2 Produce dissemination products

SDMX (Edgardo)

Many of the dissemination products are likely to be produced with tools able to fetch the data and reference metadata from the dissemination database using the SDMX API. The format and visual characteristics of these products can be specified in an standardised way following the *Guidelines on using SDMX Annotations* and *Standardising Reference Metadata Reporting in SDMX,* published by the SDMX Statistical Working Group.

(Validation: Integrity and non-disclosure)
(Data + Metadata)

Sub-process 7.3 Manage release of dissemination products

SDMX (Edgardo)

SDMX provides the *Subscription and Notification Service* of the SDMX Registry to manage the release of dissemination products, which in conjunction with the recommendations in the *Guidelines for Confidentiality and Embargo in SDMX* allows managing the timing and access authorisation for the released products.

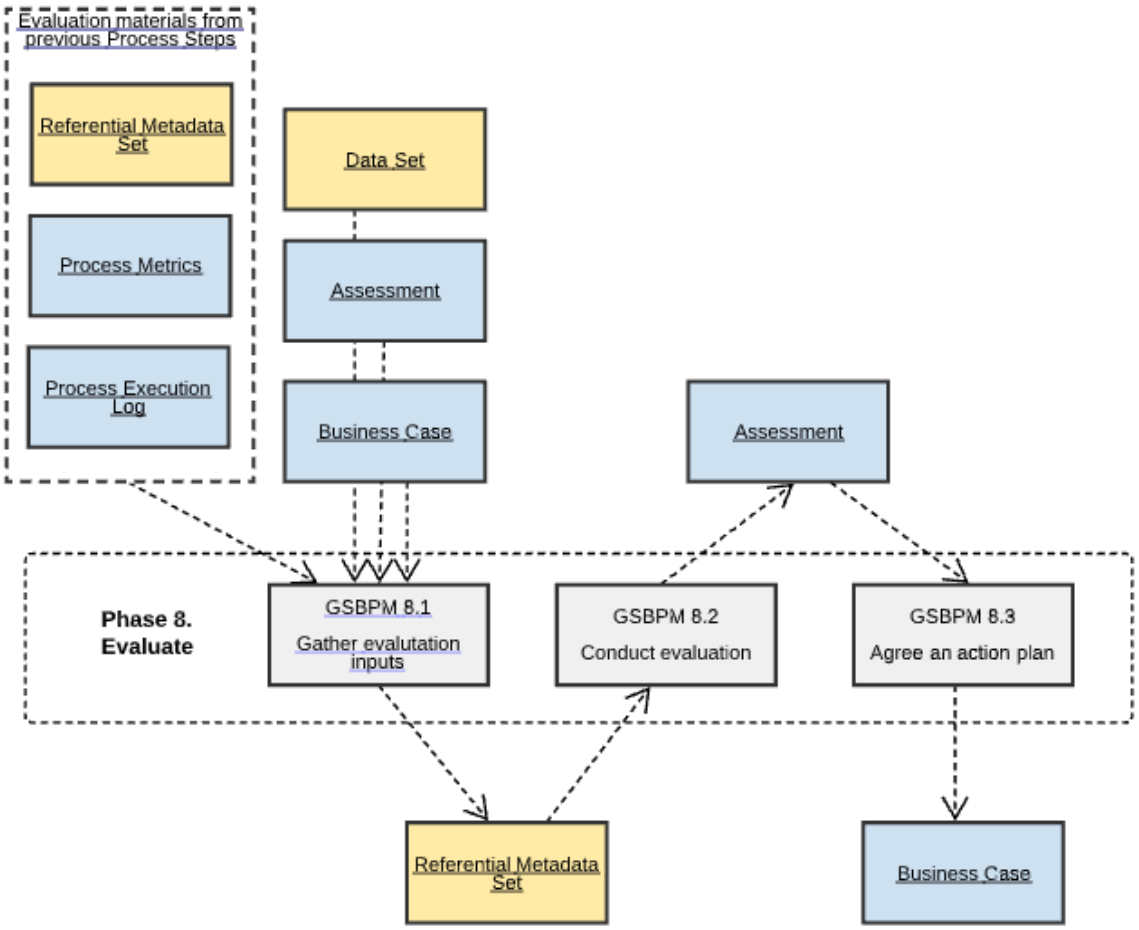
(summarize attributes, add hyperlinks)

Sub-process 7.4 Promote dissemination products

SDMX (Edgardo)

The *Subscription and Notification Service* of the SDMX Registry, and the SDMX API can be used to automatize the promotion actions through customer relationship management tools, as well as posting in websites, wikis and blogs to facilitate the process of communicating statistical information to users. VTL can also facilitate this process automation.

Phase “Evaluate”



8.2. Conduct evaluation

DDI (Christophe)

DDI 3.3 r:ExPostEvaluation

In DDI, the object **ExPostEvaluation** allows evaluation for the purpose of reviewing the study, data collection, data processing, or management processes. Results may feed into a revision process for future data collection or management. Identifies the type of evaluation undertaken, who did the evaluation, the evaluation process, outcomes and completion date.

It is possible to describe theTypeOfEvaluation realised, the EvaluationProcess, and also identify Evaluators and their roles.

GSIM Mapping

ExpostEvaluation (DDI) - Assessment (SDMX)?? (to be double check with UML)

8.3. Agree an action plan

DDI (Christophe)

DDI 3.3 r:ExPostEvaluation
Outcomes*

The Object **ExPostEvaluation** also refers to the **outcome**, that is the result of this evaluation process, thus in particular an action plan for further data collection. It may be expressed with a controlled vocabulary,

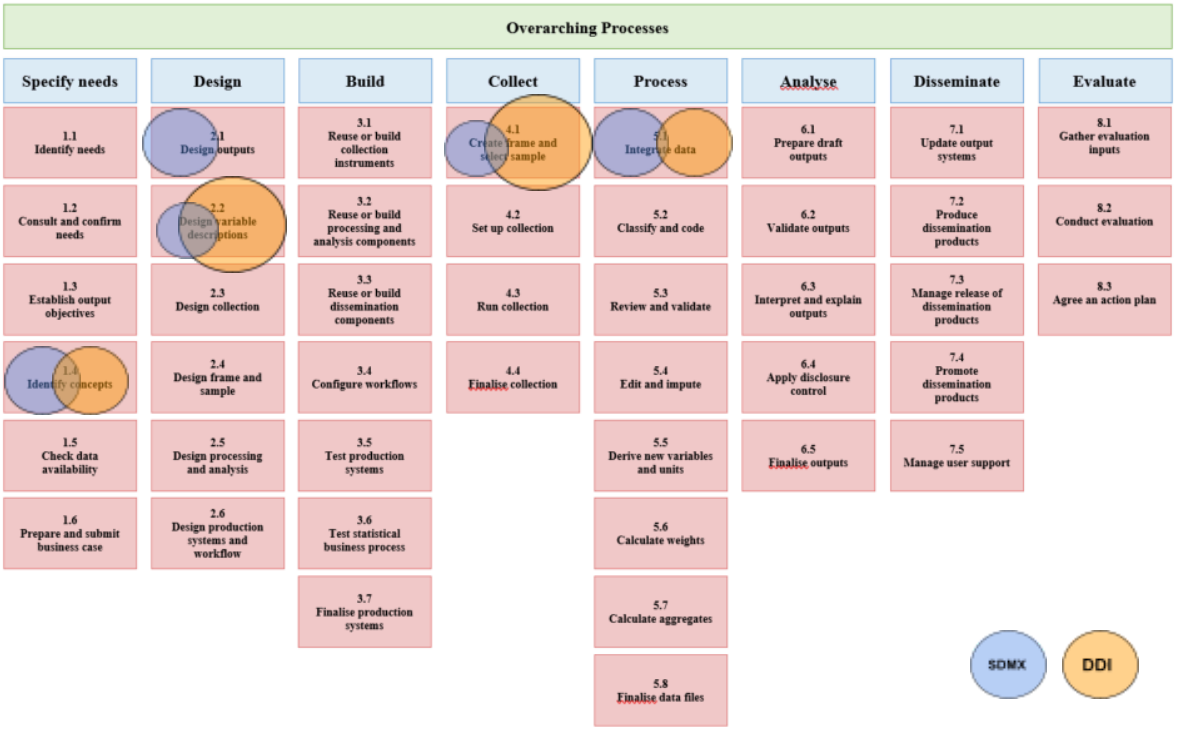

Overarching Processes
Specify needs
1.1 Identify needs
1.2 Consult and confirm needs
1.3 Establish output objectives
1.4 Identify concepts
1.5 Check data availability
1.6 Prepare and submit business case
Design
2.1 Design outputs
2.2 Design variable descriptions
2.3 Design collection
2.4 Design frame and sample
2.5 Design processing and analysis
2.6 Design production systems and workflow
Build
3.1 Reuse or build collection instruments
3.2 Reuse or build processing and analysis components
3.3 Reuse or build dissemination components
3.4 Configure workflows
3.5 Test production systems
3.6 Test statistical business process
3.7 Finalise production systems
Collect
4.1 Create frame and select sample
4.2 Set up collection
4.3 Run collection
4.4 Finalise collection
Process
5.1 Integrate data
5.2 Classify and code
5.3 Review and validate
5.4 Edit and impute
5.5 Derive new variables and units
5.6 Calculate weights
5.7 Calculate aggregates
5.8 Finalise data files
Analyse
6.1 Prepare draft outputs
6.2 Validate outputs
6.3 Interpret and explain outputs
6.4 Apply disclosure control
6.5 Finalise outputs
Disseminate
7.1 Update output systems
7.2 Produce dissemination products
7.3 Manage release of dissemination products
7.4 Promote dissemination products
7.5 Manage user support
Evaluate
8.1 Gather evaluation inputs
8.2 Conduct evaluation
8.3 Agree an action plan
SDMX
DDI

Figure 4: Combining the GSBPM, DDI and SDMX