UNITED NATIONS ECONOMIC COMMISSION FOR EUROPE

CONFERENCE OF EUROPEAN STATISTICIANS

**ModernStats World Workshop 2022**

27-29 June 2022, Belgrade, Serbia

# Piloting surveys with GSIM

*Sandra Déprez sandra.deprez@insee.fr, Christophe Dzikowski christophe.dzikowski@insee.fr, Benoît Werquin benoit.werquin@insee.fr*

For ten years now, Insee has been developing a survey data collection platform using international standards in a metadata-driven approach. This work led to the development of tools based on standards like DDI[1] and VTL[2] that allow to specify and generate multi-mode survey questionnaires.

Today, Insee is extending the use of active metadata beyond the data collection; the ambition is to use metadata to:

- configure complex survey protocols (multimode, one or several sub-cycles of collection, different collection instruments, etc.);
- be able to trigger collection events (initiate / close collections, send letters or emails, change the collection mode related to a unit, etc.);
- implement the first stages of processing the data collected.

To configure complex survey protocols and trigger collection events, the data collection platform currently in production needs to be enhanced with a new tool called Protools.

To implement Protools, investigations are being carried out by the statistical and technical teams. In this process, GSIM helps to identify and define the necessary conceptual objects.

## 1. At the origin of the needs

### 1.1. A long-term strategy: extending the use of metadata to the entire data collection system

For ten years now, Insee has been developing a survey data collection platform using international standards in a metadata-driven approach[3].

Several components covering different phases of the GSBPM have been developed one after the other:

- **Design**: a questionnaire design tool (Pogues);[4]
- **Build**: a questionnaire generator (Eno);[5]
- **Collect**: a CAWI data collection and management platform for businesses surveys (Coltrane), a CAWI data collection and management platform for household surveys (Coleman), a CATI / CAPI data collection and management platform for household surveys (Sabiane), a PAPI management tool (Sting);[6]

The components dedicated to questionnaires are based on the use of the DDI and VTL standards.

Today, Insee is pursuing the modernisation of its data collection system with the "Metallica" project.

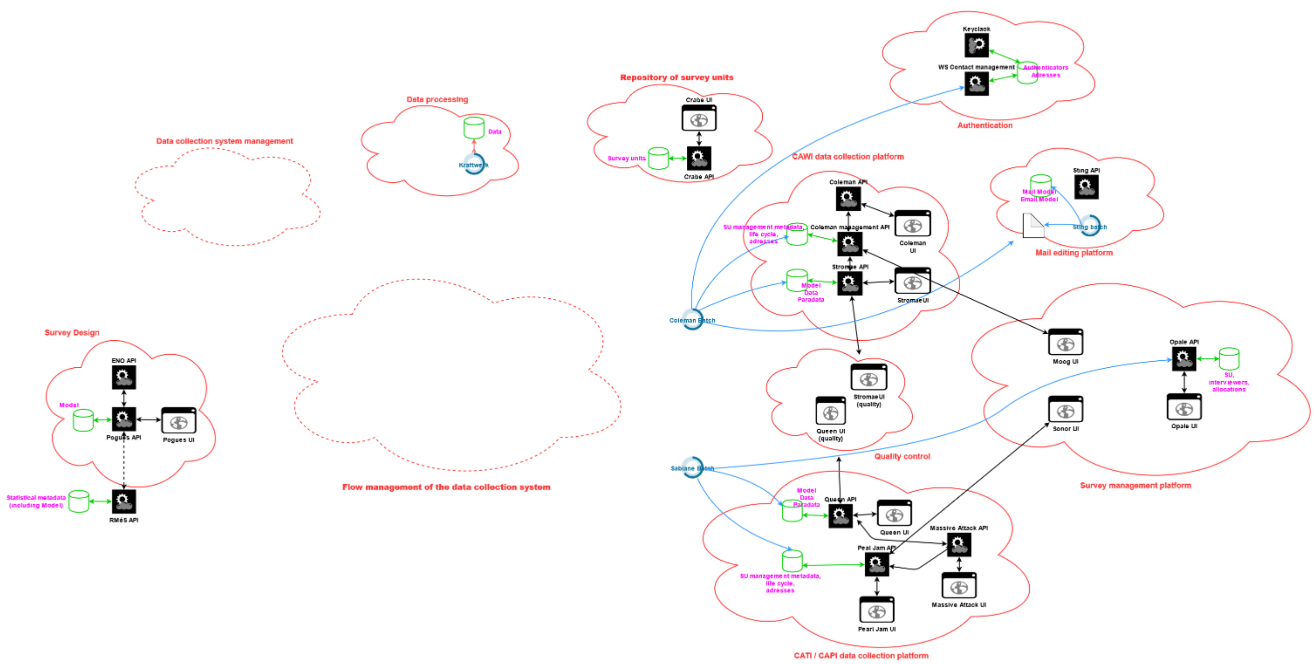
This project, is in line with our long-term strategy. Indeed, the aim of Metallica is to extend the use of active metadata beyond the data collection to:

- configure complex survey protocols (multimode, one or several sub-cycles of collection, different collection instruments, etc.);
- be able to trigger collection events (initiate / close collections, send letters or emails, change the collection mode related to a unit, etc.);
- implement the first stages of processing the data collected.

### 1.2. A data collection system with a service-oriented architecture (SOA)

Today, the data collection platform has multiple services dedicated to the implementation of surveys. The following figure gives an overview of the current system.

**Figure 1 The data collection platform today**




But surveys often have complex protocols (e.g. several modes and sub-cycles of collection and collection instruments) and this involves numerous tasks that require data manipulations and transfers.

Thus, such a modular data collection system needs a platform to manage the tasks that need to be done to implement the survey protocols.

The Metallica project team therefore aims to implement a new tool, called Protools, for orchestrating the data collection system.

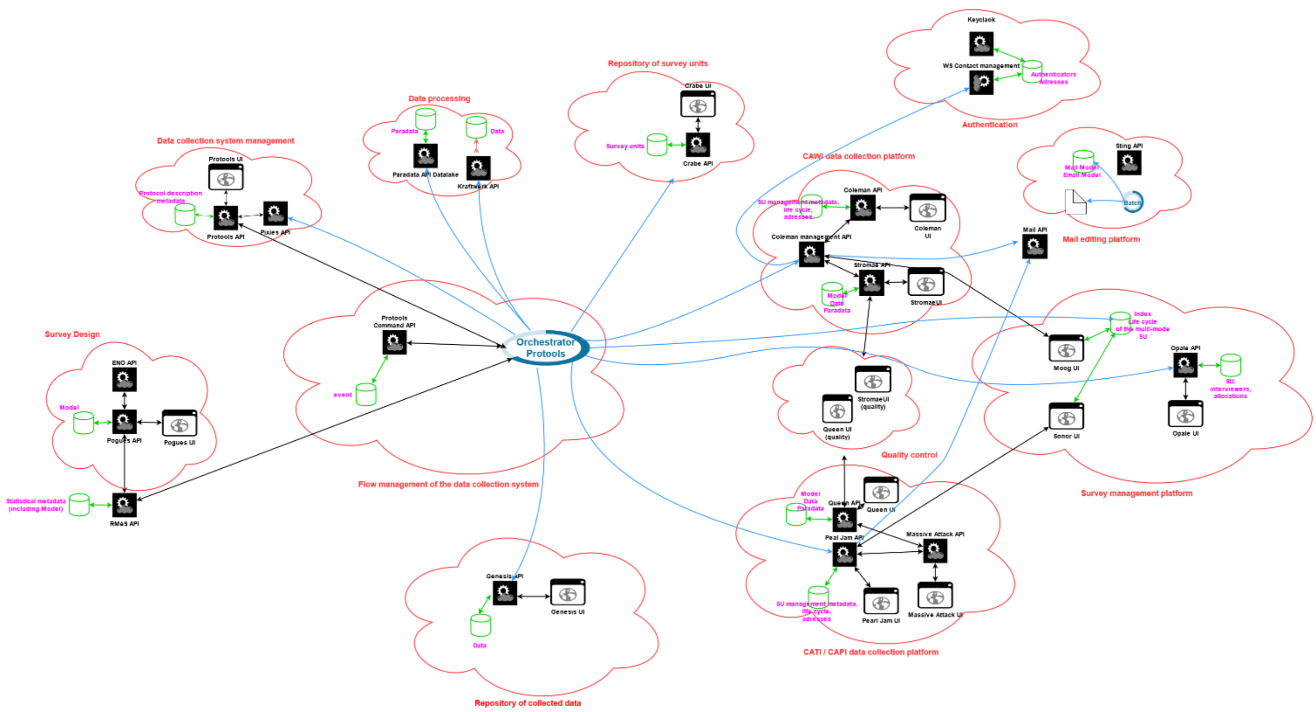
With Protools, the data collection system will further evolve to:

- enable communication between the various components;
- improve the automation of the implementation of surveys;

- ensure overall consistency from the design to the data processing phases thanks to the principle of active metadata.

With Protools, the system evolves as shown in the figure below.

**Figure 2 The data collection platform with Protools**




In practice, the main functions of the Protools tool are to:

- trigger collection events;
- expose / fill in survey metadata;
- monitor the triggering of collection events;
- manage users rights and roles on the various tools.

## 2. Building Protools: investigations from the statistical team

The construction of Protools required a first step of investigations from the statistical team. These investigations were carried out in parallel with those of the IT team so as not to restrict or influence the thinking.

The main objective of Protools is to trigger survey collection events. The investigations must therefore first focus on identifying conceptual objects:

- "static" to characterise surveys;
- "dynamic" to trigger the collection events.

### 2.1. Characterising surveys

Insee has a statistical metadata repository, RMéS[7], which structures statistical information in a centralized and coherent way.

Two types of metadata are described in RMéS using international standards:

- transversal metadata (concepts and nomenclatures);
- metadata related to statistical processes (description, questionnaires, variables and code lists).

These metadata are the reference and are easily reusable by users and client tools.

Thus, in the case of Protools, the question is: is the modelling of surveys protocols (objects and associated metadata) proposed by the RMéS metadata management system adapted and sufficient?

To investigate this issue, an analysis is conducted using two use cases: a simple one from the consumer survey, and a more complex one from the housing survey.

The result of this analysis is that in the case of complex survey protocols, the high-level objects and associated metadata needed to configure the statistical processes supported by Protools are not sufficient.

Indeed, the fine management of the process, the monitoring and organisation needs, require a low level object closer to operational considerations.

This low level object can be defined like this: any operational sample partition necessary for the management and monitoring of homogeneous units in terms of characteristics (same dates, interview rank, collection mode, collection instrument, quality of response to a previous collection, etc.).

We give below two examples of such sample partitions in two cases, one simple and the other more complex.

*The Consumer survey use case*

| | **Consumer survey** |
|---|---|
| Frequency of collection | Monthly |
| Statistical unit | Household |
| Collection mode | CATI |
| Sampling method | 3 months rotating panel. Surveyed units are selected from tax sources and have a telephone number |
| Collection period | 3 weeks |
| Questionnaire | Short - An 'opinion' module (general economic situation in France, own financial situation, saving and spending intentions) to which is added, only in the first interview, a socio-demographic module. In some months, the survey is also used for a complementary module comprising a maximum of twenty questions on specific themes such as the environment, well-being and housing conditions. In case of additional module, the questions are asked of all surveyed units. |
| Sample | Each month 3 types of units:<br>- the 1st interviews (1250 units);<br>- the 2nd interviews;<br>- the 3rd interviews. |
| Criteria for selecting units for sample operationals partitions | - 1st interview: all sampled units<br>- 2nd interview: responding units in 1st interview<br>- 3rd interview: responding units in 1st interview minus refusing units in 2nd interview |

For this survey, on a given month, the sample operational partition model is simple:

| Sample operational partition | Description |
|---|---|
| 1 | 1st interview units |
| 2 | 2nd interview units |
| 3 | 3rd interview units |

The only operational need is to be able to distinguish in the interviewers' collection tool between units according to their interview rank. This distinction makes it possible, for example, to adapt the organisation of the work and the contacts with the respondents.

For this survey, a consolidated monthly collection monitoring of the 3 partitions (1 + 2 + 3) is enough for the people in charge of managing the interviewers' activity.

*The Housing conditions survey use case*

| | Housing conditions survey |
|---|---|
| Frequency of collection | Punctual |
| Statistical unit | Housing |
| Collection mode | Sequential multi-mode: CAWI / CATI / CAPI |
| Sampling method | The surveyed units are selected from tax sources and are primary residences |
| Collection period | Several separate collection phases between August 2023 and July 2024 |
| Questionnaire | Long - full and detailed description on many sub-themes related to housing (the stock and its occupants, access to ownership, housing expenses, poor housing, residential mobility…). In order to make possible the CAWI and CATI collection, this long questionnaire is divided in 3 parts (sequences). For the CAPI collection the questionnaire covers the three sequences |
| Sample | - 2 * 20 000 units for the 1st CAWI sequence<br>- 2 300 units for the CAPI control sample |
| Criteria for selecting units for sample operationals partitions | - the units belonging to group 1 or 2 for Internet collection<br>- non-respondent Internet<br>- Internet response quality score<br>- non-respondent telephone |

For this survey, the sample operational partition model is much more complex; this is detailed in Annex 1. Indeed, 27 partitions of the initial sample are identified to implement the survey protocol. This is due to:

- the division of the questionnaire in 3 sequences for the CAWI and CATI collections;
- the need of a CAPI control sample;
- the differents scenarios for changing modes for non-responding units and bad quality responses at the CAWI collection;

- the need to spread the workload for the Insee interviewers (CAWI units are initially divided in 2 groups collected on different dates. This means implementing the entire protocol survey twice: with the group 1 units then with the group 2 units);
- the need for interviewers to be able to easily identify the life cycle of the units loaded into their collection tool in order to adapt, for example, the contact with respondents.

Moreover, a unique collection monitoring of these 27 partitions does not make sense. Indeed:

- the monitoring of the CAWI and the CATI/CAPI collections phases are not handled by the same actors;
- the partitions for a given mode are not necessarily deployed on the same dates.

In total, there are 13 monitoring phases due to the different modes and dates of collection.

### 2.2. Triggering the events

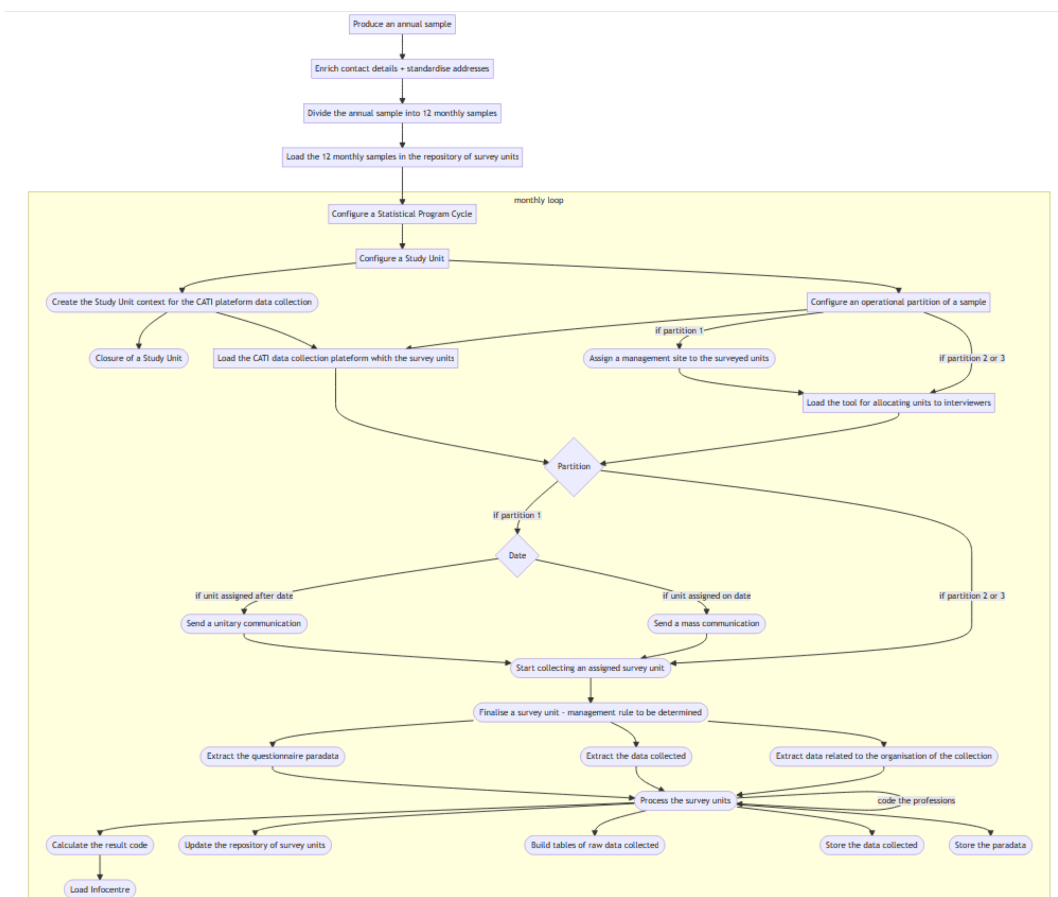
Protools' main function is to trigger the events for the implementation of the survey protocols.

Thus, the issue of modelling the events and the metadata required for their implementation arises.

To identify the objects and metadata useful for triggering the events, an analysis in three steps is carried out using the consumer survey.

The first step is to produce a flow diagram. This diagram represents, in an ordered manner, all the events that characterise the consumer survey protocol.

**Figure 3 Flow chart of the consumer survey protocol**



The second step is to complete this diagram with a sequence diagram which indicates the actors and tools involved in the implementation of each event.

Finally, the last step is to characterise each event by indicating:

- the triggering method (human or automatic);
- the metadata uses for the implementation (including selection or processing rules applied) and their origin;
- the result of the triggering of the event.

Here is an example with the event "Configure a Statistical Program Cycle":

- triggering method: human
- metadata uses:
    - metadata specified by Protools' user:
        - label of the statistical program
        - Rule for allocating the survey units to the management sites
        - Contacts in the management sites (e-mail and phone number)
        - Persons in charge of organising the survey (surname, first name, phone number)
        - E-mail of the team in charge of organising the survey
        - Rules for selecting units for sample operationals partitions
    - active metadata (from RMéS):
        - Identifier and label of the statistical program cycle
- Result: creating/updating a statistical program cycle in Protools with configuration metadata

# 3. Building Protools: GSIM approach

## 3.1. Characterising surveys

It is worth describing how we currently document statistical processes in our metadata management system. There are two aspects that have contributed to the way we have characterized these:

1. Our description of statistical processes is currently based on the two GSIM objects “statistical program” and “statistical program cycle” to which our internal terminology of “série” (series) and “opération statistique” (statistical operation) correspond respectively.
2. So far, we mostly describe our “opération statistique” with the reference metadata of SIMS (Single Integrated Metadata Structure), which are intended to describe the quality of statistical processes in the context of reports sent to Eurostat. The frequency of sending this quality metadata to Eurostat is, in our use, at most annual, so we had a framework for describing the quality of the processes based on this annual rhythm, that implicitly influenced how we considered statistical processes.

In doing so, we have at the moment no metadata to drive statistical processes, then we have to identify new objects in the standards to describe the process in relative detail. Moreover, surveys do not necessarily follow an annual rhythm; the pace could by either monthly, as in our case of use of the consumers survey, or supra-annual within the framework of a longitudinal survey. As a consequence, this article led us to review our uses of statistical process objects, to check if they are still operational or to re-specify them if necessary.

#### *3.1.1. The existing description of the consumer survey in our metadata management system*

At the time of writing, all the metadata we have about the consumer survey are general fields (summary, labels, survey type, history) that describe the issues associated with this survey at the level we call the "série", i.e., the statistical program. In particular, we have only one occurrence of this series and, therefore, there is no mention of a specific year. This type of reference metadata is not intended to drive a process, and the process obviously needs to be completed with more refined metadata, but the first question here is to determine what the statistical program cycle corresponds to in the consumer survey protocol.

#### *3.1.2. A choice to be made in the modelling*

This may seem surprising, but the above question is not so easy and will require a choice. Indeed, the Consumer survey is technically a panel planned over a year that contains monthly surveys. The panel dimension is used for methodological reasons such as reducing non-response or reducing variance. However, at the same time, the aim of this survey is to produce monthly results and analyses of the statistical outputs never use the longitudinal dimension. In summary, the choice is: 1) we have the statistical program definition and each repetition (consumer survey 2023, consumer survey 2024, etc.) is considered a statistical program cycle. Or 2) the cycles of the statistical program are the monthly surveys and the annual methodology that is shared between the monthly surveys is considered as a methodological support. These two possibilities are described below in the GSIM framework, and we will discuss these different choices.

<u>First case where we consider a panel survey</u>

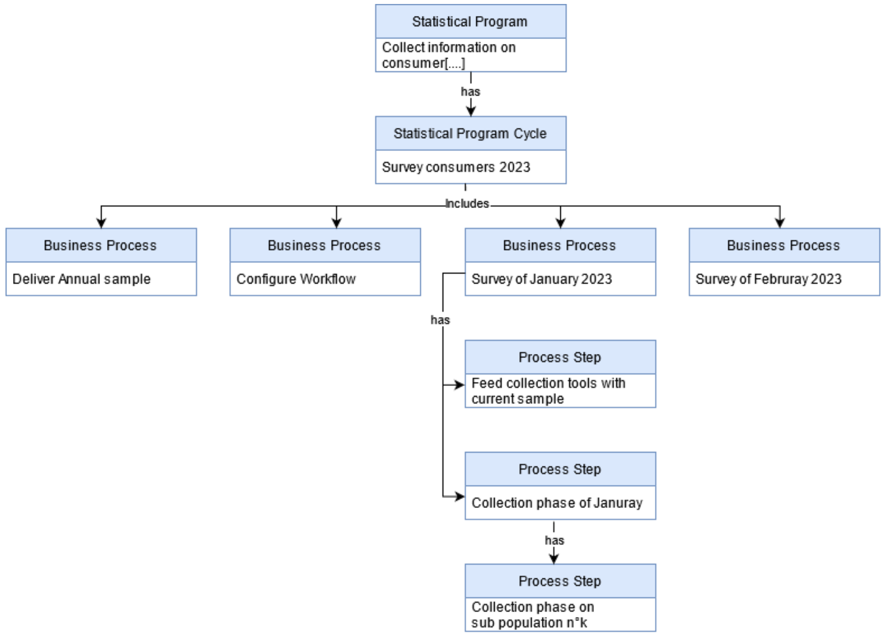
In this first proposal we will call solution 1, we have the statistical program that defines the goals and objectives that are to collect information on consumer behaviour and expectations in terms of expenditure attitudes and savings. This statistical program has a statistical program cycle that is an annual process and the reference period here is the year 2023.

This statistical program cycle includes several business processes. The first business process concerns the shared steps, like sampling or configuring the workflow. The other business processes concern the surveys, one business process for each month.

In practice, a survey may involve several phases of data collection. These can be represented by the process steps. Since each data collection phase may also contain several collection phases for specific sub-populations, these finer levels of data collection could be achieved through the nested nature of the process steps.

In this first case, we thus have a sequence of business processes linked to two levels of process steps that will yield high-level objects required the IT team (one level to express collection phases where tools are set up, and one level for collection on specific sub population for field issues). Because we use process step objects for steps that are not the same nature, these objects should be labeled and managed with relevant vocabularies using attributes inherited from identifiable artifacts and administrative details to identify and qualify these structuring phases of the statistical program cycle.

**Figure 4 Diagram summarizing solution 1**

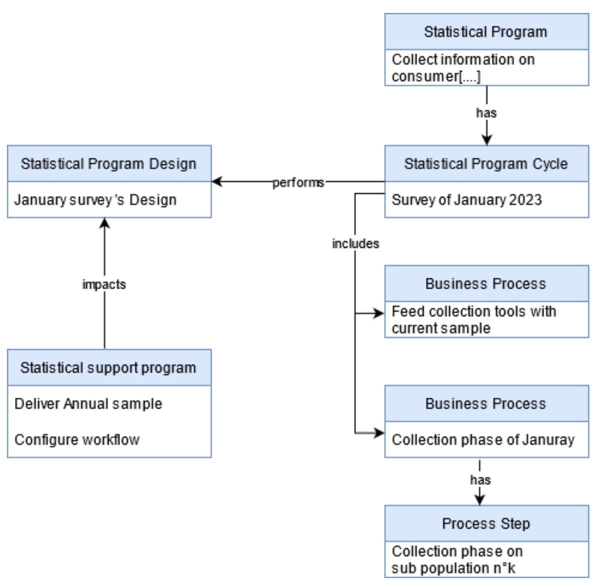



<u>The second case is where monthly surveys are considered to be statistical program cycles</u>

In solution 2, the statistical program is the same as in solution 1, but it has twelve statistical program cycles, one per monthly survey. The common methodological steps (draw the samples, configure the workflow) will be supported by the statistical support program.

Each statistical program cycle has several business processes that are closer to what we would expect from a sub-process of a statistical process. Finer levels of data collection for specific sub-populations will be supported by the process steps. And we can identify and qualify our high-level objects as above.

**Figure 5 Diagram summarizing solution 2**



Discussion and first findings

This work has led us to also delve into how we use the statistical program and the statistical program cycle for complex protocols. Here is a list of related questions or comments that we will continue to pursue:

1. How to interpret the reference period in the definition of the statistical program cycle? Is a range encompassing several collection periods valid with this definition? Should it be only a time interval for one survey?
2. In the solution 1, we use business processes to describe monthly surveys, that is to say object that include in itself the idea of having a complete GSBPM cycle. In the GSIM definition the Business process are more like GSBPM phases. Do we need to stick to this kind of scope for the business processes, as given in the official GSIM examples, or is it possible to use business processes as entire cycle?
3. An interpretive question concerns also the property "relates to" of the statistical program to another statistical program. Is it used to describe the evolution over time of a statistical program, or can it be interpreted as a relationship with another statistical program, meaning that several statistical program can work together?
4. In the solution 2, even if we assume there is no change concerning the goals of the statistical program, it is necessary to define a new "série" for each year, (one for the 2023 survey, one for the 2024 survey, and so on) that enconpasses the monthly surveys. So there is a difficulty at this stage because we create series despite goals remain the same.
5. Whatever the choice we will make specifically for the consumer survey, there remains the issue that a true panel, that encompasses several complementary surveys on the same sample seems not easy to handle with GSIM, assuming we understand it correctly, and of course in our metadata management system. If a complete replication of a panel requires that all the surveys be done, then the statistical program cycle should be all of them. And one of the surveys, by itself, cannot represent a full replication of the program and cannot meet the overall objectives. This particular point may lead us to generalize what is meant, in our terminology, by a "statistical operation" from what was essentially a survey for a given year in practice. To take account of panels, the statistical operation (that is to say our statistical program cycle) could be defined as a group containing several surveys. This point is still under discussion.

The next steps in the work will be to decide what the best option is, and possibly to discuss these issues with the GSIM community to see if the difficulties in describing the panels are confirmed.

It appears that, apart from the discussion of how we can interpret the statistical cycle in a complex case, GSIM offers the possibility of finding objects that are relevant to Metallica's first requirement, i.e. metadata to describe static objects. Thus, for example, data collection, collection on a specific sub-population are expressible with GSIM.

### 3.2. Dealing with events

Up to now we have discussed only the levels of statistical program and statistical program cycles. To be able to pilot processes we need finer objects. The statistical program cycle discussed above operationalizes one statistical program design, that is a key component providing access to high-level objects needed by the Metallica program:

- the business functions that qualitatively specify what we do with our “segmented” steps;
- the process design that gives access to the specification of the rules and methods in and between our steps.

As the Figure 3 above suggests, the process requires several steps such as “deliver the sample”, “configure the workflow”, “run the data collection”, etc. It is convenient to first use business functions to describe qualitatively each of these steps by specifying what we are doing. It would be relevant to map these objects to GBSPM sub-processes or tasks if possible.

With regard to process designs, they allows us to describe controls and methods used during the process. We will use this GSIM object to specify the process steps. Here the logic presented in the document linking GSBPM and GSIM[8]; is very useful and it is this type of usage that we will reproduce in our use cases, the idea being to specify the methods and the input/output rules associated with the design of a process.

As it is presented in the subject matter part, numerous tools are used and need to be coordinated. In GSIM, a business service can be used to mean that a specific application is called and functions as a module, even if the meaning of business service is broader than that.

Two examples are presented below.

The first example concerns the step of enrichment of our data by using the service offered by a contract (a postal service).

| **Process input specification** | **Process design** | **Process output specification** |
|---|---|---|
| <u>Core input type</u>: Dataset that is the main sample loaded in the repository of units to be collected<br><br><u>Business service</u>: Service offered by the contractor (postal service) to enrich the data<br><br><u>Process support input type</u>: Protocol agreed with the contractor for sending the data<br><br><u>Parameter input type</u>: Information that the main sample is loaded in the repository and the enrichment can be executed. | <u>Process method</u>: Specification of the type of enrichment (adresses standardisation, research for phone numbers, research of homonyms) | <u>Core output specification</u>: Updated dataset with enriched values<br><br><u>Process execution log</u>: Information that the sample is updated and can be reloaded in the repository |

Here we use a business service, we could also use an exchange channel and a provision agreement.

<u>The second example</u> presents the process design of the step of splitting the main sample into twelve sub-samples.

| Process input specification | Process design | Process output specification |
|---|---|---|
| Core input type: Dataset enriched in the repository that is to be split in twelve<br><br>Parameter input type: information that the main enriched sample is ready to be split | Process method: Simple random sampling applied eleven times | Core output specification: Twelve Datasets, corresponding to twelve samples<br><br>Process execution log: information that the twelve samples are drawn |

To the question "how do we handle events?", GSIM is able to give an answer at the conceptual level with process design objects. It would be interesting to describe each of our main steps using the method presented in the GSIM/GSBPM linkage document. This will help disseminate the GSIM terminology at least to the IT team, and is anyway important to do to clarify our understanding of our process industrialization. In GSIM, the rule, parameter input, parameter output, process method objects are appropriate for defining and specifying what to do based on the type of event that occurs.

A question is how deep should we go with this type of description, knowing that process steps are recursive? It is possible that we describe only the highest steps corresponding to the examples above in the flow chart. The even finer steps should be specified only by the technical field. So GSIM could be used to pinpoint the main objects to give the main milestones in the process.

# 4. Building Protools: investigations from the IT team

From a technical point of view, building Protools also requires various investigations to answer the following questions:

- which technical architecture?
- which software solution?
- what technical modelling to trigger the events?
- what modelling for the HMI?

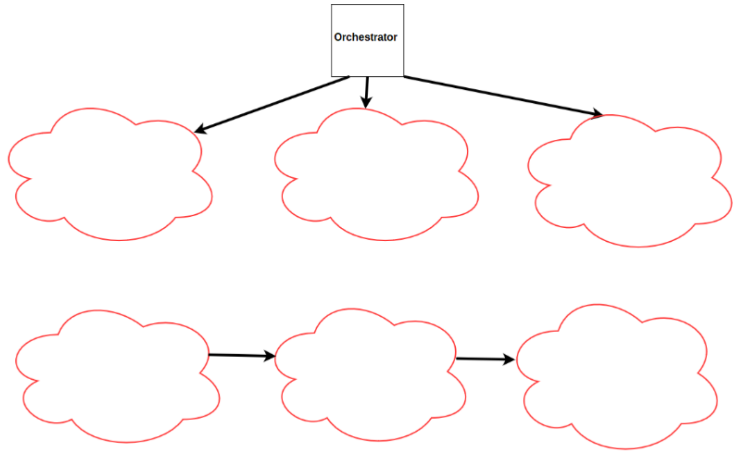
## 4.1. Choosing a technical architecture and software solution

The need is to implement a solution for orchestrating survey protocols based on active metadata. The tool will allow statistical teams to take control of the implementation and monitoring of the protocol. To do this, Protools will communicate directly with the sub-systems involved in the implementation of the protocol (sampling source, collection platform, post-collection data processing, etc.)

*orchestration vs choregraphie *(coordination)*

In the implementation of microservice architectures, it must be determined whether communication with services is centralised (orchestrated saga) or whether systems can interact directly with each other (choreographed saga). The control by metadata and the desire to set up a centralised monitoring of the progress of the process and the relatively modest size of the system led us to choose an orchestrated saga solution.

**Figure 6 Orchestrated saga vs choregraphed saga**



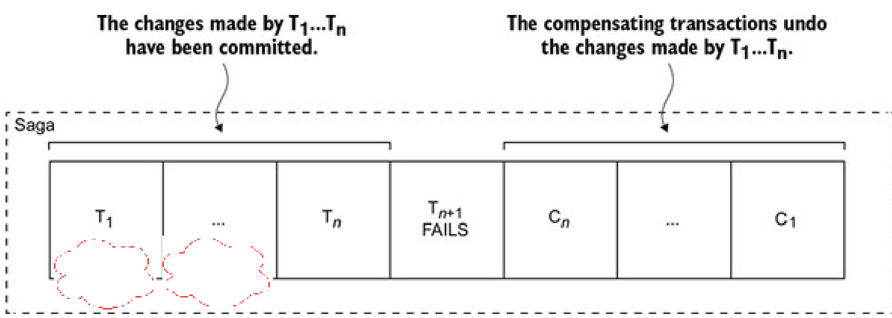

Choreographic approach : one participating service sends a message to the next after executing its local transaction.

Orchestration approach : a coordinating service invokes one participant after another.

*System status management (saga vs eventual consistency) (transaction)*

The complete transactional aspect is not required at all times so it is not appropriate to invest on a pure saga solution, which is rather complex to implement. A solution based on "eventual consistency" is sufficient, consistency being achieved in the end.

**Figure 7 Eventual consistency**




*async vs sync (communication)*

Communication with third party services can be done synchronously (the orchestrator waits for a response before proceeding) or asynchronously (the orchestrator does not specifically wait for the end of processing before proceeding but will come and take the information later). The modelling of the process will have to take these differences into account in order to specify the process in each case.

### 4.2. Choosing a technical modelling to trigger the events

The desire to drive the process directly with a formal modelling of the survey protocol, or a derivative of this modelling, leads to the study of the capacity of a processing engine to be driven by a process description model. Engines based on the BPMN2.0 standard are currently being studied (Camunda, Bonita, Activiti, Flowable).

In addition to this workflow orchestration approach, which will undoubtedly concern the level of the data collection, a fine-tuned management of communications with third party systems, driven by a set of events, will be handled by an order management system. This "command store" will be able to know the state of the process, to carry out restarts in the event of incidents and to avoid falsely replaying redundant processing.

In the end, a mixed solution was chosen: workflow orchestration (BPMN2.0) with a command store for communication with third-party services.

Note that we have ruled out event sourcing type systems, which are too complex to set up. Event Sourcing is an architecture pattern that proposes to focus on the sequence of state changes of an application that brought it to the state it is in. The idea is no longer to know where we are, but to keep track of the path taken to get there. If this type of architecture is important in systems where traceability and system coherence is essential, here, the pattern command store backed by the workflow engine is sufficient to ensure consistency at the end in all cases.

### 4.3. Choosing a modelling for the HMI Protools

Although the organisational aspects of the work around this orchestrator have not yet been studied, a set of services can be attached to this processing engine to accompany its implementation: a graphical interface for modelling processes closely linked with Insee's statistical metadata repository RMéS and an interface for monitoring and triggering processes.

## Conclusion

This joint work between IT Team and Metadata Team to identify the metadata necessary to drive statistical processes first required to review our use of the two GSIM objects and to detach them from our practical use, which was to back them up with the production of quality reports on an annual basis. By forcing the issue, the statistical program cycles were mainly interpreted as annual surveys, and this is a hindrance to managing complex protocols.

The reflections carried out in this work around the statistical program cycle and its correspondence with the framework of the consumer survey, will have to be continued in order to have an operational conceptual description of our processes, whatever their complexity. This work on the consideration of panels in GSIM is, we hope, a basis for discussion and continued reflection, whether internally or within the standards community.

Beyond the difficulties mentioned above, GSIM is able to provide a conceptual framework for piloting processes. The exercise will require a full description, but the objects mentioned so far in this document will be good reference points for managing objects in a dynamic or statistical way. Again, the document linking GSBPM and GSIM is a valuable tool.

Then it will be a matter of moving on to implementation. This one will be realized in practice, on the IT side, with the BPMN standard. Objects of this standard will be tagged with the GSIM objects and their counterpart in DDI or with the objects of the core ontology. A first version of Protools will be released during 2023 to switch the Consumer survey to the new collection information system.

## References

[1] DDI – Document, Discover and Interoperate:
https://ddialliance.org

[2] VTL – Validation and transformation Language:
https://sdmx.org/?page_id=5096

[3] A fully metadata-driven platform for the conception of survey questionnaires and the management of multimode data collection, september 2021:
https://unece.org/sites/default/files/2021-09/DC2021_S1_France_Werquin%20Cotton_AD.pdf

[4] Pogues, A questionnaire design Tool :
https://www.insee.fr/en/information/5014167?sommaire=5014796

[5] Eno, A collection instrument Generator :
https://www.insee.fr/en/information/5014703?sommaire=5014796

[6] Internet Business Data Collection: INSEE Enters the Coltrane Era :
https://www.insee.fr/en/information/5014714?sommaire=5014796

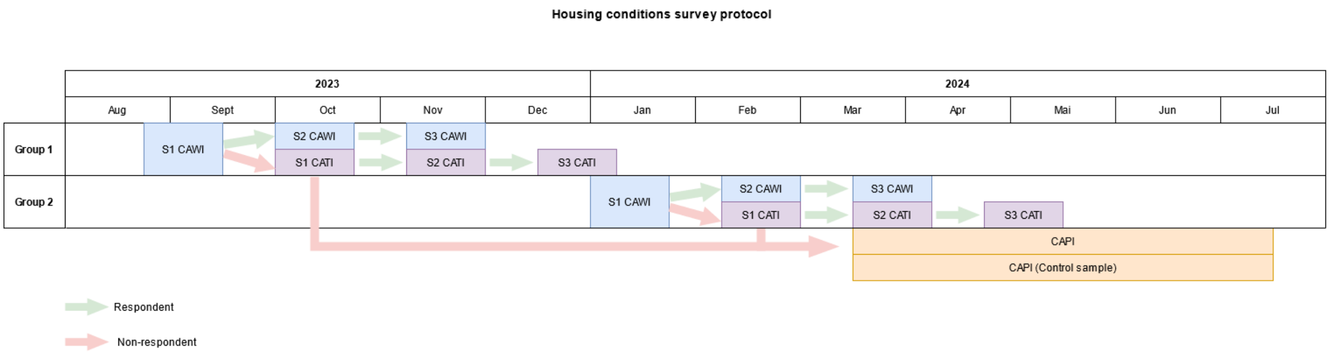
[7] RMéS: INSEE's Statistical Metadata Repository
https://www.insee.fr/en/information/4195079?sommaire=4195125

[8] Linking GSBPM and GSIM, January 2022:
https://statswiki.unece.org/display/GSBPM/Information+flow+within+GSBPM+using+GSIM?preview=/330370507/338330085/Linking

# Annexe 1

## Housing conditions survey

*The protocol diagram*



*The 27 operational partitions of the sample*

| Sample operational partition | Description | Collection mode | Questionnaire model | Group |
|---|---|---|---|---|
| 1 | CAWI1 | CAWI | Sequence 1 | 1 |
| 2 | CAWI1 | CAWI | Sequence 1 | 2 |
| 3 | CAWI1 --> CATI1 (non-respondents) | CATI | Sequence 1 | 1 |
| 4 | CAWI1 --> CATI1 (bad quality) | CATI | Sequence 1 | 1 |
| 5 | CAWI1 --> CATI1 (non-respondents) | CATI | Sequence 1 | 2 |
| 6 | CAWI1 --> CATI1 (bad quality) | CATI | Sequence 1 | 2 |
| 7 | CAWI2 | CAWI | Sequence 2 | 1 |
| 8 | CAWI2 | CAWI | Sequence 2 | 2 |
| 9 | CATI2 | CATI | Sequence 2 | 1 |
| 10 | CAWI2 --> CATI2 (non-respondents) | CATI | Sequence 2 | 1 |
| 11 | CAWI2 --> CATI2 (bad quality) | CATI | Sequence 2 | 1 |
| 12 | CATI2 | CATI | Sequence 2 | 2 |
| 13 | CAWI2 --> CATI2 (non-respondents) | CATI | Sequence 2 | 2 |
| 14 | CAWI2 --> CATI2 (bad quality) | CATI | Sequence 2 | 2 |
| 15 | CAWI3 | CAWI | Sequence 3 | 1 |
| 16 | CAWI3 | CAWI | Sequence 3 | 2 |
| 17 | CATI3 | CATI | Sequence 3 | 1 |
| 18 | CAWI3 --> CATI3 (non-respondents) | CATI | Sequence 3 | 1 |
| 19 | CAWI3 --> CATI3 (bad quality) | CATI | Sequence 3 | 1 |
| 20 | CATI3 | CATI | Sequence 3 | 2 |
| 21 | CAWI3 --> CATI3 (non-respondents) | CATI | Sequence 3 | 2 |
| 22 | CAWI3 --> CATI3 (bad quality) | CATI | Sequence 3 | 2 |
| 23 | CATI1 (non-respondents) --> CAPI123 | CAPI | Sequence 1+2+3 | 1 |
| 24 | CATI1 (non-respondents) --> CAPI123 | CAPI | Sequence 1+2+3 | 2 |
| 25 | CATI1 (bad quality) --> CAPI123 | CAPI | Sequence 1+2+3 | 1 |
| 26 | CATI1 (bad quality) --> CAPI123 | CAPI | Sequence 1+2+3 | 2 |
| 27 | CAPI123 control sample | CAPI | Sequence 1+2+3 | - |

*The 13 monitoring phases*

| Monitoring phase | Sample operational partition | Collection mode | Questionnaire model |
|---|---|---|---|
| 1 | 1 | CAWI | Sequence 1 |
| 2 | 2 | CAWI | Sequence 1 |
| 3 | 3 + 4 | CATI | Sequence 1 |
| 4 | 5 + 6 | CATI | Sequence 1 |
| 5 | 7 | CAWI | Sequence 2 |
| 6 | 8 | CAWI | Sequence 2 |
| 7 | 9 + 10 + 11 | CATI | Sequence 2 |
| 8 | 12 + 13 + 14 | CATI | Sequence 2 |
| 9 | 15 | CAWI | Sequence 3 |
| 10 | 16 | CAWI | Sequence 3 |
| 11 | 17 + 18 + 19 | CATI | Sequence 3 |
| 12 | 20 + 21 + 22 | CATI | Sequence 3 |
| 13 | 23 + 24 + 25 + 26 + 27 | CAPI | Sequence 1+2+3 |