ESSnet Smart Surveys

Grant Agreement Number: 899365 - 2019-DE-SmartStat

https://webgate.ec.europa.eu/fpfis/wikis/display/EstatBigData/ESSnet+Smart+Surveys+2020-2021

Workpackage 3

Development of a conceptual framework, reference architecture and technical specifications for the European platform for Trusted Smart Surveys

Deliverable 3.1 Report on the Preliminary Framework

Final version, 28-02-2021

Prepared by:

Massimo De Cubellis , Fabrizio De Fausti, Claudia De Vitiis, Alessio Guandalini, Francesca Inglese, Michele Karlovic Riccio, Fabiana Rocci, Roberta Varriale, Mauro Bruno, Giuseppina Ruocco, Raffaella Maria Aracri (ISTAT, Italy)

Nils Meise (DESTATIS, Germany)

Jacek Maślankowski (GUS, Poland)

Mirela Causevic, Joeri van Etten, Matjaz Jug, Rob Warmerdam (CBS, The Netherlands)

Franck Cotton (INSEE, France)

Work package Leader:
Claudia De Vitiis (ISTAT, Italy)
devitiis@istat.it; +390647737401

Table of contents

Introduction:

WP3 overview and general features of the platform

Introduction

The overall goal of the ESSNet Trusted Smart Surveys (TSS_u) is to define the specifications for the European platform supporting the use of shared smart survey solutions. Another aim of the project is to assess the usage of applications for European social surveys, such as the Time Use Survey (TUS) or the Household Budget Survey (HBS) (this aim is object of the Work Package 2).

The development of smart statistics and TSS_u offers new possibilities to improve the quality of social surveys. In smart statistics, respondents use smart devices (e.g. smartphones, tablets, activity trackers) to provide survey data. TSS_u may include data collection processes in which respondents are asked to share existing data collected by trusted third parties, like government authorities and larger organizations, stable enterprises willing to establish data delivery agreements. Abstracting from the domain, TSS_u may also combine smart and traditional data sources. Regardless the type of smart data considered, these new scenarios entail the revision of the statistical process to guarantee accuracy and reliability, according to the principle of accountability and transparency of Official Statistics.

In this context, the Work Package 3 (WP3) is carrying out preparatory work to create a European-wide platform, to share and re-use smart survey solutions and components on the TSS_u platform.

The mail goal of the Work Package 3 is the development of a conceptual framework, reference architecture and technical specifications for a European platform for trusted smart surveys, including support for secure private computing to avoid data concentration (e.g., secure multiparty computation), full transparency and public auditability. It should also include a configurable set of individualised incentive schemes. The work should include any methodological issues related e.g. to quality framework and quality standards, developments related to other standards such as GSBPM and GSIM, metadata, business enterprise architecture, IT infrastructure, etc.

The first phase of the work consists in the conceptualisation of a general platform for trusted smart surveys for collecting data for official statistics, following a top-down design approach from abstract framework through architecture down to detailed technical specifications. The following aspects are addressed among others:

- ontologies of concepts enabling knowledge sharing and development of semantics;
- measurement methods;
- smart survey methodology;
- technical infrastructure;
- privacy preservation, confidentiality protection, data and process governance;
- individualised incentives and effective communication;

- definition of required metadata throughout the process.

The second phase consists in the development of Proofs-of-Concept, in the form of modular prototype elements for essential aspects of the architecture such as:

- active and passive data collection;
- the use of machine learning for the identification of activities, identification of parallel activities, missing data, etc.;
- privacy-preserving computation solutions;
- full transparency and auditability of processing algorithms;
- integrating of incentive schemes into the platform;
- front-ends for configuration (allowing survey managers to instantiate and run new surveys with full support for multilingual needs).

The activities to pursue these objectives are organized in two main tasks:

3.1 Framework Conceptualisation and development of a general platform for trusted smart surveys

3.2 Development of proofs-of-concept

The activities of the Work Package 3 carried out in year 2020 aimed at the definition of the preliminary framework for the TSSu platform. During 2020, the activities followed the structure of the first task and the division into sub-tasks, each coordinated by a responsible partner, as shown in the following table.

3.1.1	Smart survey methodology	Destatis Istat Insee
3.1.2	Technical infrastructure	Gus Cbs Destatis Istat
3.1.3	Integration in existing architectural frameworks	Istat Insee
3.1.4	Preservation of privacy and transparency	Cbs Istat
3.1.5	Incentive schemes	Destatis Istat
3.1.6	Metadata - Process Auditability	Insee Cbs Istat

The main goal of the work carried out in 2020 was the present deliverable D.3.1 “Report on the Preliminary Framework”.

The deliverable is structured according to the sub-tasks, one chapter for each sub-task coordinated by the sub-task leader. Furthermore, a parallel activity carried out was the definition of the planning for the Proof-of-Concept, to be implemented in the second part of ESSNet in 2021.

TSS_u platform objectives

The following paragraphs provide an overview of the expected outcomes, and the guiding principles stated in the project description for designing the platform.

General requirements

The overall objective of WP3 is to provide a conceptual framework, reference architecture and technical specifications for a European platform for TSS_u, including support for secure private computing to avoid data concentration (e.g. secure multiparty computation) and guarantee full transparency and public auditability. It should also include a configurable set of individualised incentive schemes. The work should include several

methodological issues, e.g. innovative methods to process smart data, quality framework, alignment with official statistics standards.

As stated in the grant agreement, the platform should fulfil the following requirements:

1) Flexible European platform for trusted smart surveys: the platform should be implemented as a set of common (horizontal) functions and configurable services that can be used to build particular instances of TSS_u for specific application domains and/or target areas.
2) Development at the European level: the platform should be used independently by NSIs to perform national surveys but could also serve as basis to launch European transnational surveys.
3) Modular, evolvable, extensible and agnostic to particular application domains. It should provide ready-to-use solutions for horizontal functions. It should allow each platform user (i.e., any ESS member) to instantiate a specific trusted smart survey by selecting and configuring different modules.
4) Support for secure private computing to avoid data concentration (e.g., secure multiparty computation).
5) Address methodological issues. New methodological challenges rise in the context of TSS_u, e.g. analysis of new data sources, analysis of sensor data, secure private computation. The platform should offer statistical services to address such issues.

More specifically, the reference architecture and the specifications of the envisioned platform should:

- Provide a set of configurable functions and primitives that can be used to compose a wide range of specific interaction models (with the human respondent and with the smart device(s)) to meet the specific needs of various surveys, in different application domains and countries;
- Allow the reuse of existing smart survey tools and applications;
- Include appropriate methods for handling missing data, identifying and correcting dubious input data, possibly exploiting the interaction loop with the respondent and his/her smart device(s);
- Include intelligent algorithms (e.g., based on AI) to govern the interaction with the human respondent in order to minimise response burden and prevent interference with his/her activities;
- Include intelligent algorithms (e.g., based on AI) to govern the interaction with the respondent's smart device(s) in order to keep the additional resource consumption (battery power, communication bandwidth, etc.) down to minimum levels;
- Adopt a user-friendly design for end users with intuitive and easy-to-use interfaces;
- Provide strong guarantees in terms of security, privacy preservation, confidentiality protection and auditability, through a composition of appropriate technological solutions, including secure private computation (e.g., based on Secure Multiparty Computation solutions) and non-modifiable transparent logging of data transactions (queries);
- Provide flexible support for implementing individualized incentive schemes, as needed to promote participation by citizens.

Design principles

Smart statistics play an important role in the future of official statistics, in a world overwhelmed by smart technologies. Smart technologies involve real-time, automated, interactive systems that optimize the physical operation of appliances and consumer devices. Statistics themselves would then be transformed into a smart technology, embedded in smart systems that would transform "data" to "information". Trusted Smart Statistics can be seen as a service provided by smart systems, embedding auditable and transparent

data life cycles, ensuring the validity and accuracy of the outputs, respecting data subjects' privacy and protecting confidentiality.

In this context, the following design principles[1] should guide the design and development activities of TSS_u:

- Aim for output information, not input data
- Clear separation between development and production
- **Pushing computation out instead of pulling data in**
- Sharing control in development
- Sharing control in production
- **Leverage privacy enhancing technologies**
- **Stepping up transparency and accountability**
- Engaging external stakeholders
- **New methodological frameworks**
- **The smart and trusted cycle**

The list of principles is quite long and some of them are closely interconnected. Therefore, the approach adopted is to highlight the principles that are more tied to the activities in this stage.

Outline of the deliverable

The deliverable represents the product of the first year of activity of Work Package 3 on the conceptual framework for Trusted Smart Surveys. The framework is in a preliminary stage, all the sub-activities have started from a very high level following a top-down approach. At the end of the second year and considering the results of the Proof-of-Concept, the framework will take a step towards the definition of the platform specifications and requirements.

The deliverable is divided into six chapters, corresponding to the six sub-tasks 3.1.1-3.1.6 and include an Annex outlining the Proof-of-Concept.

Task 3.1.1 Smart Survey Methodology

The activity of task 3.1.1 aims at developing a robust smart survey methodology. It explores design requirements for (Trusted) Smart Surveys (TSS_u) in contrast to traditional paper-based or online surveys. The main problems addressed in the report on the preliminary framework are related to: sensor data from a variety of devices that are not standardized in structure, format or availability; innovative ways of handling sensor data (machine learning algorithms); sources of error in TSS_u and error management; GSBPM phases involved in TSS_u; automation of smart surveys and needed staff profiles.

The chapter 1 consists of three sections: section 1 - Smart data features; section 2 - Errors in smart surveys; section 3 - Smart survey processes.

In the first section, the focus is on the main methodological aspects and features of TSS_u, designs and new data collection methods. Furthermore, an overview is presented about: mobile devices and wearables to be

[1] The source of this paragraph is the paper: Ricciato F., Wirthmann A., Giannakouris K., Reis F., Skaliotis M.: Trusted smart statistics: Motivations and principles. Statistical Journal of the IAOS (November 2019).

used in smart surveys providing detailed information on the sensors and data they provide; machine learning (ML) algorithms that might be useful when dealing with new unfamiliar types of data (sensor data) and issues related to the quality of data, such as training models and metric to evaluate ML algorithms.

In second section, the sources of error in TSS_u are analysed considering the technologies and instruments used to collect data and the new type of data acquired (sensor data acquired actively and/or passively through apps). The purpose of this analysis is to highlight all the elements necessary for a redefinition of the Total Survey Error (TSE) framework for smart surveys which takes into account the new sources of error, both in representation (selection) and measurement (traditional and sensor data). For analysing data accuracy in smart surveys, the TSE for traditional surveys and the Total Error (TE) defined for big data are considered as reference frameworks: the last one to understand the sources and nature of new types of error in sensor data acquisition and management.

In the third section a preliminary methodological framework for TSS_u is outlined taking GSBPM as a reference statistical model to facilitate connections with the architectural system, in particular with data collection and data processing components of the platform. The focus is on phases and sub-processes that need changes for the use of smart technologies and smart data, innovations that require a new data collection channel, new processing tools, and new skills for managing sensor data. A GSBPM design phase for TSS_u is introduced, the take on data collection for trusted smart surveys is linked to the GSBPM collect phase and indicators for setup and monitoring are discussed. Moreover, the challenges of processing smart data are expanded by the use of machine learning drawing the link to the GSBPM process phase. Finally, automation of smart data processing and infrastructure and staff profile needs are discussed.

Task 3.1.2 - Technical Infrastructure

The chapter 2. Technical Infrastructure, shows an architecture of two different smart surveys platforms, i.e., HBS (Household Budged Survey) and MOTUS (Modular Online Time Use Survey). Its technical infrastructure analysis was also used to create a generic infrastructure for smart surveys. In this chapter there is also a list of requirements that should be fulfilled by the smart survey application. It is important to note that these two smart surveys applications are still under development.

The aim of this chapter is on the analysis of the as-is architecture of HBS and MOTUS platform, as of 2020. The current work done is a preliminary step to develop a to-be architecture in the next phase of smart surveys maturity. The next step would be to develop a Proof-of-Concept for the smart surveys.

This chapter is divided into three parts. In the first part a general information on possible ways of data collection through the mobile devices was presented. The second part shows the requirement analysis for smart surveys, i.e., mobile applications used to collect statistical data. The third part shows the logical components of technical architecture according to the BREAL - Big Data REference Architecture and Layers Application and Information Architecture.

Task 3.1.3 – Integration in Existing Architectural Frameworks

The main goal of task 3.1.3 is to provide an overview of reference standards and initiatives preceding TSS_u exploration. This analysis will foster the alignment of WP3 activities with existing architectural frameworks and will allow to benefit from the achievements of related experiences. The alignment with existing frameworks is also compliant with TSS_u driving principles addressing the privacy issue, the new methods for smart data processing, as well as process transparency, accountability and smart data life cycle.

Starting from an overview of official statistical standards and legal framework, the chapter focuses on the main results of the ESSNet Big data II, especially on the Big Data REference Architecture and Layers (BREAL). A brief description of relevant data platform projects complements the introductory analysis.

Finally, the last section analyses the key elements to consider for modelling the business layer of the TSS_u platform. This platform will implement a set of common (horizontal) functions and configurable services, to build instances of TSS_u for specific domains and/or target areas.

Task 3.1.4 – Preservation of privacy and transparency

When novel data collection strategies are proposed and data from new sources is collected, concerns with regards to security and privacy are inevitable, and rightly so. NSI' should take great care for the data they collect so as to not let the privacy of respondents be compromised. However, at the same time, NSI's are obligated to serve the public by producing accurate statistics. NSI's should hence look for ways to enable leveraging new data-sources, while preventing privacy being breached.

The aim of task 3.1.4 is to assist in this process and the chapter provides the reader with a set of tools that might help with understanding and dealing with the privacy considerations involved in the aforementioned new forms of data collection.

Specifically, in the first section, an overview is given of the Privacy-by-design framework for handling private information. This framework is subsequently converted into a set of guidelines which should assist in process of making correct decisions with regards to how private data should be dealt with by an application which collects such data.

The second section is dedicated to techniques for preserving privacy while maximizing data utility. An overview of a number of such techniques (e.g. differential privacy, homomorphic encryption, secure multi-party computation, etc.) is given. The benefits and drawbacks are highlighted, as well as typical use-cases. Finally, a discourse on what these techniques could mean TSS applications is presented. Also, a number of scenario's in which these techniques could be applied in the smart survey process are described.

In the final section, the issue of user authentication is tackled. Naturally, the identity of TSS participants should be verified. However, this should be done using a minimal amount of information to reduce privacy impact. A protocol for user authentication using a minimal set of characteristics is therefore presented in the form of IRMA, short for 'I Reveal My Attributes'. Finally, an example implementation of IRMA in the context of a TSS application is given.

Task 3.1.5 – Incentives Schemes

The activity of Task 3.1.5 aims at exploring incentives for Trusted Smart Surveys. Manual data entries can be cumbersome for respondents and potentially lead to missing entries and drop outs. This activity investigates how gamification could be used as a strong incentive scheme to engage respondents more in the data collection process.

The chapter consists of the following parts: first, an overview of common incentives schemes and an introduction into gamification. Second, how gamification can be applied to surveys and how to tailor it to specific groups of respondents. And third, a discussion of implementation and risks of gamification. The summary concludes with the next steps in line of this activity.

Task 3.1.6 - Metadata

The main objective of task 3.1.6 is the design of the TSS_U platform metadata component, which is a challenging task that requires an in-depth analysis of several aspects, namely:

- statistical concepts related to the survey theme and objectives;
- collection instruments, variables and codes;
- statistical methodology used in data processing;

- process and lineage metadata;
- quality metadata.

Starting form a preliminary analysis of standards and frameworks (i.e., ontologies, BREAL Information Architecture, GSBPM and GSIM) related to metadata concepts, the chapter focuses on the following general aspects:

- Data structures
- Sensor data
- Process steps
- Privacy issues
- Generic functionalities for reuse in different domains
- Specific characteristics of national contexts.

The approach adopted to fulfill these general objectives is based on the following principles:

- Rely on standards and align with existing frameworks;
- Use active/passive metadata;
- Use-case-driven approach. The first use case selected for testing the metadata model is the Time Use Survey, combining GPS data and other variables (e.g., “type of activity”) collected through an app on.

In order to overcome specific domain requirements, the main goal of this task is to model a repository with the minimum set of metadata for process standardization and application components reuse. The implementation stage should focus primarily on the description and integration of sensor and traditional data used in TSS_u.

Annex: Proof-of-Concept

A parallel activity carried out in 2020 was the definition of the planning for the Proof-of-Concept, to be conducted in the second part of the ESSNet, in 2021.

The objective is the development of proofs-of-concept in the form of modular prototype elements for essential aspects of the architecture such as: active and passive data collection, use of machine learning, privacy-preserving computation solutions, full transparency and auditability of processing algorithms, integrating of incentive schemes into the platform.

The proposed proofs-of-concepts have been grouped into two sets, according to the perspective of the issues addressed: one group has a more methodological point of view and includes two proofs-of-concepts, while the second concerns more technical and architectural aspects and comprises three proofs-of-concepts.

Glossary

Smart Surveys / Trusted Smart Surveys:

By the term “smart surveys” we refer to surveys that use smart personal devices, equipped with sensors and mobile applications. The concept of smart surveys goes well beyond the mere use of web-based (online) data collection that essentially transform the paper questionnaire into an electronic version. Smart surveys involve dynamic and continuous interaction with the respondent and with her personal device(s). They combine data collection modes based on input from the data subjects (active data) with data collected passively by the device sensors (e.g. accelerometer, GPS, microphone, camera, etc.).

Trusted Smart Surveys (TSS_u) - surveys in which respondents are asked to share existing data collected by trusted third parties, like government authorities and larger, stable enterprises willing to establish data delivery agreements.

The term “trusted smart surveys” refers to an augmentation of smart surveys by technological solutions that collectively increase their degree of trustworthiness and hence acceptance by the citizens. Constituent elements of a trusted smart survey are the strong protection of personal data based on privacy-preserving computation solutions, full transparency and auditability of processing algorithms.

Platform for the Trusted Smart Surveys:

The European Platform for Smart Surveys will be flexible and implementing a set of common (horizontal) functions and configurable services that can be used to build particular instances of trusted smart surveys for specific application domains and/or target areas. Such a platform will be modular, evolvable, extensible and agnostic to particular application domains. It will provide ready-to-use solutions for horizontal functions. It will allow each platform user (i.e., any ESS member) to instantiate a specific trusted smart survey by selecting and configuring different modules. The platform will include support for secure private computing to avoid data concentration (e.g., secure multiparty computation), full transparency and public auditability.

Task 3.1.1

Smart Survey Methodology

Version: February 2020

Prepared by:

Massimo De Cubellis (ISTAT, Italy)

Fabrizio De Fausti (ISTAT, Italy)

Claudia De Vitiis (ISTAT, Italy)

Alessio Guandalini (ISTAT, Italy)

Francesca Inglese (ISTAT, Italy)

Nils Meise (Destatis, Germany)

Fabiana Rocci (ISTAT, Italy)

Roberta Varriale (ISTAT, Italy)

Task leader: Nils Meise

Outline

Executive Summary

The activity of task 3.1.1 aims at developing a robust smart survey methodology. It explores design requirements for (Trusted) Smart Surveys (TSS_u) in contrast to traditional paper-based or online surveys. The main problems addressed in the report on the preliminary framework are related to: sensor data from a variety of devices that are not standardized in structure, format or availability; innovative ways of handling sensor data (machine learning algorithms); sources of error in TSS_u and error management; GSBPM phases involved in TSS_u; automation of smart surveys and needed staff profiles.

The report is organised in three sections.

In section 1, we focus on main methodological aspects and TSS_u features, opportunities and challenges (chapter 1) and on designs and new forms of data collection (chapter 2). Furthermore, we present a deepening with an overview on: mobile devices and wearables to be used in smart surveys providing detailed information on the sensors and data they provide (chapter 3); machine learning (ML) algorithms that might be useful when dealing with new unfamiliar types of data (sensor data) and issues related to the quality of data, such as training models and metric to evaluate ML algorithms (chapter 4).

In section 2, we analyse the sources of error in TSS_u (chapter 5) considering the technologies and instruments used to collect data and the new type of data acquired (sensor data acquired actively and/or passively through apps). The purpose of this analysis is to highlight all the elements necessary for a redefinition of the Total Survey Error (TSE) framework for smart surveys which takes into account the new sources of error, both in representation (selection) and measurement (traditional and sensor data). For analysing data accuracy in smart surveys, the TSE for traditional surveys remains the reference framework, but we use the Total Error (TE) framework defined for big data to understand the sources and nature of new types of error in sensor data acquisition and management.

In section 3, we outline a preliminary methodological framework for TSS_u taking GSBPM as a reference statistical model to facilitate connections with the architectural system, in particular with data collection and data processing components of the platform. We focus on phases and sub-processes that need changes for the use of smart technologies and smart data, innovations that require a new data collection channel, new processing tools, and new skills for managing sensor data. We introduce a GSBPM design phase for TSS_u not taking into account a more general survey design with the use of “non-smart” modes. Then we link our take on data collection for trusted smart surveys to the GSBPM collect phase and discuss indicators for setup and monitoring. Moreover, we go deeper into the challenges of processing smart data by the use of machine learning drawing the link to the GSBPM process phase (chapter 6). Finally, we discuss automation of smart data processing and infrastructure and staff profile needs (chapters 7 and 8).

SECTION 1 - SMART DATA FEATURES

1. Trusted Smart Surveys

1.1 Smart Survey Methodology

A smart survey is not a survey that left its analogue boundaries and shifted to some digital realm – a web questionnaire or an app-based questionnaire is not smart by itself. A smart survey is based on our *digitalization of life*, which produces data as a byproduct of our actions and is fueld by a common access to smart devices. A smart survey takes all of this into account by utilizing available data and sensor-based data of smart devices in the context of a traditional survey. Thus, a smart survey methodology is offering a link between legacy data – which means mostly traditional survey data – and smart data. The change of scope is not only in phases of data collection, but also in data processing and data analysis. A smart survey methodology not only differs from established ways of data collection, but also marks a cultural shift in the production of official statistics. A smart survey methodology for official statistics builds on already established processes and frameworks, which allows it to integrate into the production process and introduce new insights. This shift is best coined by what Ricciato et al. call a "*major evolution, rather than revolution* of the current official statistics model […]" (Ricciato et al. 2019, p. 591, emphasis in the original).

Our focus is on a smart survey methodology that considers the capabilities of smart devices and especially their sensors *to assist* a respondent during a survey. However, the scope of smart surveys goes beyond simple assistance. Not only data itself or the way it is collected changes, but also how the whole production process is influenced by this change. At the core smart surveys *augment* already established ways of data collection and force the statistical system to adapt new means of handling data, needed infrastructure and staff profile needs. The is also driven by the advent of *big data* in the domain of official statistics, which was recently addressed by two ESSNets on big data in official statistics. In terms of big data there is no consensus what it actually means. However, most definitions name volume, variety and velocity as crucial criteria. Which points out that high amounts of data, in a variety of forms and formats are constantly created (Salganik 2018, p. 14). This constant creation of data is not only a defining feature of our world of today and all around us, but also a vast resource of information for official statistics. Although, data itself or the way it is collected needs to be *repurposed* for the use in official statistics to fit its needs. In the past statisticians in official statistics mostly relied on data they created for their needs – e.g. via surveys – or on data collected by other government offices. Which means that most of the data was collected for the purpose of official statistics. Smart surveys are a fusion of traditional surveys that ask directly the information needed for official statistics and collection of repurposed sensor data of devices respondents are connected to or interact in various ways. Thus, smart surveys make use of *datafication*, which is the transformation of many aspects of our life into information (c.f. Mayer-Schönberger and Cukier 2013; Ricciato et al. 2019). Digitalization of life, Datafication and big data – as an umbrella term – describe the same phenomenon with three slightly different perspectives on how information technology changed aspects of life and the creation of data.

Dealing with big data in official statistics made huge leaps forward in the last couple of years (Vaccari 2016), which has not only considered new data sources to extract (Vichi and Hand 2019), but also the use of sensors in mobile devices for the use in official statistics has been investigated (Mussmann and Schouten 2019). In addition, a new reference architecture for big data evolved in the ESS (Scannapieco et al. 2019) that already shaped the activity in this task. However, almost all new data sources are external to statistical offices and are brought or bought in from third parties. These data sources were not created to be used in official statistics, which can be problematic during data cleaning or data analysis. Consequently, administrating

trusted smart surveys is also meant to establish a controlled and methodological sound way to integrate new data sources in the production of official statistics.

Acceptance of the use of mobile devices and passive data collection is a key element for smart surveys. Hence, the notion of trusted smart surveys, which value the privacy of respondents. Willingness to participate in surveys with passive data collection varies considerably. Wenz et al. (2019) did a study in the UK to determine the acceptance of passive data collection and came to the conclusion that the willingness to participate depends on the type of action involved. Downloading and installing an app, passive data collection in general, high concerns of data privacy and less intense use of mobile devices made respondents less likely to participate. In another study by Keusch et al. (2019) the willingness to participate in passive mobile data collection was investigated by using vignettes on respondents of a German online panel. Respondents of the panel, who owned smartphones, were invited to the study. The results suggest that the willingness to participate is strongly influenced by three categories. First, promised incentives. Second the characteristics of the study, which includes the survey organization, duration of collection and opt-out options for certain data collection in the app. And third, the respondent characteristics, which includes privacy and security concerns as well as experience with smartphones. For a smart survey methodology, these are useful hints that point at potential error categories like representation errors. In addition, the always-on assumption concerning mobile devices is flawed, if the use of a device changes during a survey period or respondents opt-out of passive data collection. The activity of this task addresses potential survey errors that might result from the observed behaviour. Questions of how to deal with privacy concerns or how to include incentives are addressed in the activities of task 3.1.2 and task 3.1.5 respectively.

The smart survey methodology that we outline is agnostic to the statistical domain, because we mainly aim at a high-level approach in our delivery. We have a strong focus on sensors on the data collection side and on machine learning on the data analysis side. In particular, we address sensor data from mobile devices, wearables and other Internet of things (IoT) devices. We think of the data as part of a dedicated data collection process with a trusted smart survey, though it is possible to transfer our results to any kind of sensor data analysis and its application in official statistics. This highlights that what we suggest here are re-useable and adaptable products that are tailored for use in a variety of smart surveys and for any third party sensor data.

1.2 Main Features of TSS_U

The main features of smart surveys go beyond simply using sensors of smart devices. They are a special kind of survey, privacy-intrusive, generate hybrid forms of data, depending on software, and need other planning than legacy surveys.

A smart survey is not smart, because it uses computer assistance or devices that are called smart in our everyday life or sensors as some new “technology” to help statisticians to do their job. Smart surveys are smart, because they demand a change in how surveys are designed and executed. First, the design phase is either more experimental or relies on already developed methods or analytical algorithms. The design is different, because smart surveys also use proxies for information: a sensor reading and its classification is not the same as an answer to a direct question or a cluster of questions. It is also expected to be more complex, time and cost-intensive, due to utilizing data sources that were not intended to serve official statistics and could quickly change over time (c.f. Ricciato et al. 2019, p. 593). Survey designers are used to design a custom-made survey, where every piece is carefully selected by them, so that the final product answers the questions they have. The true potential lies in understanding the different approaches of carefully created surveys and readily available data and their respective strength and weaknesses in order to

select or combine custom-made data and readily available data (Salganik 2018, pp. 7–8). The latter approach is what trusted smart surveys intend to achieve. Second, the execution or rather production phase including data transfers, data wrangling, and analysis is independent of human interaction. In principle fully automated, objectively described by code and thus separating development from production as well (c.f. Ricciato et al. 2019, pp. 593–594). Due to the focus on sensor data, another factor comes into play: available devices. They are in-between the design and production phase. If respondents use their own devices, they might lack some functionalities due to age or replaced some legacy technology already. A design phase has to consider this variety of devices and sensors. Or respondents get a set of devices, which causes much more troubles in the production phase, because they might be unfamiliar with them, don't want to use them all the time or the devices break and it is the responsibility of the National Statistical Institutes (NSI_s) to offer a help desk or service. Already from this short list of troubles, it gets apparent that respondents should use their own devices for the heavy lifting of the survey and if at all only supplement devices with more or different sensors (e.g. wearables) should be distributed on low scale surveys. A third option is to cooperate with companies that sell and manage smart devices. In this case roles and responsibilities have to be clearly defined and interfaces for data transfer established. In short, a smart survey needs more time in design, but less time during the data collection phase. An unavoidable risk is that the design needs adaption when smart devices change or lack functionalities. For most use cases of official statistics, it seems appropriate to relay on respondents' smart device infrastructure and not create a burden for the NSI_s during data collection.

Smart surveys are by design privacy intrusive, because they passively collect data at all times (always on). Thus, respondents must be more than ever informed about the data collection process and willing to share their information (informed consent). A respondent can easily give false information in a survey, but cheating multiple sensors that enhance and validate the data collection process is a harder task for most people. A *trusted* Smart survey methodology must consider this and proactively include measures that do not create a burden for the respondents out of their intrusive nature. Also data collection with smart devices and survey software means involving third parties in the data collection process, which needs to be addressed with clear policies and boundaries for all involved parties (e.g. hardening apps to not allow the operating system or other apps access its data).

Prior methodological research on the use of sensors of mobile devices in official statistics has shown that hybrid forms of data – survey data linked to other data sources – are the next step for including sensors and thus also smart devices into surveys (Mussmann and Schouten 2019, p. 2). This also indicates that smart devices and sensors are still mere augmentations and not replacements for surveys. However, sensors are more than a proxy of activity. In contrast to legacy surveys, smart surveys can offer (objective) data to augment or replace (subjective) answers in a questionnaire. Thus, reducing human error but also shifting potential human-made bias to a technology-based one. Or rather mark a human-machine interaction bias – if no smart device is around or used, there is no smart data. In official statistics, we are used to highly structured data (e.g. most survey or administrative data is highly structured) and have a vast domain knowledge to validate and thus clean our data for analysis or the final statistical product. Smart surveys introduce a new "dirty." Thus, we cannot rely on our experience to make sense of our data. And due to the nature of different sensors, systems and fast changes in technology, there is a high variety in our data and thus in the problems it may cause.

It should be obvious by now that (trusted) smart surveys depend on software running on a smart device or on a connected device that also communicates with the underlying devices and its sensors. The higher the dependency on sensor input is, it is less likely to conduct a survey in multi-mode. Though smart devices are essential to collect sensor data, survey questions could be asked through other modes as well. Also,

communication with respondents is still possible on multiple channels and not limited to a smart device or software. A crucial question is, what is the methodological rationale to use sensor data in a particular survey? Is sensor data used to *augment* a survey or is it used to *replace* survey questions or other forms of active data collection? Because sensor data cannot be replaced by switching to a different mode.

1.3 Opportunities and Challenges

The development of smart surveys based on mobile devices (e.g. smartphones, tablets, wearables) offers new opportunities for social surveys to collect data and hence, to generate new and unfamiliar types of data.

Mobile devices can be used in various ways for data collection. With smartphones and tablets, questionnaires can be administered in innovative ways, such as asking respondents to answer questions via text messages, or completing questionnaires in a mobile web browser or survey app. Other tasks of mobile data collection include asking respondents to take photos with the camera of their smartphone or tablet, for example to scan payslips or shopping receipts, or to track how respondents are using their mobile device, for example which websites they are visiting.

Mobile devices have additional measurement capabilities represented by sensors: Global Positioning System (GPS) together with similar sensor data (GLONASS, GALILEO) can be collected from the respondent's mobile device to measure their location and travel patterns or to trigger surveys at pre-specified locations using geo-fencing; Accelerometer data can similarly be collected from the respondent's mobile device, as can data from external devices that are connected via Bluetooth, such as activity trackers, smart scales, or transdermal devices. Such data can be used to measure physical activity as well as other biological features, such as weight, body fat, and stress.

Smartphone sensors (e.g., GPS, camera, accelerometer), apps, and wearables (e.g., smartwatches, fitness bracelets) allow researchers to collect rich behavioural data, potentially with less measurement error and lower respondent burden than self-reports through surveys. However, there are new and multiple challenges to collecting these data: participant selectivity, (non-)willingness to provide sensor data or perform additional tasks, privacy concerns and ethical issues, quality and usefulness of the data, innovative ways of handling sensor data and practical issues of implementation.

2. Smart Survey Designs and Hybrid Forms of Data Collection

The most rudimentary form of smart survey is a computer-assisted survey on a smart device with online access, i.e. it actively uses the local computing power and storage beyond merely asking questions. Computing power and storage can be used for various purposes such as plausibility checks, edit rules application, computation of new variables of interest and gamification options. Smart surveys and trusted smart surveys could be designed for any device. However, devices vary greatly in sensor availability. Desktops and laptops have only a limited set of sensors, whereas smartphones and tablets have a rich set of sensors. For some of the features, it is natural to use mobile devices only. Smart surveys are, therefore, often associated with mobile devices, but can in more simple forms also be run on fixed devices (Schouten, 2020).

A genuine smart survey design is characterized by *augmenting* surveys with additional data sources, most commonly sensor data. It utilizes smart devices by accessing their sensor and communication modules to gather and transfer data for use in a survey on a connected device or the device itself. Thus, a smart device can be any mobile device – such as a smartphone or tablet – or any other device, which uses sensors to collect information about itself or its surroundings. We acknowledge that mobile devices are, on one hand, the de

facto widespread standard for smart devices with multi-purpose use, and on the other hand a gateway for other (handheld) sensors, which further enhances the options for available sensors or cross references. Hence, mobile devices are in the focus of this discussion that emphasizes their use beyond pure digitalization of legacy paper questionnaires.

In this chapter, we try to define different types of smart surveys that use mobile devices as collection mode. For this purpose, we present: three approaches for conducting online surveys via smartphones; the taxonomy proposed by Barry Schouten (Schouten, 2020); the collection of various types of data generated by the mobile device (subjective and objective measures); primary and secondary data collection.

2.1 Online Smartphones Surveys Approaches

We consider three main approaches for conducting online surveys via smartphones including online surveys taken via the mobile browser, app-administered surveys, and a hybrid approach combining the online mobile browser and app-based approaches. Following the work of Buskirk and Andres (2012) we report the approaches features, advantages and disadvantages (Table 1).

An example of a hybrid approach is represented by the Modular Online Time Use Survey - MOTUS (SOURCE^{TM} project - Minnen et al., 2020), a software tool that supports online time use surveys via a mobile (iOS and Android) and/web application (via browser; www.motusresearch.io). The peculiarity of MOTUS consists in considering as data collection instrument not only mobile devices (app and web app for smartphone and tablet) but also fixed devices (web app for computer and laptop). To participate via a browser an internet connection is needed. Combined online-offline registration is possible via the mobile application. Respondents can use any preferred device as the design for both applications is similar and the information collected by the devices is shared and synchronized between the devices. The web app is responsive to function on different screen sizes. Behavioural information can also be captured via sensors in the smart devices.

Table 1. Approaches for surveys via smartphones devices

Approaches	Features	Advantages	Disadvantages
Mobile browser surveys - Active version approach (A-MBS)	Mobile versions modified and adapted to account for differences in browser size	• Graphic images can be sized and displayed in proper proportion to mobile web browser size resulting in a reduction in the total amount of data exchanged and faster page loads • Next/Back buttons and other navigation tools can be explicitly visible on every page • Horizontal scrolling can be more easily limited • Number of questions appearing on a page can be explicitly controlled • Question layout is optimized (to the extent possible) for particular mobile browsers and can take advantage of native features such as scroll wheels • Streamlined survey completion via mobile browser	• Persistent internet connection needed to complete survey • A server must generate each subsequent survey page (in its entirety) and data exchange occurs with each new page • Longer surveys may be difficult to process due to the form factor and the internet lag • Some data capture regarding GPS and other features may require additional • permissions set by the user • Cost of development may be slightly higher in order to develop and deploy a mobile browser-specific version of the survey • Can't control the orientation of the screen used to complete the questionnaire • Scrolling may be unavoidable if images or grids are used • Requires beta testing on various platforms/mobile browsers • Different smartphone operating systems may interpret the mark-up languages and survey formatting slightly differently than intended and these variations are often beyond the control of the survey designer
Surveys administered via smartphone apps App Based smartphone Survey approach (ABSS)	This approach relies on a smartphone operating system specific app installed on the respondent's smartphone that pushes survey content to the end-user and uploads collected data whenever an	• Question layout is self-contained in the app without the need for scrolling (horizontally) or zooming • Can capture various types of data including GPS, Pictures and Video from within the App itself • Orientation of the survey questions (presentation and completion) can be controlled • Data collection can occur without the need for a persistent internet connection, which may reduce break-off related item nonresponse	• Requires development/deployment of potentially multiple app versions (i.e. one for each smartphone operating system) • Requires the respondent to download/install an app on their smartphone • Some data capture regarding GPS and other features may require additional permissions to be set by the user

	internet connection is available.	• Page refresh rates/overall survey load times are not dependent upon the strength of the internet connection • Specific survey content can be optimized to a known operating system to reduce the overall amount of data exchanged • Can reliably deploy video/flash/image/audio content • Automates data submission and can provide survey alerts/invitations • Content pacing does not rely on strength of internet connection	• Cost of development/data capture/storage may be high for single-use surveys • Newer approach to data collection- users may have to receive some guidance on how the app works • Number of available APPS on all marketplaces continues to rise- may have a harder time with salience or may have to compete with "app clutter" • Some question types may not be readily available in App toolkits- may require additional programming to implement non-standard question types
A hybrid of the A-MBS approach and the ABSS approach.	Specifically, surveys deployed using the app-like mobile browser survey approach (A-LMBS) are completed using the mobile web, but they rely on active browser refreshment using a combination of server-side scripting and JavaScript to create "Active" Mobile web pages. The web pages appear app-like with no web address bar present after the survey commences and only new portions of the screen must be loaded on each subsequent page	• Can capture various types of data including GPS, Pictures and Video using JavaScript to activate these smartphone features • Portions of the online layout can be "preloaded" and do not need to be reloaded with each new screen (e.g. back/next buttons) • Placement of navigation bar can be controlled and can be made "persistent" (i.e. will not require reloading the entire ribbon on each page) • Data collection can occur without the need for a persistent internet connection, but will require additional JavaScript programming • Page refresh rates/overall survey load times will generally be shorter compared to Active Web Browser approach • No internet web address bar is present after the survey begins- the survey is "encased" in what appears to be an App potentially increasing available screen size • Can take advantage of the smartphone's native features including loading wheel, spinners and slider bars • Survey presentation is more uniform across smartphone operating systems • Graphic images can be optimized and scaled for mobile browsers • Can control questionnaire layout and suggested page orientation	• Requires additional JavaScript programming to make survey web pages "active" rather than dynamic • Limited number of pre-programmed survey question/response formats available • Approach will not be optimal for non-touchscreen smartphones • Initial (i.e. first page) amount of data exchanged on the introduction/welcome page will generally be higher compared to the A-MBS approach • Cannot explicitly control the orientation of the survey (i.e. landscape or portrait) • If JavaScript is disabled on a respondent's smartphone, then some functionality will not work (e.g. older Blackberry devices) • Some data capture regarding GPS and other features may require additional permissions to be set by the user • Approach is not currently prepackaged or part of survey software.

2.2 (Trusted) Smart Survey Taxonomy

More advanced smart surveys use internal sensors (sensors that are available in the device), external sensors (other sensors close to the device) and/or public online data. In order to do so, the survey usually demands explicit consent from respondents. In case of internal sensors such consent questions are automatically posed to respondents (Schouten, 2020).

A smart survey employs one or more of the following smart features:

1. Device intelligence: It can use the intelligence (computing, storage) of the device on which it runs
2. Internal sensors: It can employ the sensors that are available in the device
3. External sensors: It can communicate through the device with other sensors close by
4. Public online data: It can go online and extract generally available data
5. Personal online data: It can go online and request access to existing external personal data
6. Linkage consent: It can ask consent to link external personal data already in possession of the survey institute

A simple taxonomy of smart survey is the following:

- Simple smart survey: Uses only the device intelligence (computing, storage);
- Internal smart survey: Uses internal sensors (sensors that are available in the device) to add sensor data
- External smart survey: Uses external data (other sensors nearby and connected with the device), either collected by external sensors or available in existing sources
- Full smart survey: Uses both internal sensors and external data.

Smart surveys and trusted smart surveys can be initiated within (standard) browsers or through dedicated applications. Applications demand a separate installing procedure and, as a consequence, are only a viable option when the survey is longitudinal and/or employs more advanced sensors.

So what is different for smart surveys?

- It can initiate sensors and store the sensor data
- It can go online and download data
- It can communicate to nearby sensors and collect data (e.g. through Bluetooth)
- It can use internal sensor data and/or external sensor data and/or online data:
    - as supplements to survey data
    - to execute decision rules in the questionnaire
    - to perform plausibility/edit checks
    - to facilitate respondents in answering questions
- It can perform part of the interaction (reminders, motivation, feedback) with respondents locally
- It can perform part of the data processing locally instead of at the survey institute
- It introduces new representation errors (willingness to use sensors, availability of sensors, missing sensor data, errors in data transmission) and measurement errors (inaccuracy in sensors, incompetence/inability of respondents).

And what is different for trusted smart surveys?

- It can ask for consent from respondents to connect to an external database
- It can request data after consent through external database API's
- It can perform linkage to existing data at the backend of the survey institute

- It can use the downloaded external data:
    - as supplements to survey data
    - to execute decision rules in the questionnaire
    - to perform plausibility/edit checks
    - to facilitate respondents in answering questions
- It introduces new representation errors (willingness to consent, missing external data, data transmission/linkage failures) and measurement errors (incompetence/inability of respondents, errors in external data).

2.3 Subjective and Objective Measures

The collection of various types of data generated by the mobile device depends on the chosen strategy. Indeed, it is possible to acquire subjective measures (similar to questions or diaries) when people are the source of the collected data, or objective measures when the device itself is the source of the collected data (Jäckle et al., 2018). Objective measures are mostly passively collected data (without respondent intervention or feedback, apart from consent) that automatically measure characteristics and behaviours of users, network size and characteristics, type of smartphone usage, multiple types of sensor data and so forth. Also, data from videos and photographs can also be counted as objective measures, when they are not coded or classified by the users themselves.

During data collection, respondents take an active role: they can be interviewed by phone, video, or through SMS or chat exchanges. Moreover, data can be supplemented by other forms of data that are provided by respondents, such as through time-use diaries, audio recordings, photographs, or videos (time use diaries can be used to record people's activities, location and enjoyment; pictures or videos can be used to replace survey questions; voice input can be used to answer open-ended questions). Thus, the active collection of data from mobile devices might enrich traditional survey data, reduce response burden and might be accurate. Nevertheless, there is some self-selection associated with participating in mobile web surveys and in activities which require the involvement of respondents. Beyond the active collection of data, passive data collection can be used to acquire information on attitudes, behaviours and mobility patterns, on how respondents used their mobile devices. Passive data collection can substitute existing factual questions, it can generate additional data, and it can be used to optimize the timing of surveys. Consequently, the response burden for participants is reduced, due to a reduction in survey questions and hence, survey length.

The decision on the type of data collected – subjective and/or objective measures – has consequences on whether a mobile survey application is needed or not. In case survey designers conduct their survey with the help of mobile survey application (purpose-designed applications), they can make use of a wide range of data collection approaches. In this regard, when survey designers use mobile survey applications, which are designed for their purpose, then a further differentiation between active and passive data collection is useful.

For passive data collection, the consent of respondents is required in most countries (sometimes for each type of passive data collection separately). Consent depends on the type of information requested, the context in which the information is disclosed, etc. In active sensor data collection, respondents are asked to check, revise, accept and/or supplement sensor data, i.e. the respondent is involved in data collection. Motives for active sensor data collection are increased response rates, increased data quality and hybrid forms of survey and sensor data. Active data collection is much more demanding as it requires real-time data handling and a careful design of a user interface (Mussmann and Schouten, 2019 - ESSnet Mimod WP5-3).

2.4 Primary and Secondary Data Collection

When sensor data can be sent to the user directly we have a form of primary data collection, while we have a form of secondary data collection when sensor data are collected and owned by various parties (Mussmann and Schouten, 2019 – ESSnet Mimod WP5-3). For example, IoT sensors (user-owned sensor) fall into the latter form of data collection, while wearable sensor data can be placed in both primary and secondary data collections. In smart surveys, however, we are interested in secondary sensor data if they can be associated to sampling units of a target population after respondent consent.

Apart from privacy and legal constraints, such user-owned sensor data can potentially be linked to individual respondents in surveys. After linkage, a hybrid data collection follows where part of the data may be asked through questions, part of the data may be measured on respondent mobile devices and part of the data may be linked. An example is activity tracker data owned by a third party linked to persons in the sample supplemented by survey data on health perceptions and health determinants. Another example is budget expenditure diary data linked to supermarket scanner data and supplemented by scanned receipts for other types of stores. IoT sensor data do not necessarily include identifying information about the persons to which data correspond. As a consequence, linkage of IoT sensor data to individual respondents may be hard or even infeasible without additional information such as time stamps and/or location coordinates. Nonetheless, hybrid forms of data collection may arise where respondents are asked for additional information that enables linkage (Mussmann and Schouten, 2019 – ESSnet Mimod WP5-3).

Secondary sensor data pose the challenge of collaborating with a third party company, as well as legal and ethical challenges. It does, however, offer access to a trove of data without – or very little – respondent burden and is, thus, worth investigating. The potential impact on the privacy of the respondents suggests that secondary sensor data should be offered as a choice to the respondents with a clear explanation of the benefits and risks involved. Main secondary sensor data useful in (social) smart surveys are: social media data, mobile phone provider data, Internet provider data, smart energy use meters data (electricity, water), wearable sensor data, scanner data from shops, bank transaction data, loyalty card data, etc. Secondary data have strong features from the viewpoints of omnipresence, costs, burden and feedback, but weaker features when it comes to data access and data handling. Quality and respondent willingness to link data are mostly unknown and need to be assessed in real studies (Mussmann and Schouten, 2019 –ESSnet Mimod WP5-3).

By overcoming privacy concerns and legal constraints, user-owned sensor data can potentially be linked to individual respondents in surveys. In this situation, a hybrid data collection is performed by linking data acquired through questions, and data measured on respondent mobile devices. IoT sensor data do not necessarily include identifying information about the individual to which data correspond. Linkage of IoT sensor data to individual respondents requires additional information, such as time stamps and/or location coordinates that enable linkage (Mussmann and Schouten, 2019 –ESSnet Mimod WP5-3).

3. Devices and Sensors: an Overview

3.1 Smart Devices

A smart device is an electronic device, generally connected to other devices or networks via different wireless protocols such as Bluetooth, NFC, Wi-Fi, cellular network, etc., that can operate to some extent interactively and autonomously. Smart devices are typically composed of a hardware layer (including a radio

that transmits signals), a network layer (through which devices communicate with each other), and an application layer (through which end users deliver commands).

Two key elements in this definition are the observation that the smart device has to be *interactive* and *autonomous* to some extent. Interactive meaning that users can somehow influence the behaviour of the device, autonomous meaning that the device can operate for a longer time without human intervention to do its task. Moreover, a smart device is part of the IoT – a system of interrelated computing devices, mechanical and digital machines provided with unique identifiers (UIDs) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction. IoT covers a wide range of objects including autos, household appliances, scanners, traffic lights, security cameras, wearable sensors, GPS locators, and so forth. Because of it connectedness, but not every IoT object is a smart device because it might lack interactiveness and autonomy (ESSnet Big Data II, Workpackage L – WPL-3).

Smartphone and tablet

In line with former methodological research on the use of mobile device sensors, we refer to a mobile device as a "[...] computing device small enough to hold, operate in the hand with a flatscreen interface, that has many of the following functions: It has a mobile operating system – which is not a full-fledged PC operating system – that allows third-party apps to be installed and run; It can place and receive phone calls; It can connect to the Internet; It can interconnect with other devices through one or more wireless channels (Wi-Fi, Bluetooth, GSM or NFC); It has an integrated camera; It has a Global Positioning System (GPS)" (Mussmann and Schouten 2019, p. 2).

Therefore, smartphones are not only communication devices, but also processing and sensing devices with rich sets of embedded sensors (accelerometers, digital compasses, gyroscopes, positioning systems, microphones and cameras). These sensors make it possible to develop new applications in many domains, leading to a new research area known as *mobile phone sensing*. The increasing size of the data generated by sensors and applications lead to a new research domain across computing and social science.

Other types of sensors include wearable sensors, which detect the movement, and the Bluetooth sensor, which enables the transfer of data from one device to another using the data communication channels. Sensing and recording become possible with these mobile sensors, which help to monitor the subjects constantly. Activities can also be stored when monitored in their preferred environment.

Wearable devices

Wearable devices (e.g., smartwatches, fitness bracelets, etc.), also known as a wearable or body-borne computer, is a small computing device worn on the body. Their functionality is more specific and targeted than smartphones and tablets. Wearables are often health-related, but do support other functions. Most common wearables are activity trackers, fitness bands and smartwatches, which are intended for continuous use. An activity tracker, also known as a fitness tracker, is a device or application for monitoring and tracking fitness-related metrics such as distance walked or run, calorie consumption, and in some cases heartbeat. Due to their more specific functionality, wearables often require other mobile devices to communicate with users. A dedicated app needs to be installed on a smartphone or tablet that provides a user interface to set or alter wearable settings and to read summary data and statistics.

Wearable sensor data are mostly self-initiated but often are maintained and owned by another party. These data can be sent to the user directly or indirectly through the producer of the wearable. In the latter case,

often only edited and aggregated data are available and the raw sensor data remain at the side of the producer.

Due to the closer proximity to a respondent's body and the fact that they can be worn 24 hours per day and 7 days per week, wearables can measure data that smartphones and tablets cannot. They may incorporate special sensors such as accelerometers, gravity, gyroscope, Bluetooth, GPS, thermometer, heart rate monitors, vibration, etc. Less common sensors are: blood oxygen levels, camera, light sensor, magnetic field, NFC (near field communication), speaker, thermometer, Wi-Fi (Mussmann and Schouten, 2019 - ESSnet Mimod WP5-3). Wearable sensors have been widely used in medical sciences, sports and security.

3.2 Sensors in Mobile Phones and Wearables

Mobile phones have computer platforms with numerous sensing features such as user location sensing, ambient light measurement, sensing device orientation, recording of high-quality audio, geomagnetic sensing, etc. They also contain specific sensors capable of recording, reading/obtaining, categorizing, analysing, and transmitting different types of data (Ali, 2014).

Sensors in mobile phone can be categorized into physical sensors and virtual sensors: the first are hardware-based sensors embedded directly into mobile phone devices and derive their data directly by measuring particular environmental characteristics (accelerometer, gyroscope, proximity, etc). Virtual sensors, also called synthetic sensors, are software-based sensor deriving their data from several hardware-based sensors (e.g. for the Android platform linear acceleration, and gravity sensors).

Focusing on the physical sensors, it is possible to distinguish in embedded **(Em)** or external **(Ex)** sensors. Embedded sensors are integral parts of the devices and are accessible through predefined interfaces (accelerometer, gyroscope, etc.), while external sensors in the environment need devices to be put into communication (wireless protocols, Bluetooth, etc.). Another characteristic of sensors is to be proprioceptive **(PC)** or exteroceptive **(EC)**. Proprioceptive sensors determine/measure physical properties related to the internal conditions of devices/systems, whereas, exteroceptive sensors obtain information from the environment outside of the device. Exteroceptive sensors can be divided into contact and non-contact sensors. Contact sensors contain the same modalities as used in proprioception and non-contact sensors contain modalities which could be used for measuring physical properties at a distance such as direction, size, intensity, and range etc. Finally, sensors can be divided into passive **(P)** and active **(A)**: the former measure the energy generated in the environment external to the device and do not require power or battery, while the latter emit energy into the environment by measuring the generated reaction and require power or battery to operate.

The combination of mobile phones and sensors can be obtained either with embedded sensors provided as integral parts of phones or positioned on the screen accessed by using the mobile phone APIs, or external sensors connected to mobile phones with wireless network technologies. Most of the today's smartphones are open and programmable and come with several embedded and external sensors.

In the following tables, we report the main sensors present in mobile phones (Ali, 2014) and wearable devices, describing the category of belonging, characteristics and type of associated data (structured, semi-structured, unstructured). All sensors listed are objective measures, except for the microphone that can be both objective and subjective.

Table 2. Sensors in mobile phones

Sensor category	Sensor type	Short description	Characteristics	Type of data (structured, semi-structured, unstructured)
Tactile	Proximity	Proximity sensor releases electromagnetic or electrostatic field or a beam of electromagnetic radiation and looks for changes in the field or return signal and detects any nearby object presence without any physical contact. In smartphones, proximity sensor senses that how close a phone's screen is to a user's body while bringing the phone near to his/her ear. Proximity sensor will automatically turn off and lock the screen not only to save battery power but also to protect against any unwanted input by accidentally touching the screen by a user's ear or check/face. As soon as, the mobile phone is taken away from the ear, it will be automatically restored to its previous state.	Em/Ex EC P/A	Structured (On/Off)
Acceleration	Accelerometer	Accelerometer sensor is used as the user interface controller: changing the screen display by sensing the orientation of the device based on the way the device is being held by a user.	Em PC P	Structured
	Gyroscope	Gyroscope sensor is a movement sensor pretty much like accelerometer. It uses principles of angular momentum for measuring and maintaining the position and orientation of devices. Gyroscope when combined with the accelerometer will allow devices to measure motion along six axes: left, right, up, down, forward, and backward, along with roll, pitch and yaw rotations and will provide more accurate motion measuring capabilities to the devices.	Em PC P	Structured
Thermal	Temperature	Temperature sensors give information about the ambient temperature. This sensor uses solid-state principles to determine the temperature instead of using mercury, bimetallic strips, or thermistors. There are two types of temperature sensors: contact sensors and non-contact sensors. Contact temperature sensors calculate the temperature of objects to which they are physically connected by assuming that there	Ex EC P/A	Structured

Table 2. Sensors in mobile phones				
Sensor category	**Sensor type**	**Short description**	**Characteristics**	**Type of data (structured, semi-structured, unstructured)**
		is no heat flow between the sensor and object (e.g. thermocouple sensor, and thermometer sensor etc). Non-contact temperature sensors receive thermal radiant power of the Infrared or Optical radiation from the surroundings (objects) (e.g. pyrometer sensor etc).		
Image	CMOS Camera	A CMOS image sensor uses MOS (Metal Oxide Semiconductor) transistors to convert an optical image into electrical signals. This sensor forms an image by using unit pixel, where each pixel is a semiconductor which creates current signals by transforming incident light photons and the size of signal produced is relative to the amount of incident light photons.	Em EC P/A	Structured/Semi-structured
Light	Ambient Light	This sensor measures light of the surrounding and adjust brightness of mobile phone accordingly to optimize screen visibility. Adjusting display brightness will not only optimize visibility but will also save battery power in smartphones.	Em/Ex EC A	Structured/Semi-structured
	Back-Illuminated	Back-Illuminated sensor is a digital image sensor which improves low-light performance and increases the amount of light during image capturing and makes more prominent image elements.	Em EC A	Structured/Semi-structured
Water	Moisture	Moisture sensor is used to determine the cause of smartphone damage.	Em EC P	
	Humidity	Humidity sensor,or hygrometer, measures the relative humidity (i.e. both air temperature and moisture) present in the environment/air.	Ex EC P	Structured/Semi-structured

Table 2. Sensors in mobile phones				
Sensor category	**Sensor type**	**Short description**	**Characteristics**	**Type of data (structured, semi-structured, unstructured)**
Location	Digital Compass	This sensor is a magnetometer, used to recognize the North and defines direction for users.	Em EC P	Structured
	GPS	GPS is a navigation tracking system where GPS receivers get information sent by the GPS satellites and calculates a user's exact location using triangulation. In smartphones, GPS applications can be programmed with a map in the background thus displaying to the users the routes where they have been or want to go.	Em EC A	Structured
Height	Altimeter	Barometric Altimeter sensor, is a mechanical device designed to measure altitude above Mean Sea Level (MSL) using the idea of change in the atmospheric pressure. It works on how the pressure and temperature change with altitude. In smartphones, altimeter sensor can be used to detect a user's elevation such as letting users know on which floor of a building they are or users can more precisely determine their location to their friends on a location-based service, etc .	--- EC P	Structured
	Barometer	Barometer performs pressure measurements. With barometer, GPS would work faster and more accurate because barometer would calculate one of the 3 main values (i.e. longitude, latitude, and altitude) to facilitate the work of GPS and make it faster.	Em EC P	Structured
Medical	Heart Rate Monitor	Heart rate monitor allows users to monitor their heart rate in real-time or record their heart rate for later use. Heart rate monitors are of two basic kinds: strap heart rate monitor and wrist heart rate monitor.	--- EC P	Structured

Table 2. Sensors in mobile phones				
Sensor category	**Sensor type**	**Short description**	**Characteristics**	**Type of data (structured, semi-structured, unstructured)**
	Biosensor	A biosensor can detect, record, and transmit physiological data using electrical signals. Without using the biological system directly, they can determine the concentration of substances and other parameters of biological interest. A biosensor has two components: a bioreceptor (a biomolecule that recognizes the target analyte) and a transducer (which converts the recognition event into a measurable signal) and these two components are integrated into one single sensor, which enables that without using reagents, one can measure the target analyte. They can provide valuable context information in different applications like sports and medical while measuring skin resistance, and blood pressure etc.	--- EC P	
Acoustic	Microphone	A microphone sensor is an electromechanical device that detects air pressure as vibration and creates an electrical signal that is proportional to the vibration. Microphones can provide very interesting information with minimal processing such as noise level, types of input (noise, music, and speaking etc), and base frequency.	Em EC P	Unstructured
Radio	RFID	RFID is the abbreviation of Radio-Frequency Identification, an automatic identification method that works by storing and retrieving data remotely using devices called RFID tags or transponders.	--- EC A	Structured
	Bluetooth	Bluetooth sensor is a short-range low-power radio communication device, designed primarily to connect personal consumer gadgets and peripherals available in proximity in a wireless network. In smartphones, Bluetooth sensor can be used for communication with other external computing devices which could be sensors or any other communication devices.	Em EC A	Structured

Table 3. Common sensors in wearable devices

Sensor type	Short description	Type of data (structured, semi-structured, unstructured)
Microcontroller	This is typically viewed as a mini-computer (system on a chip). It enables the Internet of Things (IoT) to be present in this application. It is used desirably in wearable technology because of its simplicity to program, reprogram, cost, size, compatibility with other sensors, and the ability to control complex outputs, such as graphical displays.	
Accelerometer	Accelerometers are a common sensor found in wearables. Their sensing capabilities range from different types of accelerations (linear and gravity). Their measuring capabilities allow monitored data to be programmed for different uses (sport and medical). Accelerometer-based wearable can produce a diverse range of meaningful data.	Structured
Gyroscope	Gyroscopes are another common sensor found in wearables. They differ from accelerometers in that they measure angular accelerations exclusively. Some applications will prefer to use the accelerometer to determine rotational acceleration, whereas some would want to combine both in order to filter errors. This is to increase the accuracy of the monitored data.	Structured
Magnetometer	Magnetometers can typically be combined with accelerometers and gyroscopes to form the Inertial Measuring Unit (IMU). Each of these sensors can possess three axes each, depending on the type. It is very similar to what a compass does, and it helps with coordination. Whilst it is normally used with the other two sensors, it	Structured

	complements them by filtering the orientation of the movements. Magnetometers measure magnetic forces in relation to Earth's magnetic field.	
Heart Rate	There are a variety of sensors and techniques that can measure heart rates (as Fitness trackers, like Fitbit).	Structured
Pedometer	Pedometers are commonly found in lifestyle-based fitness wearables. They are used to count a user's steps. This can be in the form of walking or running. There are two versions of pedometers, mechanical and electrical. The latter is the most common form presently and relies on MEMS for accuracy but still works on principles based on mechanical pedometers. The accuracy of sensory measurement depends on wearable positioning.	Structured
Pressure	Pressure sensors typically work from strain gauges. When forces are applied on sensors, they produce a resistance change in the circuit. Mechanical quantities such as force are experienced in multiple ways for sport and are converted into an electronic measurement dependent on resistance. This form of measuring strain is done by a Wheatstone Bridge formation, which can detect resistance changes in static or dynamic form.	
GPS	GPS is a very common sensor found in multiple appliances (smartphones). It is used for navigation, as it informs users about their location. Data are sent to a satellite where the precise location and time are measured. This works as a transmitter and a receiver, where the information is fed back into the sensor to inform the location. It is used in wearables to measure key data, such as distance, which can be viewed in different ways for different applications.	Structured

3.3 Secondary Sensor Data

Secondary sensor data are collected and owned by various parties but not for use in surveys or official statistics. Generally, these are data collected for different purposes, such as for business insides based on statistics or functionality needs, which therefore need to be transformed, structured for the purposes of official statistics.

Secondary sensor data can be distinguished on the basis of sensors owned by the user and data based on public IoT sensors (i.e. not owned by private users). However, the main distinction is between individual and public use: privately owned IoT type sensors, such as weather stations or burglar protection systems, can be seen as user-owned sensors (Mussmann and Schouten, 2019 - ESSnet Mimod WP5-3). Within user-owned sensors, a distinction can be made between commercial and non-commercial data collectors. A range of companies produces apps for mobile devices, in particular wearable devices, to provide paid services to individual customers. Other companies, such as Google, also provide unpaid services in exchange for the right to use the resulting data for commercial purposes. These services are usually continuous and, consequently, create a dynamic, vast stream of sensor data. The resulting data are mostly based on self-selection, i.e. the initiative is with the user, and not based on invitation. Non-commercial parties may collect mobile device sensor data for research or policy-making motives. Such data collection often has a finite time horizon and is invitation-only. In all these cases, however, the sensor data are self-initiated by users, but the data is stored, handled and owned by others (Mussmann and Schouten, 2019 – ESSnet Mimod WP5-3).

4. Machine Learning Techniques: an Overview

4.1 Algorithms, Tasks and Sensors

"Machine learning is the science (and art) of programming computers so they can *learn from data*." (Géron 2017, p. 4, emphasis in the original). A more formal definition of machine learning due by Tom M. Mitchell (2013) is: "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E." Machine Learning's algorithms build a model, based on data known as "training data" (experience), in order to make predictions or decisions (tasks) without being explicitly programmed to perform the task (machine learning).

We live in a datafied world in which every action of our daily life produces a huge quantity of data. This enormous propagation of data and its exponential growth, on one hand, represents a valuable resource for the study of phenomena and processes of different nature (economic, social, demographic, etc.), but, on the other hand, they require the use of special techniques for statistical data processing. Machine learning is one of these techniques.

The 3Vs (volume, variety and velocity) are three defining properties or dimensions of big data. The big data produced by the modern digitized and datafied society, combined with ML techniques for their processing, allows us to build predictive models, aiming to analyse several processes and phenomena. Machine learning algorithms are typically classified into the following groups:

- Supervised
- Unsupervised
- Reinforcement learning

Supervised algorithms require a labeled dataset (training set) in which the relationship between the input variables and the output variable is explained through a set of examples. At the end of the learning phase, the algorithms approximate the rule that links the input variables to the output one; this category of algorithms includes, for example, regression (when the output variable is numeric) and classification (when the output variable is categorical).

Unsupervised algorithms require an unlabeled dataset (training set) in which there are not meaningful tags or labels associated with the data; the purpose of unsupervised algorithms is to find hidden structures (patterns) in the data. For example, the clustering (e.g. k-Means) and dimension reduction algorithms (e.g. SVD) are unsupervised.

In reinforcement learning, the learning phase takes place through software agents who act in an environment, in order to maximize the notion of reward. Unlike supervised learning, the reward is awarded only after a series of actions by the agent. Some application examples are self-driving cars and the chess game, in which the reward takes place only in case of success at the end of the whole game and not for each single move.

How does all this relate to the world of sensors? How should the data produced by the smart devices used by humans, mainly smartphones and wearable devices, be managed through machine learning algorithms? We try to answer to these questions through some considerations, taking into account the context of official statistics.

The National Statistical Institutes (NSI_s) have conducted several experiments (ESSnet Big Data, UNECE HLG-MOS Machine learning project) in big data topics and in particular in the study of their possible use as an integrative data source in the production of official statistics. The first of these mainly concerned data sources from social media or web scraping activities; they then moved on to the study of data sources held by mobile providers, to finally experiment with data coming from smart sensors. Smart sensors represent one of the main hardware resources through which huge amounts of different types of data are collected.

When we talk about sensors, we should distinguish smartphones with their multiple advanced embedded sensors, by all those sensors used in home or industrial automation, in climate or transport and traffic management, etc. All these smart sensors in their whole represent the IoT.

Smartphones have a particular role in the IoT ecosystem due to their spread over the population, this feature allows them to be useful in smart statistics where statistical units are subsets of the population. Furthermore, smartphones often interact with IoT devices in the environment (e.g. smartwatches or smarthome devices) that can be useful to integrate information in a smart survey. The data produced by smart sensors represent a valuable source of data potentially very useful for producing official statistics as well. This data could be integrated with traditional sources with the aim of obtaining not only more accurate statistics, but help to get closer to real-time reports.

There is no direct relationship between the data collected by each type of sensor and a machine learning algorithm for their processing. The algorithms to be applied to the data collected by the sensors depend on the predetermined use cases; we must define the use case, then identify the necessary data sources, and finally apply the most suitable machine learning algorithm for that use case. For instance, the same data from a sensor, such as the signal of a microphone, can be used to recognize the gender of the speaker's voice in a classification task, or the noise in a room, measured in decibel in a regression task. We represent this concept

in table 4 where the machine learning task and the type of label, defined from the use case, are associated with an appropriate list of ML algorithms.

Table 4. ML algorithms related with data structure

ML Task	Label Type	Machine Learning Algorithm
Classification	Categorical	SVC, KNN, SGD Classifier, Linear SVC
Text Classification	Categorical	Naive Bayes
Regression	Quantity	SGD Regression, Lasso, SVR
Clustering	No Labeled	Kmean, Spectral Clustering, Mean shift
Dimensionality Reduction	Quantity	PCA, Isomap, LLE, kernel approximation

In the following table 5, we report a list of use cases which explanation is available in the "Links" column and for each of them the related task, the sensors used and the Machine Learning algorithms applied in the use case.

Table 5. ML algorithms and sensors

ML algorithms	Use case	Sensors	Links
Principal Component Analysis (PCA)	Activity recognition	AccelerometerMagnetometer	https://ieeexplore.ieee.org/document/8368718
Random Forest, Support Vector Machine (SVM), Naive Bayes	Human Activity Recognition (HAR)	Gyroscope	https://ieeexplore.ieee.org/document/8944512
Classification and regression	To predict the number of person in a closed space	Temperature	https://www.researchgate.net/publication/220066130_Using_Machine_Learning_on_Sensor_Data
Decision Tree, Linear regression	To predict the number of people in a lab.	Temperature, Ambient Light, Back-Illuminated, Humidity	https://www.researchgate.net/publication/220066130_Using_Machine_Learning_on_Sensor_Data
Logistic regression, K-nearest neighbours (KNN), various decision trees including, SVM, and Naive Bayes	*Physical* Activity Monitoring Using Smartphone Sensors and Machine Learning	Digital Compass, GPS	https://towardsdatascience.com/physical-activity-monitoring-using-smartphone-sensors-and-machine-learning-93f51f4e744a
KNN, Decision Trees and Naive Bayes classifiers	Floor Identification Using Magnetic Field Data With Smartphone Sensors	Altimeter Barometer	https://www.researchgate.net/publication/333592510_Floor_Identification_Using_Magnetic_Field_Data_With_Smartphone_Sensors/fulltext/5cf5cf6da6fdcc847502da49/Floor-Identification-Using-Magnetic-Field-Data-With-Smartphone-Sensors.pdf

Logistic regression	A novel three-tier Internet of Things architecture with Machine Learning algorithm for early detection of heart diseases	Heart Rate Monitor	https://www.sciencedirect.com/science/article/abs/pii/S0045790617328410
SVM, Decision tree, Random forest	Audio IoT Analytics for Home Automation Safety	Microphone	https://www.researchgate.net/publication/330626246_Audio_IoT_Analytics_for_Home_Automation_Safety
SVM, Linear regression, decision tree	Automatic detection of false-positive RFID readings using machine learning algorithms	RFID	https://www.researchgate.net/publication/319672231_Automatic_detection_of_false_positive_RFID_readings_using_machine_learning_algorithms
Logistic regression	Improving the power of activity-based heat detection using additional automatically captured data	Pedometer	https://www.researchgate.net/profile/Kathryn_Hempstalk/publication/259042983_Improving_the_power_of_activity-based_heat_detection_using_additional_automatically_captured_data/links/00b7d529cf9c876627000000/Improving-the-power-of-activity-based-heat-detection-using-additional-automatically-captured-data.pdf

4.2 Training Models

In general, the construction of a training set is essential for the training of machine learning models. The construction of the training set can be obtained, for example, by searching among the external datasets made available by the scientific community. The external dataset must have the same distributive characteristics for the input and output variables as those of the data collection. This is desirable if we want to have a good generalization ability of the trained model. Another technique, with which to build a training set, is to label a subsection of the data in the data collection in a semi-automatic or clerical manner. If the sampling is properly performed, this technique prevents a problem that arises by using an external dataset.

To train a machine learning model, a data preparation phase is generally required. Data preparation is a fundamental phase for producing good quality models. The quality of the output strongly depends on the input data used. Better quality data will lead to better quality results. The data preparation with its various sub-phases improves the quality of data through the handling of non-responses items, outliers, missing values, etc.

4.3 Quality Aspects in ML Algorithms

As we have seen so far, ML algorithms can be very useful in the statistical production process phase; we mentioned them both as a support tool for example for classification tasks and as a tool for handling unstructured data, typical of smart statistics.

Smart surveys must ensure the same quality standards of the output and the processes, as assured by traditional surveys. The current and traditional quality frameworks must therefore be expanded to take into account the ML algorithms typically used in the intermediate outputs of statistical production processes of smart surveys. A very interesting framework from this point of view is the Quality Framework for Statistical Algorithms (QF4SA), proposed in the HLG-MOS ML context. The QF4SA proposes the following five quality dimensions:

- Explainability
- Accuracy
- Reproducibility
- Timeliness
- Cost-Effectiveness

As with most quality frameworks, the five dimensions have to be considered jointly; one may choose to place more emphasis on particular dimensions but none should be ignored.

Explainability

The explainability is the one of dimension of the QF4SA. It is defined as the degree to which a human can understand how a prediction is made by a statistical algorithm using its input features.

Generally, a greater complexity of the chosen algorithms leads to a lower prediction error but also a lower explainability. The good way to use ML techniques without having a quality reduction in the output is to find a right trade-off between the complexity of the algorithms and the quantity of data on one side and the explainability of the algorithms used on the other side.

Accuracy

In the context of QF4SA we have a broader and more general definition of accuracy than that typically used in the context of the quality measurement metrics of ML classification algorithms.

The accuracy according to this framework indicates the degree to which the phenomenon we are measuring is described; it is the closeness of computations or estimates to the exact or true values that the algorithms were intended to measure. To measure this closeness to real values, different metrics are used in machine learning tasks; the metrics typically used depend on the type of task you want to solve. For example, typical metrics for classification tasks are *accuracy*, *precision*, *recall*, *f1-score*. Instead, the *mean absolute error* (MAE) or *mean square error* (MSE) is used for regression tasks.

To obtain a good quality of accuracy it is important to take this aspect into consideration: the value of a metric generally depends on the training set we are using, so different training sets will produce different metric values. In particular, among these training sets, there may be some that are not very representative of the reality we want to represent; we can measure this effect with some training set resampling techniques as the cross-validation or the bootstrap.

Reproducibility

In general reproducibility is the ability of a researcher to reproduce the same results of another research starting from the same raw data and using the same methods. Reproducibility is a necessary condition for a study to be credible and is the very basis of the scientific method.

The QF4SA recognizes three types of reproducibility

- *Methods reproducibility* occurs when the same results are obtained using the same methods and data;
- *Results reproducibility* occurs when corroborating results are obtained using the same methods but in a new study with new data;
- *Inferential reproducibility* occurs when knowledge claims of similar strength are obtained from a study replication or reanalysis. This is not identical to results reproducibility, because not all investigators will draw the same conclusions from the same results, or they might make different analytical choices that lead to different inferences from the same data.

Timeliness

In the context of QF4SA timeliness is the time taken to produce a result from conceptualization, algorithm construction, processing and production. A distinction must be made between timeliness in development and production phases in machine learning algorithms. The former phase generally takes longer than the latter. The definition in QF4SA that includes the processing of machine learning is different from the timeliness defined in the usual official statistical frameworks, where it is referred to the speed of the dissemination of the data. ML can lead to substantial gains in timelines. Despite a development phase slower due to correct choice of the algorithm, hyper-parameters, an opportune features selection and a training of the model, there is a substantial gain in phases where the manual work was important; coding is a prime example.

Cost effectiveness

The last, but not the least dimension of the QF4SA is the cost effectiveness. It represents the degree on which results are effective in relation to the costs they need to reach them. Costs can be either fixed or ongoing. Some examples of fixed costs are those needs to acquire the necessary hardware and software or to train the staff on ML techniques. Ongoing costs are related to the acquisition of necessary on cloud storage space, or to the IT infrastructure maintenance or to keep the staff up to date regarding new ML developments. Both traditional methods and modern ones have fixed costs, but ML may have advantages due to scalability and automation; in this way the initial costs may be amortized over following years. When considering algorithms, the QF4SA is recommending to evaluate two aspects of cost: cheaper operating costs and time to recoup fixed costs.

SECTION 2 – ERRORS IN SMART SURVEYS

5. Sources of Error in TSS_u

Relying on smart devices in the hands of respondents has consequences in terms of representation/selection (row errors) and measurement (column and cell errors). The first type of errors is substantially determined by the availability or not of a smartphone or other mobile devices by the individuals selected in the sample (coverage error) or by their willingness to participate in a smart survey that may require considerable effort for respondents (non-response). The non-participation of respondents in the transmission of sensor data passively collected is determined by multiple factors, some of a technological nature (availability, ability), others of a more general nature, such as the perceived benefits or costs of participating, as well as the concerns of respondents about privacy and trust in the organization of the survey.

For representation, the errors for smart surveys coincide with those for traditional surveys up to response, but participation in the survey may be reduced by technological barriers or other causes: respondents need to have access to sensors or external data, they need to be willing to perform sensor measurements and/or provide access to sensor data and other forms of data, they need to execute the sensor measurements or link data, and the resulting data should be processed and transmitted. In all these steps, population units may drop out and cause representation to be selective and unbalanced across relevant population characteristics.

Measurement errors must be viewed and analysed according to whether it is a data acquired with traditional tools (questionnaire, diary) or via apps (sensors). While for the former the measurement errors are mainly due to the behaviour of the respondent (for example, the potentially more distracting environment of mobile device users) and the use of the smartphone (for example the small screen size of mobile devices), for the latter new errors may occur during the data collection and processing phases for different reasons. For traditional data resulting from questionnaires or diaries, we can identify measurement errors specific to a mobile web mode (mobile browser surveys, app-based smartphone surveys) assuming that a mobile version modified and adapted to account for differences in browser size is adopted. Familiarity with smartphones may affect response quality, as well as the context in which the answers are provided: distractions/ multitasking can reduce the attention and cognitive effort of respondents (satisficing bias); the presence of other people can affect the responses (social desirability bias).

Many measurement errors can be common for both traditional and sensor data. However, some distinctions need to be made between the two types of data with regard to cell errors, in particular for content errors that can be the result of both different measurement errors and data processing errors.

For sensor data, measurement errors occur during the data collection process (missing, outliers, etc.) or are the result of the processing phase. These errors can be classified as: errors generated by the smartphones themselves; operating errors produced by the interviewees who can incorrectly initialize the measurements or position the devices in the wrong points; random or systematic errors resulting from quality and age of the sensor; errors due to data manipulation to search for patterns or to explore the accuracy and precision of data; errors for inadequate or incorrect treatment of missing or outliers; errors in calibrating measured data; errors in the combination of multiple sensor data.

Methodological solutions for TSS_u, aimed at preventing and managing different types of errors, must be studied considering both the smart survey features and the hybrid nature of data (traditional data, internal

or external sensors, public or personal online data). Indeed, the severity of the errors depends on the specific data sources, on the type of sensor and analytic goals involved.
In this chapter, we analyse more deeply representation errors and sensor measurement errors starting from two reference frameworks: the Total Survey Error (TSE) for traditional surveys and the Total Error (TE) defined for Big Data (Biemer et al., 2017; ESSnet Big Data II – Work Package K – Methodology and quality).

5.1 Reference Frameworks for Total Smart Survey Error

Total Survey Error framework for traditional surveys

TSE for traditional surveys provides a conceptual structure to identify, describe, and quantify the errors of survey estimates (Biemer et al, 2017). For (traditional) survey dataset, TSE may be expressed by the sum of row error, column error and cell error. In the table below they are summarized in row, column and cell errors.

Table 6. Total error for survey dataset

Row errors	Omission, duplications, erroneous inclusions	Representation errors
Column errors	Labelling/metadata errors, specification error, errors in reports or data capture, coding and processing error	Measurement errors
Cell errors	Content error, specification error, missing or incomplete data A content error occurs when the value in a cell satisfies the column definition but is still erroneous. Content errors may be the result of a measurement error, a data-processing error (e.g., keying, coding, editing, etc.), an imputation error, or some other cause. A specification error is just as described for the column error but applied to a cell. Missing data or incomplete data, is just an empty cell that should be filled.	Measurement errors

The various sources of error can be grouped in two main categories: errors of representation and errors of measurement.

Errors of representation include:

- Coverage error is a source of survey error due to systematic exclusion of sections of the population of inference.
- Non-response or participation error is a source of survey error arising when systematic sections of the population do not participate in a survey.

Errors of measurement include:

- Specification error occurs if the concepts measured by the data or method do not coincide with the concepts of interest for research purposes. Specification error can for example be due to categories of items that are not captured by the nature of the data or method, or due to errors in the unit about which data are collected (e.g. individuals or households).
- Missing or duplicate items or episodes arise when some items are 'missed' or 'duplicated'.
- Errors in reports or data capture arise when information is incorrect or miscategorised due to human or technological errors.
- Coding and processing error arises in the process necessary to make the data useable for research purposes, for example by deriving indicators from unstructured large data sets.
- Error due to panel conditioning can arise when the use of the technology influences the behaviour of sampling units.

Total error for Big Data

In most practical situations, the traditional TSE framework is quite limited because it does not attempt to describe the sources of error in non-standardized data management and transformation processes. For these processes, the best approach is to attempt to evaluate the quality of the end product.

The steps involved in the processing phase to highlight different types of error are the Extract/Transform/Load (ETL) step and the analyse step. The first step includes several stages: the extracting stage, where data are harvested from their sources, parsed, validated, curated, and stored; the transforming stage, where data are translated, coded, recoded, aggregated/disaggregated, and/or edited; and the loading stage, where data are integrated and stored in the data warehouse. In the analyse step, data are converted to information through a process involving two stages. The first stage is the filtering /reduction stage, where unwanted features and content are deleted; features may be combined to produce new ones; and data elements may be thinned. The second stage is the computation/analysis/visualization stage, where data are analysed and/or presented for interpretation and information extraction. In table 7, we report the steps involved in the processing phase, features and type of errors.

Table 7. Sensor data processing steps and types of errors

Steps	Features	Types of errors
Data generation	• Similar to data collection errors in surveys • Data may be erroneous or missing • Data generating units may be self-selected • Metadata may be lacking or absent	• Low signal/noise ratio • Lost signals • Failure to capture • Metadata that are lacking, absent, or erroneous
Process: - Extract - Transform (Cleanse) - Load (Store)	• Similar to data processing stages in surveys • Includes creating or enhancing meta-data • Record matching • Variable coding, editing, data munging or scrubbing, and data integration	• Specification error (including errors in metadata), matching error, coding error, editing error, data munging errors, and data integration errors
Analyse, visualization	• Data are filtered, or otherwise reduced. This may involve further transformations of the data	• Modelling errors, inadequate or erroneous treatment of missing, algorithmic errors • Selectivity • Errors in data visualization step

5.2 Representation (selection) Errors

The selection of respondents and thus the representation errors are at first a question of availability, because the assumption that everyone has and uses a smartphone all the time is merely a myth. For example, coverage errors occur due to the “Digital Divide” or the “Second Digital Divide.” Digital Divide means the separation of those who have access to computers and digital communication technology and those who have no access. The first Digital Divide is accessible by understanding the smartphone penetration in a population of interest. What is harder to track is the Second Digital Divide, which is the separation of those who have the skills to benefit from the use of computers and those who lack that skill. ICT skills and digital

literacy are not evenly distributed in a society or a sample population and provide a burden for the access to and acceptance of smart surveys.

Beside socially induced representation errors, also technical ones exist that are due to differences of sensor availability, utilization of different sensors by smartphone generations and thus also by price category. Not all respondents will have the latest “flagship” models with the latest sensors on board. Also covering all possible variations of smartphone generations is tedious and forces the researcher to decide for which potential population a smart survey is developed now and maintained for what time period. The operating system of a device can also make a difference by being more or less restrictive by default. For example, iOS is known to be more restrictive than Android. A combination of social and technical barriers is in place, which could also be due to different user settings for sensor access that can limit the passive data collection of a smart survey.

In smart surveys that require respondents to install a survey app on their phones or use some other mobile-only features (positioning systems, Bluetooth-enabled sensors, etc.), the coverage error is determined by the fact that individuals without smartphones could never participate in such surveys (row errors). Some respondents might only use their smart devices to access the internet and have no computer or laptop. This group might have never participated in a web based survey, which often is not responsive enough on a smart device or demands cumbersome manual entries.

Furthermore, passive mobile data collection – that can be used to decrease measurement errors and reduce respondent burden – can be affected by problems of selectivity, representativeness and ethics in the collection of such data. Respondents have to be willing to provide access to sensor data or perform additional tasks (downloading apps, taking pictures, etc.). If respondents who are willing to engage in these tasks differ from non-willing smartphone users, results based on passively collected data might be biased.

Respondents need to be asked for consent for activate sensors, to store and to send data. Most mobile device sensors require consent by default. Exceptions are the various motion sensors that can be activated in Android without consent. However, even without the technical necessity to ask for consent, there are legal and ethical reasons why consent is imperative. Willingness to consent varies per type of sensor and depends on the context and purpose of the measurements. The more intrusive a sensor measurement is, the more respondents will refuse and the larger the potential damage of missing sensor data will be (Mussmann and Schouten, 2019 – ESSnet Mimod WP5-3).

Non-response can occur at many stages, not only from consent to participate, to downloading and installing an app or device, but also to use that app (whether actively or passively), to capture and transmit data, often repeatedly over a period of time. The question of non-response becomes more complex as the additional tasks that can be performed increase. These activities can vary in the degree of involvement of the participants, the level of burden, the sensitivity of the data collected, the technical requirements (e.g. battery usage or data transmission volume).

Other barriers relate to the respondents’ willingness to engage in specific activities. This may depend on the general willingness and motivation to participate in surveys, as well as reactions to specific characteristics of the requested activity, on time constraints, general concerns about confidentiality and privacy issues related to technology and specific concerns about sharing personal information. Additional non-response can occur when the app is working: participants must then remember to use it to provide what is required. This requires continued motivation and engagement. Insufficient battery power, storage limitations, and other technical limitations may also lead to missed events.

Identifying strategies that maximize participation rates in smart population-based surveys is certainly a goal to pursue. However, the possibility that a smart survey may not be able to reach parts of the population should be considered, making the sample of respondents not representative of the general population. Since nonresponse bias is a function of the differences between respondents and non-respondents, the question of which units respond is important. Non-response bias is actually a function of the correlations between the propensity to respond and the survey variable (Bethlehem, 2002). A higher response rate does not imply a reduction in bias as the correlation can change in a nonlinear fashion as response rates change. Minimizing the non-response bias is a more relevant outcome than merely non-response rates. Model-based indicators are needed to assess and reduce the risk of non-response bias. Not only indicators calculated at the survey level, as R-indicators for the analysis of representativeness of response, but also indicators estimated at the statistic (variable) level to take into account any information from the partially missing survey data. The first type of indicators can be monitored during data collection in order to direct effort to cases with lower response propensities, thereby reducing the variability among subgroup response rates, while the second type of indicators are needed if we are to begin developing data collection strategies that are aimed at producing the highest quality data.

Non-coverage due to unavailability of a smartphone or other mobile devices by the individuals selected in the sample remains a major problem, as this error can be the largest contributor to Total Survey Error. To counteract coverage errors, it may be necessary to consider in the survey designs other traditional data collection modes to intercept specific subpopulations not reachable through the tools of a smart survey.

5.3 Measurement Errors in Sensor Data

Measurement errors come up as anomalies within the data collection process. As such, they are at the core a classification problem of wanted data and erroneous data. To better understand how measurement errors occur, we have to understand what sensors do. They measure a physical quantity and convert it into a signal, which a human or machine can process. Thus, processing of signals is always the result of a categorization process that puts meaning in conceived patterns. This is especially problematic, because sensors are often not capable to measure directly what we are interested in, and we rely on patterns found in data. Our assumptions about these patterns can be wrong or, as already described before, aiming at the wrong entity and resulting in specification errors.

Missed or duplicates items are not just a result of fault readings, but can happen along the whole data collection and processing pipeline. Not synchronized time settings, programming errors, or physical damage to a sensor are a likely source for such errors. Moreover, a sensor should only be sensitive to the physical quantity it measures. A source of error is often introduced by temperature. In addition, the sensor itself should not be a source of error, by influencing the physical quantity it is supposed to measure. Systematic errors in sensor readings can be the result of improper calibration (or corrected by calibration if discovered). These systematic errors are called drift because the signal output changes independently of the measured physical quantity. If the signal deviates randomly over time, it is called noise. These types of errors are report or data capture errors and are not only technological, but also human introduced errors.

In the following table, we report a classification of the measurement errors in collecting sensor data with a brief description (Teh et al., 2020).

Table 8. Description of measurement errors in sensor data

Error type	Description

Outliers	*Outliers*, anomalies and spikes, are values that exceed thresholds or largely deviate from the normal behaviour provided by the model. Outliers are also known as *faults*, though faults also include other types of errors such as bias, drifts, noise, constant value, and stuck-at-zero.
Missing data	Incomplete data
Bias	*Bias* is a value that is shifted in comparison with the normal behaviour of a sensor. This type of error would usually require calibration to subtract the offset from the observed reading to get its true value.
Drift	*Drifts* are readings that deviate from its true value over time due to the degradation of sensing material
Noise	*Noise* is a type of fault, and they are small variations in the dataset. Noise is similar to uncertainty.
Constant value	*Constant values* are readings with a constant value over time, though it might belong to a normal range.
Uncertainty	*Uncertainty* is, "a parameter, associated with the result of a measurement, that characterizes the dispersion of the values that could reasonably be attributed to the measurement".
Stuck-at-zero	*Stuck-at-zero* refers to values that are constantly at zero over an extended period of time

Coding and processing errors arise in the process of making sense of the data. All other factors discussed above do influence this category, but the distinct feature of this error category is that the process itself is responsible for introducing errors, e.g. by using a classification algorithm, which is not suited for the data or by selecting weak parameters during optimization of said algorithm.

Any instrument that is used in the field affects it and can thus cause panel-conditioning errors. For example, mobile devices are not passive devices; they are used actively and in a constant human interaction setting and a device could be used more during a survey and thus influence media use patterns. Or if considered too battery heavy even used less, also resulting in survey specific bias. However, smart surveys aim at reducing measurement errors. Passive data collection shifts errors from mostly human flaws to technological ones. The tradeoff is that technological errors can be better dealt with systematically and automatically.

Errors that occur in sensor data must be detected or quantified, removed or corrected to improve the accuracy and precision of data. To do this, it will be important to identify the best methodologies for analysing sensor data, exploring the accuracy and precision of a measurement, calibrating measured data, and testing a prediction against measurements to determine their statistical significance.

SECTION 3 – SMART SURVEY PROCESSES

6. GSBPM Phases involved in TSS_u

The Generic Statistical Business Process Model (GSBPM) and the Quality Indicators for the Generic Statistical Business Process Model (GSBPM) - For Statistics derived from Surveys and Administrative Data Sources (Version 2.0, October 2017) respectively represent the reference model and a guide for quality management within the statistical business process for the development of a methodological framework for smart surveys.

The Generic Statistical Business Process Model (GSBPM) describes and defines the set of business processes needed to produce official statistics. It provides a standard framework and harmonised terminology to help statistical organisations to modernise their statistical production processes, as well as to share methods and components. The GSBPM can also be used for integrating data and metadata standards, as a template for process documentation, for harmonising statistical computing infrastructures, and to provide a framework for process quality assessment and improvement (GSBPM Version 5.1, January 2019).

The GSBPM comprises three levels: (the statistical business process, the eight phases of the statistical business process, the sub-processes within each phase) and recognises several overarching processes with a strong statistical component that apply throughout the eight phases. These overarching processes include quality management, metadata management and data management, process data management, knowledge management, provider management. In quality management, the definition of a series of quality indicators that should be implemented within the GSBPM sub-processes to prevent and monitor errors has a fundamental role.

The GSBPM identifies the possible steps in the statistical business process and the inter-dependencies, but It is not a rigid framework in which all steps must be followed in a strict order. For smart surveys, it is possible that some phases and sub-processes need to be revisited from multiple points of view.

Following the methodological scope (not technical scope) of the GSBPM design phase, we briefly describe the sub-processes most interested in a smart survey that may require changes. Smart technologies and smart data (sensor data) are innovations that need a new data collection channel, new processing tools and new skills for the development of methodologies different from those used in traditional surveys.

Most mobile data collection activities – with the exception of the administration of a web questionnaire – require a particular respondent's involvement in the data collection process: to download and install an app on their smartphone or tablet; to activate features on their device; to give data capture permissions (for GPS coordinates of the mobile device); to provide survey data using the built-in sensors (e.g., accelerometer, GPS, microphone, camera); to interact in a dynamic and continuous way with the personal devices, thereby combining self-initiated data with passively collected sensor data.

Mobile data collection activities have different technical demands, including how much battery power and storage capacity they require. Some activities are used intermittently, such as answering questions sent via text messaging, and others allow the collection of continuous data over a certain period (passive data). In this last case, an automation of the data collection phase is necessary. The required storage capacity also varies between tasks, for example taking photos for data collection requires more storage capacity, as photos need to be stored on the mobile device before they are sent, whereas other tasks require no additional storage capacity.

The data collection activities differ in the extent to which they potentially intrude on the respondent's privacy and in what they require from respondents, willingness to use them is likely to vary between tasks, but also between types of respondents. These activities impact differently in the data collection process and on data quality. For this and other reasons, it is necessary to redesign the data collection methodologies and define new monitoring activities.

At the design stage they must be defined: data collection tools (app, questionnaire for mobile); paradata, etc; strategies to prevent measurement errors; contact strategies – incentives and content of invitation letters (app information, procedures to ensure data confidentiality, levels of incentives, etc.) – to reduce non-response. The smart design phase is crucial not only for the choice of data collection methodologies, but also for the choice of methods in the data processing phase and the allocation of resources in sub-processes of the statistics production.

The different nature and characteristics of data that can be acquired through mobile devices imply significant changes in the process phase. The survey variables are the result of a mix of information acquired with traditional instruments (questionnaire, diary) that can be processed in a standard way and information acquired through sensor data which instead require new processes and methods for their transformation and validation.

Other phases of GSBPM are involved by these changes, for example the building phase is also extremely important to establish a connection between the collection instruments and the metadata system, to test production system and statistical business process.

When changes in data collection and data processing methodologies are introduced, it is necessary to evaluate the risks to introduce new types of errors that can affect the quality of statistical output. In this case usability testing, pilot surveys, and experiments are different ways to examine the consequences on the data quality, for example, to test whether and how new technology influences measurements or systems.

6.1 GSBPM Design Phase and Data Quality Dimension

As for traditional survey, in the design phase of a smart survey, it is important to consider the main components of the quality of statistical output – relevance, accuracy and reliability, timeliness and punctuality, coherence and comparability, accessibility and clarity – and the complex relationship between the resulting quality of the statistical output and some preparatory sub-processes of the GSBPM design phase. In fact, each quality component depends on many design sub-processes, and most design sub-processes influence several quality components. Working with the quality component accuracy in the design phase means highlighting possible errors, different causes and characteristics, and the strategies that can be adopted to eliminate, minimize, and reduce them.

Another important task for a smart survey is the redesign of the production of information about metadata, and paradata useful in the production processes to control the processes and to drive the production system and the management/survey. Paradata may be selected to improve process quality and efficiency or to adjust some processes, for example in the contact strategies to increase response with motivation and incentives for the respondents, or in the strategies useful to reduce response burden.

For secondary data generated by sensors owned by third parties, the 'fitness for use' of the data source for official statistics needs to be determined focusing on the metadata quality of the source, on the data quality of the input data, the timely and stable delivery of these sources, the integration with survey data. When a secondary data source is found suited for use, delivery agreements with the data provider need to be set up.

It is therefore important to design the processes so that documentation, metadata, and paradata are as much as possible automatically generated as by-products from the processes. The GSBPM is complemented with GSIM, the Generic Statistical Information Model, which will facilitate communication, and with the Common Statistical Production Architecture (CSPA), useful to respond to the challenges of emerging information sources such as sensor data.

In the table below we describe the design phase of the GSBPM for a smart survey and the relationship between sub-processes and data quality dimension. Here, we do not consider sample design issues for dealing with the problems of coverage and self-selection.

Table 9. GSBPM design phase and relationship between sub-processes and data quality dimension

2. Design phase
2.1 Design outputs (accuracy and reliability)
2.2 Design variable descriptions (accuracy and reliability; coherence and comparability)
This sub-process defines: - the variables to be collected - the variables of interest measured through sensor data that will be derived from them
2.3 Design collection methodology (accuracy and reliability; timeless and punctuality)
This sub-process defines: - smart survey typology (Simple smart survey, Full smart survey, etc.) - collection instruments (questionnaire, diary, apps) - passive and active data collection - respondent consent and consent for other purposes (access to existing external personal data, linkage to external personal data already in possession of the survey institute) - privacy preservation, confidentiality protection, data and process governance - individualized incentives and effective communication - monitoring indicators - data quality indicators (completeness of the questionnaire and / or diary, interruptions in sensor data transmission)
2.4 Design frame and sample (coherence and comparability) – as in traditional surveys
2.5 Design statistical methodology for processing and analysis (coherence and comparability; accessibility and clarity)
This sub-process defines methods for treatment of structured (traditional) and unstructured data (sensor data) and indicators that provide a summary of the: • identified errors (missing, erroneous data), • type of corrections made (imputation), • quality of probabilistic and non-probabilistic record linkage methods in data integration process. For unstructured data, new sub-processes and methods must be used: - Tools and algorithms able to synthesize and transform the unstructured data into structured ones (this task can be performed both centrally and internally to the smart device prior to data transmission); - Methods/ ML techniques • to predict survey variable

- to solve probabilistic record linkage tasks in data integration sub-process to combine data from multiple sources
- for classifying input data to standard classification
- to automatically identify errors in the data, outlier values and miscoding errors
- to impute missing or incorrect data

6.2 GSBPM Collect Phase and Monitoring Indicators

When the collection has to be monitored/assessed, different dimensions have to be taken into account. GSBPM helps us to identify features of each phase to define proper measures to ensure good quality of collected data, if properly controlled. In GSBPM terms, three different steps for the collection phase are usually considered: set up, run collection, finalise collection, that in this case should be properly adapted to a smart survey approach.

In the setup step, the design is tested to minimize problems that could arise during the entire survey process. This step is aimed at ensuring that the people, processes and technologies are ready to collect data and metadata and, in the context of smart surveys, also paradata. In the context of smart surveys, the setup step consists of:

- preparing a collection strategy and tools that are differentiated for different type of data collection - i.e. device intelligence, internal/external sensors, etc.;
- ensuring collection resources are suitable - this can be translated in ensuring the collection tools, such as apps and web questionnaires, are available and work well with regard to acceptance and usability by the respondents;
- training collection staff - that is mostly employed to support respondents in managing the collection tools, such as apps installation, usability, etc.;
- configuring collection systems to request and receive the data;
- ensuring the security of data to be collected.

The second step that needs to be monitored is the run of the collection: indicators can be defined to be continuously run during the collection once it has started. This is usually necessary to keep under control every possible unexpected problem that needs to be promptly solved. During this step, all choices about survey collection are implemented, such as the different instruments to be used to collect and to achieve the full release of the requested information. It includes the initial contact with “providers” and any subsequent follow-up to obtain consent, and reminder actions.

In simple smart surveys, where the behaviour of respondents has to be “active”, the attention is to the participation of potential respondents, and to ensure the respondents to answer in the right way (to avoid non-sampling errors). The former actions can be thought of as being similar to the traditional survey context. In this view, among several controls, it is expected to record how respondents were contacted, and whether they have responded. This phase also includes the management of contacts, ensuring that the relationship between the statistical organisation and respondents remains positive, and recording and responding to comments, queries and complaints. This part can play an important role in the case the survey method is very innovative, since it can provide useful information to understand how the innovation has to be properly settled, for example, information about smart survey participation. When the behaviour of respondents has to be “passive”, i.e. only through sensors, the indicators are thought to identify mostly technical problems, such as connection problems, interruptions in the sensor data transmission, extraction of public or personal online data.

Some basic validation of the structure and integrity of the information received may take place within the run collection step, for example checking that files are in the right format and contain the expected fields. Nevertheless, the most of validation of the contents takes place in the process phases following the collection. At last, the finalise step has to be assessed. It includes loading the collected data, metadata and paradata into a suitable electronic environment for further processing. It may also include analysis of the process metadata and paradata associated with collection to ensure the collection activities have met requirements.

In the GSBPM collection phase, a set of quality indicators may be defined to help monitoring the different problems that might arise during data collection that can cause non-response errors, coverage errors and measurement errors, as reported in the following table:

Table 10. GSBPM collection phase - sub-processes and indicators for monitoring

4. Collection phase	Traditional and new data
4.2 Set up step	
Statistical Confidentiality and security	Risk of a breach while data is being transferred
Adequacy of resources	Rate of HR requirements fulfilled; the rate of IT requirements fulfilled
	Success rate for collection staff to perform collection tasks after having been trained
Soundness of implementation	Success rate for testing collection systems, under expected as well as high volume data and extreme situations
Timeliness and punctuality	Delay between expected and actual sign-off of collection systems (including data transmission, security, collection management systems, and quality control systems)
4.3 Run step	
Managing respondent burden	Are there enough staff responsible for dealing with the respondent's questions?
	Support is provided to respondents (e.g. toll-free number for content and technical questions)
Accuracy and reliability	Quality control for monitoring participation, the consent: - to download and install an app, - to use an app (whether actively or passively), - to give authorization to use sensors, - to transmit data, etc.
	Quality control is used to manage the quality of data collection and data capture processes - soft or hard checks of the plausibility of entered data and notifications of missing data (edits or checking rules)

	Domain response rates; representativity indicators; achieved CVs of key variables in domains of interest
	Unit nonresponse rate; item nonresponse rate; proxy rate
Timeliness and punctuality	Delay between expected and actual start and close of collection
4.4 Finalise step	
Cost-effectiveness	Discrepancy between planned versus actual collection costs Percentage of collection activities that met requirements (assessed through analysis of paradata)

6.3 GSBPM Process Phase for Sensor Data and ML Algorithms

Statistical methodology for processing smart data must be developed considering the sources of data and their features. The process phase must be seen as a composite phase in which for each type of data acquired – traditional, sensors by mobile devices, sensors owned by third parties, administrative, etc. - the sub-processes are declined differently both by providing new ones and by changing their sequence. For example, sensor data require a new sub-process with tools and algorithms able to synthesize and transform the unstructured data into structured ones.

Indeed, in a smart survey, unlike traditional ones, the data collected are often unstructured: the smart devices gather data such as text, video, photo audio, signals in general, and other kinds of data that can't be processed through traditional techniques. In the process phase, the ML techniques transform unstructured collected data in structured data, useful to produce statistics. Moreover, the ML techniques are very useful also to find hidden patterns in the data and to find relationships between the data collected. ML techniques allow us to build data-driven models. To build good models, we need a huge amount of data and the smart devices are the right tools to collect them: they can guarantee a continuous and very frequently collection of data.

In general, in this phase it is necessary to define quality indicators at different levels for synthesizing the identified errors in the data (missing, erroneous data), the type of corrections made (imputation), the data integration process (i.e. indicators to assess the quality of probabilistic record linkage methods), etc..

The Process phase in GSPBM (phase 5) framework is located between the Collection (GSBPM 4) and the Analyse (GSBPM 6) and it deals with data processing and describes their cleaning and preparation for analysis. This phase is composed of several sub-processes. Here, we analyse the following sub-processes considering link with ML algorithms. For each sub-process, we try to provide some suggestions on when and how ML techniques can be applied.

Data integration

The data integration sub-process (GSBPM 5.1) integrates data from one or more sources. It is where the results of sub-processes in the "Collect" phase are combined. The input data can be from a mixture of external or internal data sources, and a variety of collection modes, including extracts of administrative data.

The availability of data from different sources suggests the production of statistics through their combined use. In the smart surveys, the data collection uses multiple data sources as traditional questionnaires, administrative data, Big Data sources, apps, and so on. The availability of a large amount of information is an

opportunity that has to be exploited in the production of official statistics and this can be done through the integration of the different data sources available for our survey. The traditional techniques that allow integration of data coming from different sources are record linkage and statistical matching, as we well know.

In smart surveys, the data collection comes from different sources: the same questionnaire and related questions are submitted to respondents through different tools: web questionnaires and questionnaires in apps. In this case, the data integration is vertical and should be understood as adding records with the same format, but collected in different ways.

Now our question is: how ML techniques could help the data integration? One possible answer is related to the fact that the new and not traditional data sources, generally produce unstructured data. To extract information useful for statistical production from these unstructured data, it is necessary to apply ML techniques for their processing.

Moreover, ML techniques can be used in data integration phase to solve probabilistic record linkage tasks. It has been recognized that the classic algorithm for probabilistic record linkage is equivalent to the Naïve Bayes algorithm used in the machine learning field. With other ML techniques, as the single-layer perceptron, in conjunction with distributed technology, it is possible to achieve higher accuracy.

Classify and Coding

This sub-process (GSBPM 5.2) classifies and codes the input data. For example, automatic (or clerical) coding routines may assign numeric codes to text responses according to a pre-determined classification scheme.

Classification is a typical task of machine learning. There are many ML algorithms to classify data; the choice of algorithm heavily depends on data type (numbers, characters, images, etc.); the most common algorithms are SVM, decision tree, random forest etc. In traditional surveys the classification used is pre-determined. With unsupervised ML techniques, we could build new classifications starting from data only. This activity is another typical task of machine learning, known as clustering.

Classification tasks and clustering tasks are different: in the former a standard classification already exists and the task to be performed is to assign a class of belonging to each input data. Moreover, machine learning classification techniques are generally supervised techniques so that they require a dataset of examples of classification (labeled dataset) already carried out; in the latter task, instead, there is not pre-determined classification, but starting from the entire data set, the clusters are built through unsupervised ML techniques. One of the most common ML algorithms for clustering is K-means, in which a given anonymous data set (a set containing no information as to class identity) is split into a fixed number (k) of clusters. Initially, k number of so called centroids are chosen. A centroid is a data point (imaginary or real) at the center of a cluster. The process of classification and centroid adjustment is repeated until the values of the centroids stabilize.

Review and Validate

The sub-process review and validate (GSBPM 5.3) examines data to identify potential problems that can occur with the collected data such as errors, outliers, item non-response and miscoding. The review and validate phase together with the subsequent edit and imputation phase is very important because it improves the quality of the data and consequently the quality of the statistical models produced. The reviewing and validating can apply to data from any type of source, so in smart statistics it is needed too.

Machine learning techniques are very powerful and efficiently in finding hidden patterns in the data, and they can be applied in this stage to automatically identify errors in the data, outliers values and miscoding errors. Outlier detection can be handled both with supervised techniques, where you can train a model on a labeled dataset and unsupervised techniques, under the assumption that the majority of the instances in the data set are normal by looking for instances that seem to fit least to the remainder of the data set.

Edit and Imputation

In the edit and imputation sub-process (GSBPM 5.4), where data are considered incorrect, missing or unreliable, new values may be inserted in this sub-process. The terms editing and imputation cover a variety of methods to do this, often using a rule-based approach. Specific steps typically include:

- the determination of whether to add or change data;
- the selection of the method to be used;
- adding / changing data values;
- writing the new data values back to the data set, and flagging them as changed;
- the production of metadata on the editing and imputation process.

Machine learning techniques can be used as an alternative to the traditional imputation task. ML classification algorithms such as multilayer perceptron or random forest can be used to impute a missing univariate variable. The usefulness of these techniques has emerged in terms of accuracy on microdata and the quality of aggregated distributions. For the multivariate imputation models such as multivariate random forest and generative adversarial neural network (GANN) for imputation have been implemented with remarkable results.

Since imputation is usually improved when homogeneous units are grouped into imputation classes, ML methods similar to those used for stratification can be used to define these classes.

7. Automation of Smart Surveys

The vision in regard to the potential of Trusted Smart Surveys is ambitious, because “The final outcome of the methodological development should include a detailed and unambiguous description of the whole processing methodology, possibly in the form of a software program that can be executed fully automatically with no further manual intervention” (Ricciato et al. 2019, p. 593). This marks no less than another cultural change to fully transform the production process into a highly reproducible form based on a formal and machine read-able language for automatic procession of data. Furthermore, “[…] this trend pushes towards a full automatisation of the statistical production process, that must be necessarily encoded into binary code executable by machines (and not only into methodological handbooks targeted to human experts) as a prerequisite to be exported outside the SO. If the statistical methodology is encoded into a software program, we can clearly decouple the phase of methodological development (writing the source code) and production (executing the binary code). Such decoupling allows to export the physical computation (code execution) outside the SO, partially or in full, without giving away control over the methodology (code writing)” (Ricciato and Wirthmann 2019, pp. 2–3). This pushes to something that is coined as smart statistics, which determines the results of a statistical production process that is “[…] statistical production in real-time, incorporating functions of autonomous, automated and continuous data collection, data-driven processing, and adaptively responding to environmental changes.” (Vichi and Hand 2019, p. 606). Thus, again promoting a diverse team beyond a focus on methodology and domain expertise.

Due to the focus on smart devices, it is of particular interest what could and should be outsourced as reproducible processes in coded form. Two challenges are at hand, first, dealing with new data and thus new problems that are mostly handled with machine learning. And second, not to create a burden for the respondents or their smart devices in the process. Both challenges were already addressed above in detail. However, a smart way of dealing with these challenges was not addressed in particular. A first and important step to reach the overall goal is to automate all parts of a survey that deal with smart data. Lessons learned could then be applied to the overall process in a second step, which takes into account the various ways on how surveys are handled by a NSI.

In this chapter, we will discuss automation of analysis on a device and also take a glimpse at an automated statistical production process.

7.1 Analysis on Device

Analysis on device uses the computing power of the device to make sense of sensor data (e.g. what kind of activity is producing particular sensor readings) and aggregates it in a meaningful way (which also means less data traffic).[2] An important by-product of this process can also be enhanced privacy, because the level of detail is abstracted to a level, which is of interest for the survey only.

An example for this is a study by Kreuter at al. (2020), which used an app for a survey with passive data collection on a sample drawn from a panel population. Among other things, the app measured characteristics of the social network of the respondents by analysing the respondents' phonebook entries in regard to gender and nationality of the contacts. Information from outside services was brought onto the device, where this information was used to classify first names for gender and last names for nationality. As pointed out, information for the classification was pushed to the app and no personal information needed to be transferred to a third party. The data that left the app for the research were classification probabilities only. This example shows how expertise or tools from third parties can be used without exposing data.

A second example is a study by Malekzadeh et al. (2019), which addressed an inherent problem of raw data streams from motion sensors. Motion data discloses information on the weight or gender of a user and thus allow re-identification. [3] The goal of the study was to be able to use motion data for activity recognition in a privacy friendly way, by making it very hard to re-identify users. The suggested action is to transform sensor data before it is shared with specific applications that use it for monitoring purposes (e.g. tracking of daily activities). Thus, using several independent data processing steps.

Both examples show methodological challenges for data processing on a device. The first example shows that raw data collection was not necessary, because trained models can be used on the devices. The second example shows that additional measures might be taken into account, which change the data in a particular way, but still deliver the desired results. As a result, not only are new ways of thinking required to gather data passively, but also acceptance to lose a bit of control over the data during pre-processing and processing on a device. You cannot always evaluate the performance of a trained model in the field and might have no chance to go back to raw data and verify results. Analysis on device is a trade-off in many ways.

[2] An account on how to build (and improve) such models was made by Concone et al. (2017).

[3] The data set, which was used fort he study, is also availabe on kaggle: https://www.kaggle.com/malekzadeh/motionsense-dataset [2020/11/10]

The following lists include activities or information that could and should be processed on device and, if needed, transformed to make re-identification of respondents harder and data collection less invasive.

- Common activity tracking that is processed on smart devices or wearables includes
  - still or sitting
  - walking
  - running
  - vehicle use (car, bicycle, etc.)
  - use of stairs (up or down)
- Combined with time of day and position this also includes
  - commuting (from and to work)
  - places for leisure time (e.g. restaurant, gyms, private and public places)
  - places of family (household) vs. places of friends
- Further information to be derived include
  - information on gender
  - information on height
  - information on weight
  - health status (based on activities or health oriented sensors)
  - information on socio-economic status
- Combined with monitoring of the smart device itself
  - general level of activity
  - media use
  - social network

7.2 Automated Statistical Production Process

Rethinking the whole statistical production process in an automated way almost ready for real-time statistics, is already coined by the term smart statistics. The benefit of automation is that "[w]ith more and more (new, big) data available out there, statisticians can be partially relieved from the burden of collecting raw input data and can focus increasingly more on distilling high-quality output information (final statistics) from the available data." (Ricciato and Wirthmann 2019, p. 3). One new data source could provide a multitude of information for different statistical domains. Thus, these data are multi-purpose sources and lead to multi-source statistics, which is means better quality for statistics and presumably faster production considering the different speeds and entities of data production.

The transition to an automated statistical production process – beyond smart data – needs to tackle other challenges as well. Quality of statistics often derives from experts' domain knowledge as well, which, for example, is used for error or plausibility checks on data or econometric models. These are all tasks that can be supported – if not replaced – in an automated process. Such automation does mean a cultural change in the production of official statistics and vast changes in infrastructure and staff profile needs. The tasks involved do not decrease, as the complexity of the process increases, but it requires other expertise. We highlighted some of the overreaching nature of smart surveys in the section on GSBPM Phases.

Neither is automation a goal in itself, nor does it mean to replace human staff with machines. It replaces the need for humans to do tedious work and offers the opportunity to introduce more quality checks along the production process of statistics. The higher the degree of automation grows the quicker descriptions of phenomena and crucial information reaches decision-makers. It also allows them to base their actions on facts quicker and removes layers of uncertainty. Hence, smart surveys can be understood as a large leap

towards a socio-technical ecosystem with less uncertainty in a more complex world. However, such a chance in the way of production of statistics does not go without a change in infrastructure of hardware, software and human ware. In the next chapter we will have a look at the infrastructure side and shed some light on the actual requirements.

8. Infrastructure and Staff Profiles Needs

8.1 Infrastructure

There is no one size fits all solution when it comes to infrastructure needs, because individual needs of NSIs are different, organisational requirements for infrastructure planning and deployment could set boundaries or particular technologies are not available. Therefore, we suggest core functions that an infrastructure for trusted smart surveys should satisfy. The guiding principle of the infrastructure is the automation of smart surveys and in particular the automation of smart survey parts that deal with smart data.

A typical infrastructure would include a development layer (for analytics), a metadata layer (for statistical metadata, version control etc.) and an execution or deployment layer (for data processing or for the deployment of the preliminary or final statistical product). The development layer is the place where the coding for the machine learning models, analytics or ad hoc analytics is done. This layer includes one or more analytical tools that are commonly used at an NSI. The metadata layer keeps control of the different states of data and the analytical process itself. Typically scheduling tools and version control systems are part of such a layer. The deployment layer includes the bits and pieces that are triggered by the metadata layer. Results are piped back to the metadata layer and are available for future processes. Such an infrastructure that is also heavy on machine learning is described as ModelOPS or MLOps, because it transformed the DevOps approach from software engineering to fit in analytics environments. In case of ModelOPS it means to close the gap between data science and IT infrastructure teams. It aims in particular at improving machine-learning models. However, it is also applicable on any automated analytics process. The entire infrastructure must rely on infrastructure as code to be reproducible, scalable and suitable for automation in the first place. An infrastructure for smart data processing is an entry-level to automation or an opportunity to expand it. Automation depends on the overall architecture of the infrastructure, which is why it will be describe in more detail in a later chapter. In terms of methodology, we have to comprehend that the term smart does not only define the survey or the way data is collected but also the infrastructure behind it.

8.2 Staff Profiles

We have identified three distinct staff profiles that come in handy for Trusted Smart Surveys. First, a Smart Data Methodologist, who researches and develops surveys that utilize sensor data as an essential part of data collection. Thus, not just asking a question that a survey should answer, but also thinking of new and innovative ways to generate the data needed for reports. Second, a Data Engineer, who transforms smart data into a useful format for analysis by building data pipelines from the points of data collection to the analytical environment or even to the distribution and publication environment. Thus, providing the tools needed for a processing smart data. Third, a Data Scientist, who specializes on the analytical part with an emphasis on machine learning. Thus, finding patterns in data, providing (automated) reports and visualizations of findings.

Smart Data Methodologist

A Smart Data Methodologist is responsible for the design of a smart survey, data models and metadata models. An understanding of sensor-based (para) data and how sensors work (and fail) is essential for this

profile. A person with this profile can enhance a legacy survey with smart data collection, develop new smart data collection methods and integrate the data collection process into the business logic. The Smart Data Methodologist has the most in-depth knowledge of the statistical domain and in particular of the requirements of official statistics. This person acts as the main link between these requirements and the technical implementation of smart surveys. This also includes the communication of these requirements to team members, who are involved deeper on the technical level (Data Engineer) and the analytical level (Data Scientist).

A background in social science, statistics or computer science is a good start. A mandatory skill is the ability to programmatically deal with data, which allows for a better understanding of the data life cycle. An ideal profile would include experience with UX- or UI-design for the adaptation of smart surveys for smart devices with human interaction.

Data Engineer

A Data Engineer handles the heavy lifting of building architectures, pipelines (ETL or ELT) and databases that prepare data for analysis. Storing raw data in an organised way (e.g. data lakes), automatic cleaning and validation for storage in databases for Data Scientists or even full automatic pipelines to store cleaned and validated data in statistical data warehouse are part of this profile. In addition, automated publication of results could be build or assisted by a Data Engineer. A must are skills that cover databases, ranging from RDBMS (e.g. MySQL), NoSQL (e.g. PostgreSQL) to document-based databases (e.g. Mongo DB), and pipeline/ scheduling tools (e.g. Apache Airflow). Ideally, all data focused infrastructure is maintained by a data engineer to allow for auto-scaling for an efficient use of resources.

A background in computer science is preferred, but other data analysis backgrounds are also a good base to build upon. A thorough understanding of IT infrastructure (e.g. server-client-based infrastructures, container technologies) is necessary. Particular knowledge of UNIX based systems often welcomed. Programming skills (e.g. bash, Python or other languages that allow easy scripting) are required to implement pipelines.

Data Scientist

A Data Scientist covers the analytical part from EDA in case of new smart surveys or changes in the data collection process to fully automated reports with rich data storytelling. This person often uses advanced statistical methods and machine learning to get new insights from structured and unstructured data. Hence, a Data Scientist is also the person, who deals with all problematic (erroneous) parts of smart survey data.

A background in mathematics, computer science, statistics or social science is a good start, as this profile is a hybrid of all these disciplines. A strong analytical insight is the focus of this profile, which means the use of scientific methods to extract insight from data has a priority over mere technical knowledge. In an ideal situation a Data Scientist is working together with a technical versatile Data Engineer to handle the analytical infrastructure and to ensure that data is accessible for analysis. An ideal profile would include knowledge of programming languages with a strong data analytical part (e.g. Python, R, SAS), frameworks for machine learning (e.g. scikit-learn, tensorflow) and a how to use these skills with a cluster computing frameworks (e.g. Spark).

References

Ali, S.; Khusro, S.; Rauf, A.; Mahfooz S. (2014): Sensors and Mobile Phones: Evolution and State of the Art. Pakistan journal of science · December 2014

Bethlehem, J. G. (2002): Weighting Nonresponse Adjustments Based on Auxiliary Information. Survey Nonresponse. R. M. Groves. New York, Wiley.

Biemer, P. P.; de Leeuw E.; Eckman S.; Edwards B.; Kreuter T.; Lyberg L. E.; Tucker N. C.; West B. T. (Editors). (2017): Total Survey Error in Practice. John Wiley & Sons, Inc., Hoboken, New Jersey

Buskirk, Trent D.; Andres, Charles (2012): Smart Surveys for Smart Phones. Exploring Various Approaches for Conducting Online Mobile Surveys via Smartphones. In *Survey Practice* 5 (1). DOI: 10.29115/SP-2012-0001.

Concone, Federico; Gaglio, Salvatore; Lo Re, Giuseppe; Morana, Marco (2017): Smartphone Data Analysis for Human Activity Recognition. In Floriana Esposito, Roberto Basili, Stefano Ferilli, Francesca A. Lisi (Eds.): AI*IA 2017 Advances in Artificial Intelligence. Conference of the Italian Association for Artificial Intelligence. Cham: Springer International Publishing (10640), pp. 58–71.

Dillman, Don A.; Smyth, Jolene D.; Christian, Leah Melani (2014): Internet, Phone, Mail, and Mixed-Mode Surveys. The Tailored Design Method. 4.th ed. Hoboken: Wiley.

Géron, Aurélien (2017): Hands-On Machine Learning with Scikit-Learn & TensorFlow. Sebastopol: O'Reilly.

Keusch, Florian; Struminskaya, Bella; Antoun, Christopher; Couper, Mick P.; Kreuter, Frauke (2019): Willingness to Participate in Passive Mobile Data Collection. In *Public Opinion Quarterly* 83, pp. 210–235.

Kreuter, Frauke; Haas, Georg-Christoph; Keusch, Florian; Bähr, Sebastian; Trappmann, Mark (2020): Collecting Survey and Smartphone Sensor Data With an App. Opportunities and Challenges Around Privacy and Informed Consent. In *Social Science Computer Review* 38 (5), pp. 533–549. DOI: 10.1177/0894439318816389.

Malekzadeh, Mohammad; Clegg, Richard G.; Cavallaro, Andrea; Haddadi, Hamed (2019): Mobile sensor data anonymization. In Gowri Sankar Ramachandran, Jorge Ortiz (Eds.): IoTDI '19. Proceedings of the 2019 Internet of Things Design and Implementation : April 15-18, 2019, Montreal, QC, Canada. IoTDI '19. Montreal Quebec Canada. CPS-IoT Week. New York, New York: The Association for Computing Machinery, pp. 49–58.

Mayer-Schönberger, Viktor; Cukier, Kenneth (2013): Big Data. A Revolution that will transform how we live, work, and think. New York: Houghtin Mifflin Harcourt Publishing.

Minnen, Joeri; Sabbe, Kelly; Nagel, Elke; Lenuweit, Birgit (2020): Methodological and evaluation report Eurostat Grant number: 847218 BE 2018 TUS Belgium, April 2020

Mitchell, Tom M. (2013): "Machine Learning" - Mc GrawHill - Isbn 0-07-115467-1

Mussmann, Ole; Schouten, Barry (2019): Final methodological report discussing the use of mobile device sensors in ESS surveys. MIXED MODE DESIGNS FOR SOCIAL SURVEYS - MIMOD, WP5 deliverable. Edited by EUROSTAT.

Ricciato, Fabio; Wirthmann, Albrecht (2019): Trusted Smart Statistics. How new data will change official statistics. Edited by EUROSTAT European Commission.

Ricciato, Fabio; Wirthmann, Albrecht; Giannakouris, Konstantinos; Reis, Fernando; Skaliotis, Michail (2019): Trusted smart statistics. Motivations and principles. In *Statistical Journal of the IAOS* 35, pp. 589–603.

Salganik, Matthew J. (2018): Bit by Bit. Social Research in the Digital Age. Princeton: Princeton University Press.

Scannapieco, Monica; Bogdanovits, Frederik; Gallois, Frederic; Fischer, Bernhard; Georgiev, Kostadin; Paulussen, Remco et al. (2019): BREAL. Big Data Reference Architecture and Layers. Version 2019-12-09. Edited by EUROSTAT.

Teh Hui Yie; Kempa-Liehr, Andreas W.; I-Kai Wang, Kevin (2020): Sensor data quality: a systematic review. Journal of Big Data. https://doi.org/10.1186/s40537-020-0285-1

Vaccari, Carlo (2016): Big Data and Official Statistics. Saarbrücken: LAP LAMBERT.

Vichi, Maurizio; Hand, David J. (2019): Trusted smart statistics. The challenge of extracting usable aggregate information from new data sources. In *Statistical Journal of the IAOS* 35 (4), pp. 605–613.

Wenz, Alexander; Jäckle, Annette; Couper, Mick P. (2019): Willingness to use mobile technologies for data collection in a probability household panel. In *Survey Research Methods* 13 (1), pp. 1–22.

Task 3.1.2:

Technical infrastructure

Version: February 2021

Prepared by:

Jacek Maślankowski (Statistics Poland, Poland)

Task leader:

Jacek Maślankowski (Statistics Poland)

Outline

Executive summary

In recent years we could observe the rapid development of mobile devices, including smartphones, smart watches, smart bands and other, less popular devices. Currently, almost every type of mobile phone is equipped with GPS, accelerometer or gyroscope. Such sensors allow software producers to create more flexible applications, e.g., depending whether the mobile phone is in vertical or horizontal position. GPS originally was used to find the current position on map, but it was adopted to many new different applications nowadays. One of them is to track the position of the mobile phone owner, to count the distance walked or driven. The presence of different sensors also allows to use them to support traditional statistical surveys. For example, crossing the border between countries can be monitored this way. Another application is to use photos to gather information on products bought or food consumed. Dedicated supervised machine learning algorithms can be used to detect, respectively, goods or meals.

In this chapter an architecture of two different smart surveys platform was presented, i.e., HBS (Household Budged Survey) and MOTUS (Modular Online Time Use Survey). Its technical infrastructure analysis was also used to create a generic infrastructure for smart surveys. In this chapter there is also a list of requirements that should be fulfilled by the smart survey application. It is important to note that these two smart surveys applications are still under development.

The aim of this chapter is on the analysis of the as-is architecture of HBS and MOTUS platform, as of 2020. The current work done is a preliminary step to develop a to-be architecture in the next phase of smart surveys maturity. The next step would be to develop a Proof-of-Concept for the smart surveys.

This chapter was divided into three parts. In the first part a general information on possible ways of data collection through the mobile devices was presented. The second part shows the requirement analysis for smart surveys, i.e., mobile applications used to collect statistical data. The third part shows the logical components of technical architecture according to the BREAL - Big Data REference Architecture and Layers Application and Information Architecture.

1. Data collection with the use of mobile phones and smart devices

1.1. Possible data available

Modern mobile devices are equipped with several different sensors, depending on its type (e.g., smartwatch, mobile phone) and the advanced technology used in the device. There is a variety of different sensors that can be used to provide information from mobile devices. It includes:

- Microphone,
- GPS (coordinates, altimeter),
- Accelerometer,
- Proximity Sensor,
- Light Sensor,
- Touchscreen sensors,
- Gyroscope,
- Magnetometer.

Some of the devices, especially wearable, can also include the following sensors:

- Heart Rate Sensor,
- Pedometer,
- Barometer.

All of aforementioned sensors is not available in most of the mobile devices currently offered in the market. Usually, less expensive devices have only few of them. However, in most cases smart surveys will not expect to monitor the activity of the mobile phone user with all sensors used at the same time. The issue is that in most cases, the use of all available sensors will lead to increased battery consumption and as a result, most of potential respondents will not use the smart survey application all the time.

1.2. Suggested structure of data

Because of the large amount of data generated automatically from mobile phone sensors, it is very likely to store them in the Big Data ecosystem. The current experience with smart surveys shows that for a limited number of users it is possible to record the data in the real-time in traditional databases, including SQL-like databases. For instance, HBS (Household Budget Survey) mobile application is forwarding most of the data directly to the PostgreSQL databases. However, at a large scale, using a centralized environment to store the data from various countries will led to the need of the use of Big Data environment.

The variety of data formats used for Big Data shows that the most common for preprocessed data is:

- JSON file (available in most of NoSQL databases formats),
- CSV file,
- XML file.

These data can be stored in data storage applications, such as NoSQL (typically for JSON files), SQL (with CSV files imported) databases as well as more advanced Apache Hadoop environment (for flat files).

As the data stored in databases may increase rapidly, it is highly recommended to consider the use of scalable large databases. It is especially important when providing text information that is also used by smart surveys application. The data provided in clear text will have to be processed in most advanced way, including the text mining rules, what makes the use of Big Data ecosystem more reliable.

1.3. Security issues

To be added later.

1.4. API structure for data communication

To be added later (REST API based on HBS).

1.5. Use Cases

1.5.1. MOTUS

The aim of the MOTUS application is to collect the data regarding the time use. For example, it is possible to use the mobile phones to detect the type of the place according to the current location of the mobile phone owner. Depending on the location, it is possible to identify whether it is a home (most of time spent during night), work/school (time spent during the day) or any other location identified with geolocation on maps. However, some data must be gathered in traditional way, e.g., what activity was done at home, e.g., reading books or watching TV etc.

Originally in MOTUS software there were three modules [Minnen et al., 2014]:

- the base module, only questions the 'typical' context information (what activity was done, where it took place or what transport mode was used, and with whom the activity was undertaken),
- the media module, with questions for every activity whether any media (smartphone, tablet, laptop, written media, …),
- the transport module, questions contextual information.

From technical point of view, MOTUS is divided into back-office and front-office parts. The back-office is designed for researchers. It is possible to design, collect and analyse the data. Front-office is used by respondents. They can use the MOTUS software to provide the data in time diaries, including primary, secondary activities as well as another context [User Guide of MOTUS, 2020].

1.5.2. HBS

HBS – Household Budget Survey – is used to collect the data on the money spent on different goods and services. In this context, a smart survey can be used to detect the product bought based on its picture taken by mobile phone camera. Another possibility is to use the photo of the bill to scan the product list with prices and store it in databases. Thus, the respondent will not have to provide the list manually, as it was done in the questionnaire of the traditional survey.

Currently, there are two modules of HBS smart survey application [HBS, 2020]:

- Frontend – mobile phone application,
- Backend – data storage and processing.

The tool collects the data in two different ways. One of them is to collect user input data. The second option is to collect the data based on sensors in the device, i.e., mobile phone.

The data management can be divided into three parts:

- Data collection
    - Mobile phone sensors
- Data communication
    - REST API with a frontend
- Data storage
    - PostgreSQL database

CBS App Backend is a REST API backend for smartphone apps used for primary data collection, targeting a preselected sample of respondents. Three main functions of smart surveys are covered by backend of this application:

- executed data collection for a defined reference period,
- exchange of data via REST API,
- store data in PostgreSQL database.

2. Requirements analysis for smart surveys

There are several different types of functionalities that are expected from the smart surveys. They can be divided into two parts:

- functional,
- non-functional.

The first part includes all requirements that are related to the user interaction. Non-functional part concerns the speed and user friendliness. Below there is a SysML requirements table.

#	Id	Name	Text	Owner
1	1.1.	Data storage	All data gathered should be stored in structured format in database or machine-readable files.	NSI
2	1.2.	Sensors	Application should gather most of the data from sensors available in devices.	Eurostat
3	1.3.	API	The application should transmit the data with standards, i.e. pass through JSON, SDMX etc.	Eurostat
4	1.4.	Devices	The smart survey should use the most common devices, such as mobile phones rather than computers.	Eurostat
5	1.5.	Platforms	The application should be available to all popular platforms of devices, e.g. Google Android, Apple iOS.	Eurostat
6	1.6.	Alternative versions	Respondents who are in the sample frame and does not have a device needed by smart surveys should have a possibility to use alternative format, e.g. paperback questionnaire.	Eurostat
7	2.1.	User interaction	Smart survey (e.g. mobile application) should gather as many data as possible without user interaction.	Eurostat
8	2.2.	Performance	The application should gather the data with minimum use of mobile phone/device resources, such as memory or processor.	Eurostat
9	2.3.	Security	The application should be secure, and no sensitive data should be accessible by third parties.	Eurostat

Table 1: SysML requirement analysis for smart surveys

Among the list included in Table 1 some of the requirements are not covered by the applications mentioned in this sub-chapter, i.e. MOTUS or HBS.

3. Technical requirements according to BREAL

3.1. General overview

The BREAL (BREAL: Big Data REference Architecture and Layers Application and Information Architecture) is related to the specification of technical components, structured by different phases of data processing (Figure 1).

In this Technical infrastructure description of the smart surveys, the BREAL was used to visualize the main functions and interfaces used in the applications used to collect, process and analyse the data. Not all phases from the BREAL were taken into account as the complex structure of the Smart Survey application does not include all the components that were incorporated into the BREAL framework.

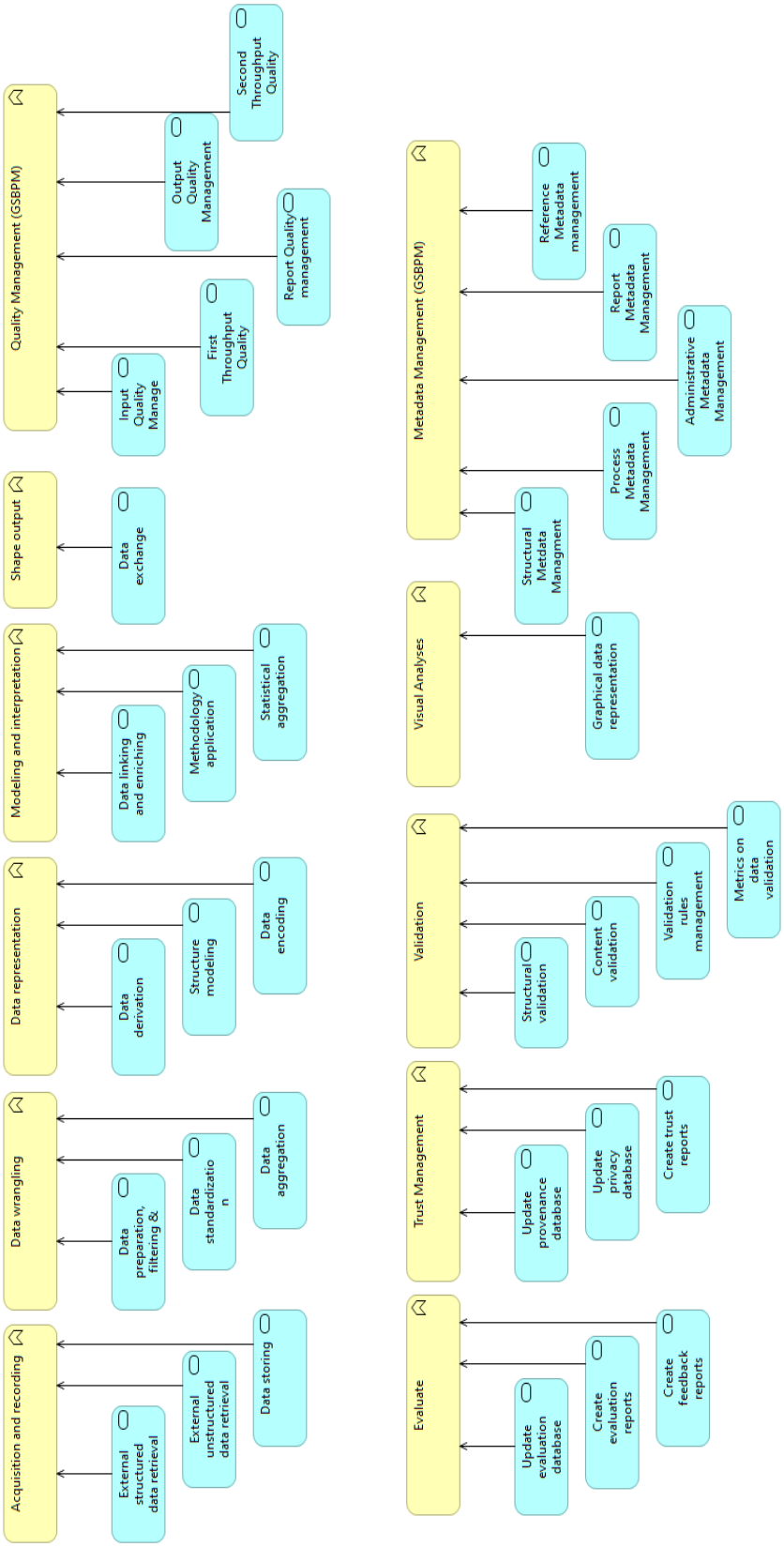

Figure 1: General overview of the BREAL architecture

Source: ESSnet Big Data II, Deliverable F2

With respect to the services described above, each of the function is represented with technical components specific for the smart surveys' applications. It includes all the input components written in previous subchapters as well as data bases used to store and validate the data.

3.2. Acquisition and Recording

The first service used for data acquisition and recording relies on the characteristics of smart surveys' application used. Thus, a mobile application which uses sensors as a main data source will gather structured data. A smart surveys' application supported by web interface to gather additional data will mostly store the data in unstructured form.

General overview of the technical components of the application on smart survey is presented in Figure 2.

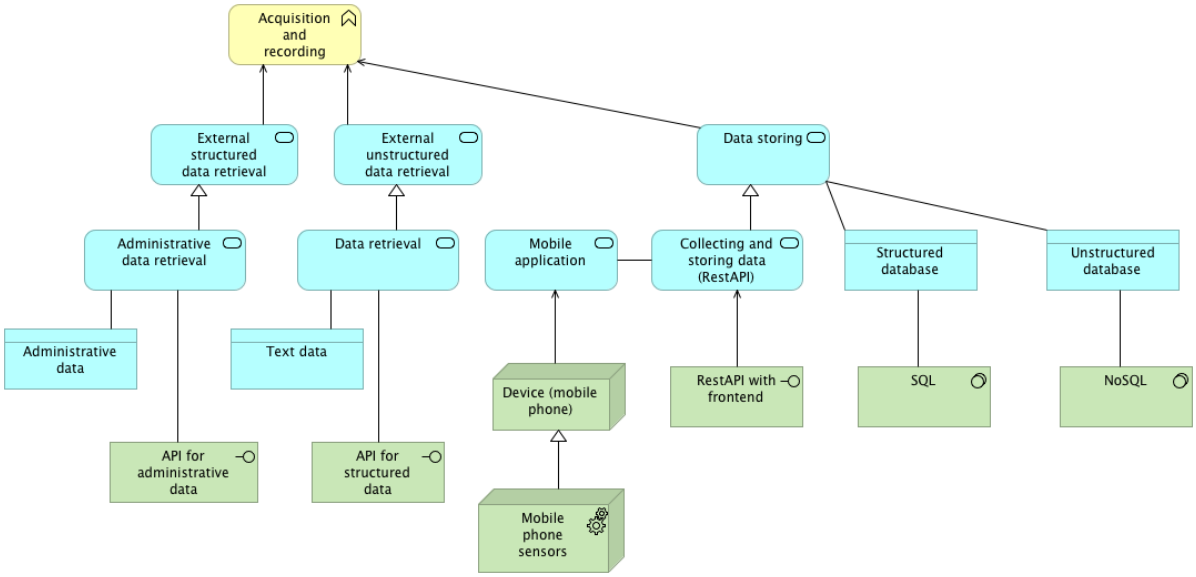

Figure 2: Acquisition and recording services for generic smart survey applications

In Figure 2, the yellow boxes visualize the business processes and functions. Blue boxes show external applications and services. The green ones refer to application components and services.

The generic smart survey application should consist of three services that include:

- external structured data retrieval,
- external unstructured data retrieval,
- internal data storing.

In this step it is important to ensure that the data will be available at NSI. It means that the data acquired from the mobile phone as well as other data sources should be available in internal data storage at NSI. In Figure 2 two possible ways of data storing are presented, i.e., structured and unstructured database. In particular cases, it is also possible to use flat files as data storage, including TXT, CSV and JSON, as mentioned in Chapter 3.1.2 of this document.

The first, external structured data retrieval should be related to the administrative data sources that will be available in most of the smart surveys' applications. It includes information about households, addresses etc. The suggested approach of connecting to external structured data sources is the use of dedicated API, if possible. In most of the modern and accredited data sources, it is possible to get the data through the dedicated API.

The second, external unstructured data retrieval is mostly related to the retrieve of text data – it may refer to the list of products and categories to recognize a specific input by user in passive mode. This mode is used when it is not possible to get all the information automatically through mobile phone sensors.

The data storing services is suggested to be integrated with internal mobile phone application, i.e., the dedicated application for smart surveys. The key component of the data storing services is the mobile application that can be used to acquire the data from mobile phone sensors. Then the internal API can be used to upload the data from sensors to the data storage, including SQL-like and NoSQL databases. Depending on the amount of the data, it is more likely to use NoSQL databases to store large amount of the data acquired through the mobile phones.

The first use case concerns HBS (Household Budget Survey) mobile application to take and process the information taken from mobile phone sensors (Figure 3).

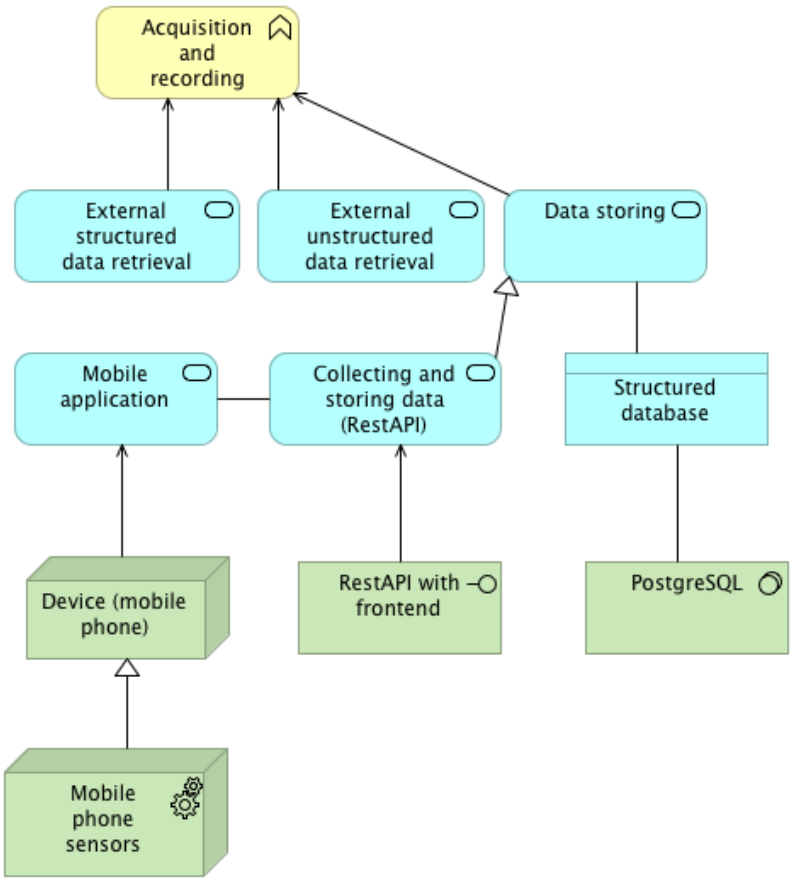

Figure 3: Acquisition and recording services for HBS application

The data comes from mobile phone sensors and is directed via the mobile application to the structured database, which is a structured database – in this use case PostgreSQL.

According to Figure 3, the data in HBS application is collected through sensors available in the mobile phone. The data is stored in the structured database, in this use case it is PostgreSQL database. The communication channel between the data retrieval and data storage components is a RestAPI with frontend application to manage the data obtained from the mobile device.

Currently, no structured and unstructured data retrieval components for external data sources are available in the HBS application.

The next use case concerns the MOTUS application. As mentioned in previous subchapters, MOTUS application has several different types of the sources used to gather data on survey' participants (Figure 4).

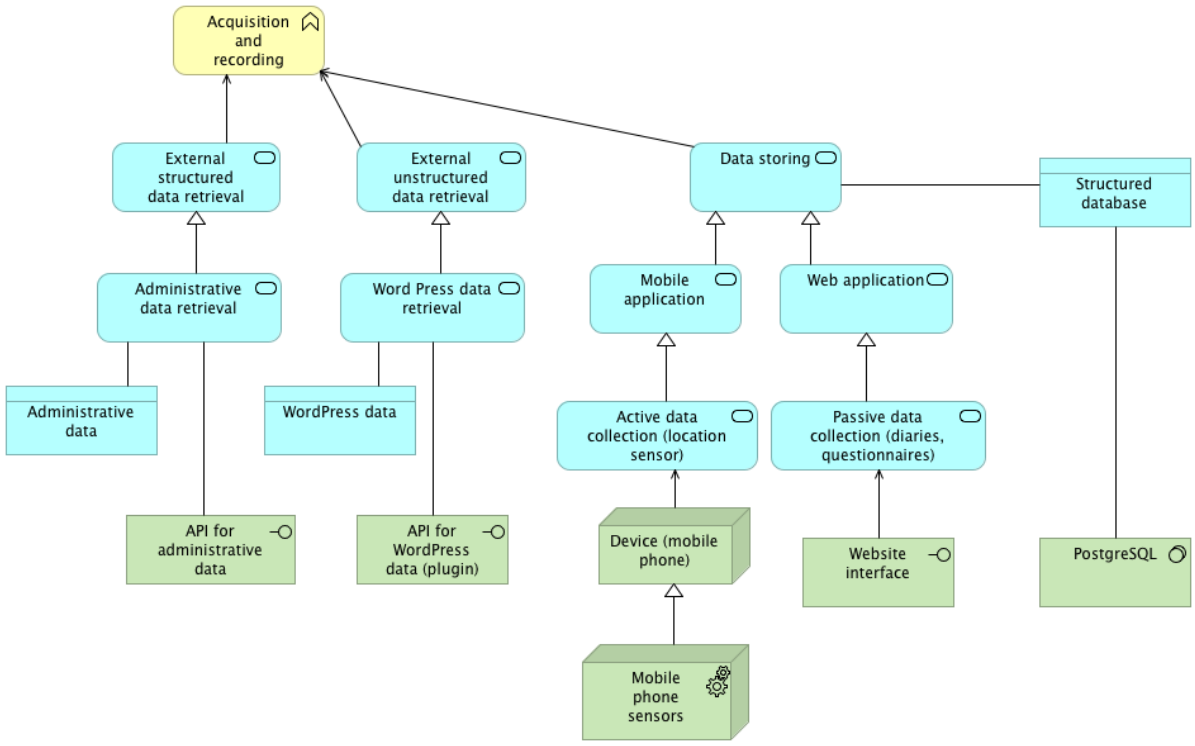

Figure 4: Acquisition and recording services for MOTUS application

First of all, MOTUS is based on the traditional questionnaire used to get the data on the activities taken by survey' participants. The data is collected via interface that is available on the web (laptop or desktop), as a mobile application (smartphone or tablet) or via text questionnaire (written media). All information is stored in the structured database. Mobile phones are also used to collect location of the respondent.

Thus, there are four different data sources that can be used to acquire the data by MOTUS application. The first includes administrative data retrieval component through external structured data retrieval service. Due to the opportunity of data collection from external administrative data sources via API, it is highly recommended to use the interface for API for administrative data if possible.

The next logical component includes external unstructured data retrieval through the API for WordPress data plugin.

The most important part of the MOTUS software consists of mobile application used for active data collection. The location sensor is used to get the current position of the mobile phone user. It includes the use of one of mobile phone sensors available in most of the devices currently present in the market.

Passive data collection with the possible interaction of the user is possible through the website interfaces. It includes diaries and questionnaires to be filled by the respondent. It is important to emphasise the fact that current responsive websites can fit to the type of the device, e.g., desktop computer vs. mobile phone, so it is not necessary to create a dedicated application for the mobile device.

All the data acquired through the MOTUS application is stored in the structured SQL-like database.

Some unstructured data collection is regarding the images used for product recognition in shops based on the receipt. In this part of data processing OCR techniques are used to prepare the list of products bought.

3.3. Data Wrangling

For smart surveys, the most important part is to get the data from the mobile phone and transform it to the most usable format. It includes the use of coding schemes to deliver the most appropriate and usable format for the data analysis. One of the possible aspects regarding the use of smart surveys is to collect the data by camera and transform it into structured dataset. It includes the recognition of receipt (with OCR) as well as the recognition of products (photo of the specific item bought). In Figure 5 there is a general approach of the picture recognition.

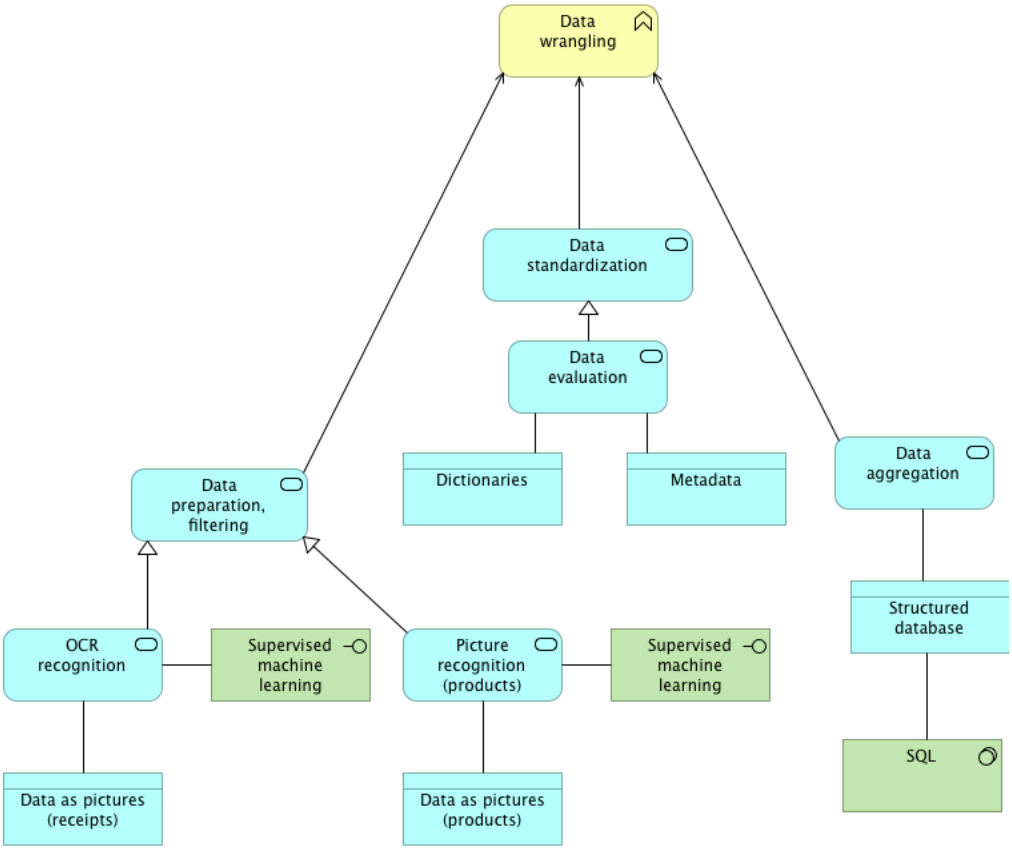

Figure 5: Data Wrangling for picture recognition

In Figure 5 the data shown considers only the analysis of the pictures. The data is transformed into structured form and written in database. The main logical component is the picture recognition with supervised machine learning. In this case, training dataset must be used to provide the output data with names / prices of the products bought. Because of the necessity of combining the output information with existing dictionaries, the final data should also be evaluated, e.g., if the list of products based on the receipt is acceptable or the names were not recognized properly. The final data should be recorded in the SQL database, as it is the most common data storage for smart surveys applications.

Such approach has been done for the HBS application. It is divided by two parts – the one is used to deal with OCR data, the second with the picture recognition.

3.4. Data Representation

Data representation concerns all the issues regarding data derivation, structure modelling as well as data encoding. In smart surveys it includes all the aspects specifically oriented to mobile devices. For example, one of the issues concerns data on geo-location based on mobile phone sensor responsible for this. The most common examples were presented in Figure 6.

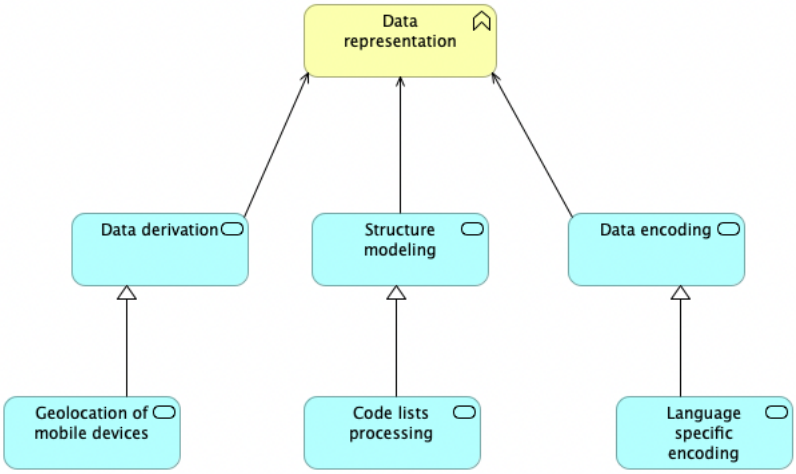

Figure 6: Geo-location – most common data representation activities with mobile phone

As presented in Figure 6, the data derivation concerns the proper identification of the data gathered from mobile devices and stored in the repositories, as it was shown in subchapter 3.2 (Acquisition and Recording). It means that raw data stored in the first phase of the BREAL framework is processed and transformed to the unified and understandable format. The second item is a structure modeling in which most of the final data are merged with the code lists, classification etc. to represent the surveyed entities. The last – data encoding is especially related to the language specific encoding, in which data can be transferred in various coding pages (e.g., UTF-8, cp1252) and may generate unexpected exceptions by the processing algorithm.

As shown in Figure 6, most of the aspects of data representation in smart surveys are strongly related to the nature of mobile devices. It shows that the wide use of different mobile devices in ESS countries can lead to the problems related to the reliable representation of the data. It means that several different services must be in use to ensure the right data encoding, structure modeling and data derivation.

3.5. Data Modeling and Interpretation

As it was mentioned in the previous sub-chapter, specific aspects of mobile phones have a direct influence on the BREAL services regarding the smart surveys. It means that the functions of geolocation or data encoding are specific to the mobile phone characteristics and its current settings. Thus, the services concerning data modelling and interpretation for smart surveys are very similar to the ones identified for the mobile phones. It is shown in the generic example for the mobile phones shown in Figure 7.

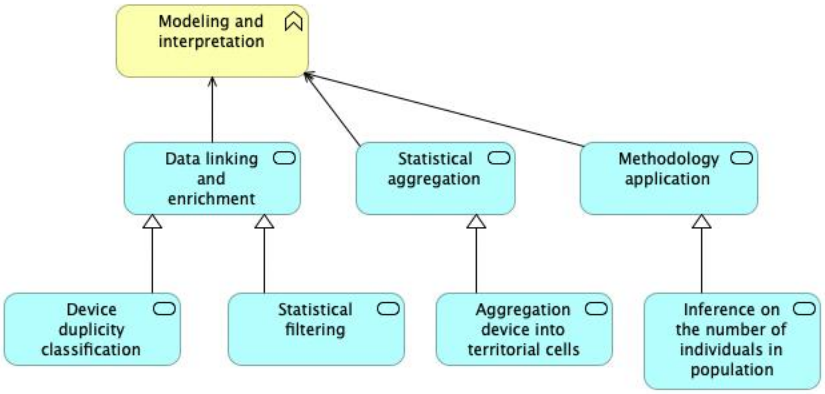

Figure 7: Data modelling and interpretation for mobile devices – a generic example

Source: Own elaboration based on (Deliv. I.6, 2020)

Modeling and Interpretation component is using three services, i.e., data linking and enrichment, statistical aggregation and methodology application. In the generic example for mobile devices presented in Figure 7, data linking and enrichment considers two services device duplicity classification and statistical filtering. In smart surveys the device duplicates are not very likely to occur. However, we should consider de-duplication framework in case the respondent will install the mobile application for smart surveys on two different devices and duplicate the results sent periodically. Statistical aggregation may concern all the possible attributes but in most cases it will be on the aggregation device into territorial cells, in which the survey is conducted. The methodology application regards the inference on the number of individuals in population.

However, taking into account more specific aspects of smart surveys dedicated to the different statistical domains, the use of this services may differ from the generic model shown in Figure 7. One of such examples is a MOTUS application. The specification of this application is the wide use of different communication channels (e.g., mobile phone, web browser) to provide the data on the Time Use.

In most cases, the smart surveys should use the most common methods and data formats, available in most applications. One of the examples is a MOTUS application. Its back-office functionality allows preparing the results data in several different formats, including machine readable formats compatible with Excel (csv, xls) as well as SPSS files. In Figure 5 there is generalized example of the Data representation for MOTUS back-office application.

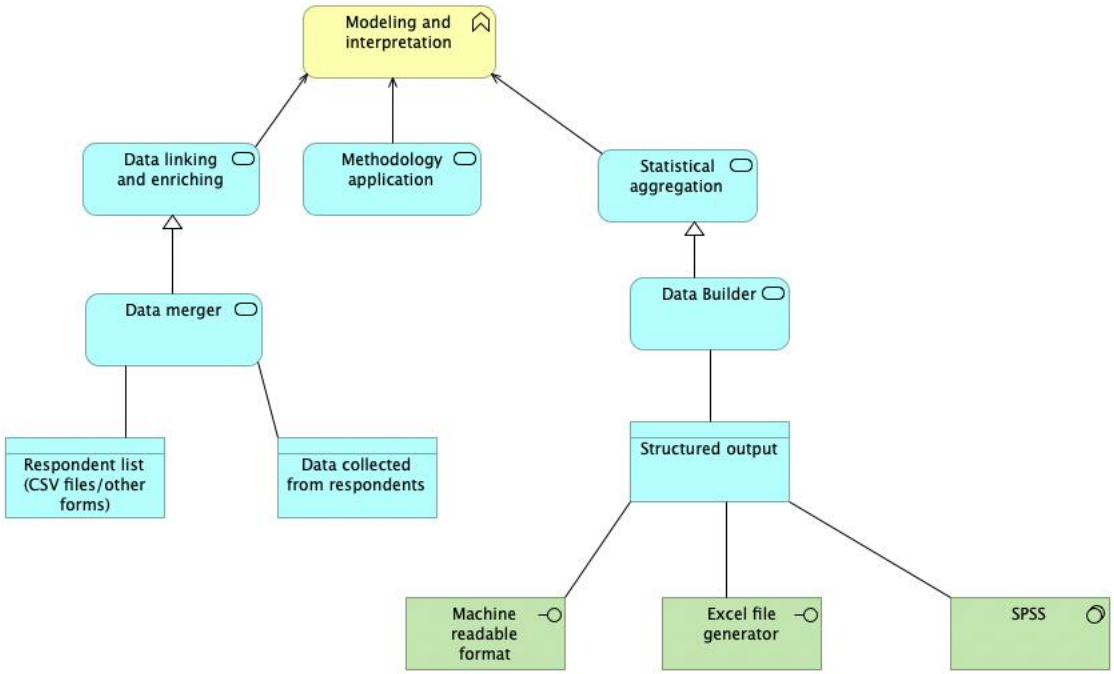

Figure 8: Data modelling and interpretation for MOTUS application – a selected functionality from back-office

In particular, smart surveys should deliver the data for analysis in the format that will make it possible to link all the data from the smart survey (i.e., machine generated data and human sourced information) with the data taken from administrative data sources or any other data source with information about sample. The most important aspect is to allow researchers to use the data. Considering the most common data formats it is hardly recommended to store the result data in the repository that allows to export the data in the most common formats.

As mentioned in the second section of this technical infrastructure description, the most recommended data repository for the data acquired directly from the mobile phone, including text and image data, is the unstructured databases. However, because of the common use of structured data repositories, it is suggested to use the interface to allow export the data into the most usable file formats.

4. Requirement analysis for

According to the requirement analysis in chapter 2, in table 2 we provided the basic features of two smart surveys described in this chapter.

#	Id	Name	MOTUS	HBS
1	1.1.	Data storage	Yes, relational database	Yes, PostgreSQL database
2	1.2.	Sensors	Yes, GPS	Yes, GPS, camera
3	1.3.	API	n/a	Yes, Rest API
4	1.4.	Devices	Mostly Web, mobile available	Mobile

#	Id	Name	MOTUS	HBS
5	1.5.	Platforms	All (web)	Android
6	1.6.	Alternative versions	Manual data collection through web	n/a
7	2.1.	User interaction	Most of the data	Most of the data
8	2.2.	Performance	n/a	n/a
9	2.3.	Security	SSL	SSL

Table 2: Main features of MOTUS/HBS according to the requirements analysis

It is important to understand that the features presented in the current documented versions of the applications is not the final list. We expect to improve the application to collect more data automatically without user interaction. Currently, this information was based on the documentation provided for these applications.

Next steps / Conclusions

The current applications of smart surveys allow to collect selected data without user interaction. However, most of the data must be collected in traditional way, i.e., by the respondent input. It shows that it is not possible to gather all information just based on sensors. Such data that cannot be gathered through the sensors includes, e.g., type of activity at home (e.g., reading books, watching TV, cooking) for Time Use Surveys. In Household Budget Survey it is necessary to write specific goods bought if there is no information on the bill or it is not possible to take the photo of the products. Moreover, sometimes the OCR to change the bill into text may be not sufficient to detect a specific product, as the name on the bill may be not precised (e.g., producer name instead of the product name).

The analysis of logical components used in the smart surveys shows that the main function of smart surveys is to support traditional surveys. It will not be possible to replace the traditional questionnaires with data based on sensors. Thus, the hybrid application that can collect selected data with sensors and other, missing data with electronic questionnaire, can be the model of the smart surveys in most of the statistical domains.

Thus, the main conclusion is that smart surveys application from technological point of view can play a complementary role to give more comprehensive and precise information on the user location in terms of time use survey. In household budget survey the main smart survey support is based on prices and products bought. In many cases, this information must be supplemented with additional information provided by the user manually in the mobile or web application.

In this meaning, the smart survey application uses the technology to collect all the data but in most cases the data will be collected with user interaction. Only part of this information will be collected automatically, including GPS location etc. However, both cases (i.e. MOTUS and HBS) will decrease the time needed to collect this information because of the facilities, for instance to get the information directly from the photo.

Finally, the chapter shows as-is architecture. The to-be architecture will be developed in the next steps of the ESSnet. Although some thoughts on the generic smart surveys' application were described in this chapter,

we expect in the future work more detailed information. It will include: main components, processes that use such components, an operational model (where the components are deployed (National, EU, etc.).

References

Minnen, Joeri & Glorieux, Ignace & van Tienoven, Theun Pieter & Daniels, Sarah & Weenas, Djiwo & Deyaert, Jef & Bogaert, Sarah & Rymenants, Sven. (2014). Modular Online Time Use Survey (MOTUS) – Translating an existing method to the 21st century. electronic International Journal for Time Use Research. 11. 73-93. 10.13085/eIJTUR.11.1.73-93.

[HBS, 2020] https://gitlab.com/tabi/hbs-budget-app/hbs_go_server/-/tree/legal/CSPA, as of 19th of May 2020

Deliverable F2, BREAL: Big Data REference Architecture and Layers Application and Information Architecture, ESSnet Big Data II, as of 20th of July 2020 (draft)

SOURCE™ Software Outreach and Redefinition to Collect E-data Through MOTUS. User Guide of MOTUS, 2020.

Deliverable I.6: Workpackage I, Mobile Network Data Deliverable I.6 (Quality). A Proposal for a Statistical Production Process with. Mobile Network Data, 2020

List of Figures

List of Tables

Task 3.1.3 - Integration in existing architectural framework

Version: February 2021

Prepared by:

Raffaella Maria Aracri (ISTAT, Italy)

Mauro Bruno (ISTAT, Italy)

Massimo De Cubellis (ISTAT, Italy)

Francesca Inglese (ISTAT, Italy)

Giuseppina Ruocco (ISTAT, Italy)

Task leader:

Mauro Bruno (ISTAT, Italy)

Outline

Executive summary

The main goal of task 3.1.3 is to provide an overview of reference standards and initiatives preceding TSS_u exploration. This analysis will foster the alignment of WP3 activities with existing architectural frameworks and will allow to benefit from the achievements of related experiences. The alignment with existing frameworks is also compliant with TSS_u driving principles addressing the privacy issue, the new methods for smart data processing, as well as process transparency, accountability and smart data life cycle.

Starting from an overview of official statistical standards and legal framework, the chapter focuses on the main results of the ESSNet Big data II, especially on the Big Data REference Architecture and Layers (BREAL). A brief description of relevant data platform projects complements the introductory analysis. Finally, the last section analyses the key elements to consider for modelling the business layer of a TSS_u platform. The activities of task 3.1.3 will deliver preparatory work to share and re-use smart survey solutions and components for the development of a European-wide platform. This platform will implement a set of common (horizontal) functions and configurable services, to build instances of TSS_u for specific domains and/or target areas.

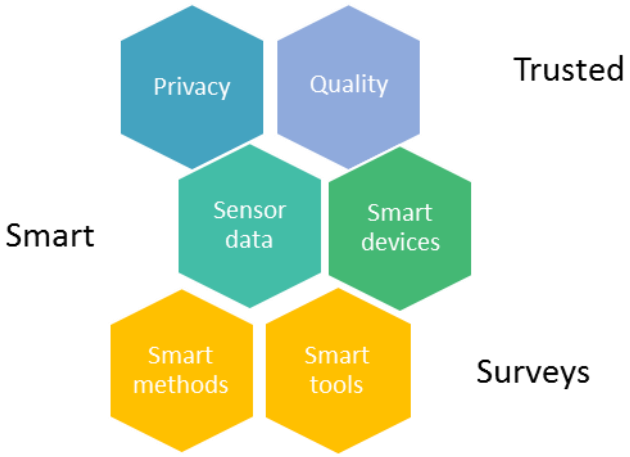

Figure 1: TSS_u driving principles

1. Reference standards and legal framework

This chapter provides an overview of the main official guidelines and recommendations of the reference framework of the European Statistical System. In addition, the second paragraph briefly describes the main Principles and Regulations that guarantee reliability of statistical processes and statistics, thus having a huge impact on the design principles of the platform.

1.1 Official statistical standards

The official statistical standards guiding the design and implementation activities of the TSS_u platform are:

- **Generic Activity Model for Statistical Organizations (GAMSO)**[4]. The GAMSO describes and defines the activities that take place within a typical organization producing official statistics. It extends and complements the Generic Statistical Business Process Model (GSBPM) by adding additional activities needed to support statistical production. Like the GSBPM, the GAMSO aims to provide a common vocabulary and framework to support international collaboration activities, particularly in the field of modernization.
- **Generic Statistical Business Process Model (GSBPM)**[5]**.** The GSBPM describes and defines the set of business processes needed to produce official statistics. It provides a standard framework and harmonized terminology to help statistical organizations to modernize their statistical production processes, as well as to share methods and components. The GSBPM can also be used for integrating data and metadata standards, as a template for process documentation for harmonizing statistical computing infrastructures, and as a framework for process quality assessment and improvement.
- **Generic Statistical Information Model (GSIM)**[6]. GSIM is a reference framework for statistical information, designed to play an important part in modernizing and streamlining official statistics at both national and international levels. It enables generic descriptions of the definition, management and use of data and metadata throughout the statistical production process. It provides a set of standardized, consistently described information objects, which are the inputs and outputs in the design and production of statistics.
- **Common Statistical Production Architecture (CSPA)**[7]**.** CSPA is a reference architecture for the statistical industry. The scope of CSPA is to support statistical production across the processes defined by the GSBPM (it does not characterize a full enterprise architecture for a statistical organization).
- **Common Statistical Data Architecture (CSDA)**[8]**.** CSDA is a reference architecture supporting statistical organizations in the design, integration, production and dissemination of official statistics based on both traditional and new types of data sources, such as Big Data, Scanner data, Web Scraping, etc.

[4] https://statswiki.unece.org/display/GAMSO/I.+Introduction

[5] https://statswiki.unece.org/display/GSBPM/I.+Introduction

[6] https://statswiki.unece.org/display/gsim/Generic+Statistical+Information+Model

[7] https://statswiki.unece.org/display/CSPA/Common+Statistical+Production+Architecture

[8] https://statswiki.unece.org/display/DA

- **Enterprise Architecture Reference Framework (EARF)[9].** ESS EARF supports the implementation of the ESS Vision 2020 as the guiding frame for ESS development up to 2020. ESS EARF is compliant with GSBPM, GAMSO, and CSPA.

1.2 Legal framework

In addition to the main statistical standards, the TSS_u platform implementation should be compliant with:

- The Principles and Laws that regulate European statistics;
- The Data protection principles, and more in general to the General Data Protection Regulation (GDPR)[10].

The set of UN Principles that guide the official statistical activity and support the NSIs in accomplishing their institutional role are:

- the Fundamental Principles of Official Statistics[11], initially adopted by the United Nations Statistical Commission in 1994, reaffirmed in 2013, and finally endorsed by the Economic and Social Council in its resolution 2013/21 of 24 July 2013.
- The Principles Governing International Statistical Activities[12], a set of principles and good practices recommended by the Chief Statisticians or coordinators of statistical activities of United Nations agencies and related organizations, to improve the functioning of the international statistical system. These principles have been endorsed by the Committee for the Coordination of Statistical Activities on 14 September 2005.

Built upon a common ESS definition of quality, the European Statistics Code of Practice[13] declares 16 principles that relate to the development, production and dissemination of European official statistics. Since 2011, the European Statistical System Committee has also adopted The Quality Assurance Framework[14] that provides guidance for implementing the Code of Practice principles. The following tables reports the subset of principles that refer to the statistical processes and output.

Statistical processes Principles	Statistical output Principles
Principle 7: Sound Methodology	Principle 11: Relevance
Principle 8: Appropriate Statistical Procedures	Principle 12: Accuracy and Reliability

[9] https://ec.europa.eu/eurostat/cros/content/ess-enterprise-architecture-reference-framework_en

[10] Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016

[11] The implementation guidelines of the are available from: https://unstats.un.org/unsd/dnss/gp/impguide.aspx

[12] Principles Governing International Statistical Activities are available from: http://hdr.undp.org/sites/default/files/principles_stat_activities.pdf

[13] European Statistics Code of Practice available from: https://ec.europa.eu/eurostat/web/quality/european-statistics-code-of-practice

[14] ESS Quality Assurance Framework, available from: https://ec.europa.eu/eurostat/documents/64157/4392716/ESS-QAF-V1-2final.pdf/bbf5970c-1adf-46c8-afc3-58ce177a0646

Principle 9: Non –excessive Burden on Respondents	Principle 13: Timeless and Punctuality
Principle 10: Cost effectiveness	Principle 14: Coherence and Comparability
	Principle 15: Accessibility

Table 3: ES CoP principles related to statistical processes and output

Among UNECE Regulations, the "Generic Law on Official Statistics" (GLOS)[15] is consistent with, both the Fundamental Principles of Official Statistics and the European statistics Code of Practice. GLOS aims at developing a solid legal basis for harmonising the activities of the National Statistical System within the countries of Eastern Europe, Caucasus and Central Asia and achieving high quality official statistics.

At European level, Regulation (EU) 2015/759 governs the European Statistical System, the activities and the role of National Statistical Institutes.

In addition to the legal framework, also the Data Protection assessment has a huge impact on the platform requirements. GDPR principles and rules must be applied throughout the statistical production process and relates to many aspects and perspectives such as: consent, security, accuracy, methodology, data storage. In brief, the modular implementations of the platform must be compliant with all the key data protection issues. Therefore, adopting the approach of privacy by design is one of the main requirements for TSS_u platform. From an architectural perspective, privacy preserving, and data protection are overarching dimensions affecting the requirements of each platform component. All the platform modules must be compliant with the privacy requirements related to different aspects. This may result in additional functionalities to be integrated in each module, or the design of a specific module for privacy preserving.

[15] Generic Law on Official Statistics available from: https://www.unece.org/index.php?id=45114

2. Preparatory projects for TSS_u

This chapter provides an overview of the results of several ESSNet research projects and other relevant initiatives, considered as starting point for designing the TSS_u platform. In addition, a description of ESSNet MIMOD - Mixed MOde Designs in Social Surveys is provided in Annex1.

2.1 Innovative tools and sources for living conditions surveys - HBS and TUS

This initiative, launched during 2011 DGINS in Wiesbaden and carried out by two Task forces, has the following overall objectives:

- Develop new and improving existing tools to reduce respondents burden (e.g. use of mobile apps, eDiaries, smart cards);
- Implement back ends to configure and manage surveys and web diaries, to increase efficiency of NSIs data collection systems;
- Link the data collection with new data sources and profit of the new technologies available.

Among the results, particularly relevant for this project is the test of a mobile application developed for the diary-based Household Budget Survey (HBS)[16]. The application extends the work done by Stat Austria adding three extra features related to smart data:

- receipt scanning and classification, for occasional expenditures of multiple products;
- geo-locations measurements to match the likely shopping places;
- sensor data linkage through consent, to enrich information and link respondent expenses and scanner data or banking data.

The results of this project provide high-level specifications for the data collection module of TSS_u platform. These requirements may relate to different components and aspects, such as:

- front-end
- back-end
- in-app features
- use of machine learning algorithms
- in-house data processing
- internationalization
- country-specific features
- consent management
- manage the combination of collection modes and devices

During this ESSnet the field test is extended to other countries, focusing on interviewer support in helping and motivating respondents to use an app, and the type of feedback to respondents, based on the information provided.

[16] For more details, see: Schouten B., Bulman J., Järvensivu M., Plate M., Vrabič-Kek B., Report on the action @HBS - Version 1

2.2 ESSNet Big data II

The ESSNet Big Data II has explored a wide range of smart devices features, with the aim to evaluate their potential in producing official statistics.

Starting from the preparatory work described in "Workpackage L - Preparing Smart Statistics"[17], this paragraph summarizes the main features of smart devices from different points of views. More precisely, the following table groups the different smart devices, considering the type of smart data, the data collection modalities (active or passive), the data collection tool implemented or embedded in the smart device, the frequency and the data-processing type.

With reference to the type of smart data, the following classification is particularly relevant for TSS_u:

- Sensor data on mobile device initiated by respondent (APPs for smartphone, tablet);
- Sensor data on external device(s) but collected by respondent (other smart devices than smartphone or tablet);
- External data not in possession of NSI where respondent may request access (through API), called data donation (Providers, Citizens Science Portals, Open data portals, etc.);
- External data already in possession of NSI where NSI may ask permission to link.

Moreover, the data collection modality is 'passive' when the respondent is not involved in terms of checking/supplementing/correcting the data collected and 'active' in all other cases.

Considering the data pre-processing, in some cases, the smart device may execute parts of data pre-processing ('In-app/devices'), or this task may be completely external to the smart device ('In-house').

Considering the frequency, the data collection can be realized through a data streaming or through a set of data that can be updated by the respondent.

The proposed classification is useful to start identifying the similarities between the different types of smart devices during data acquisition.

[17] Bosch O., Quaresma S., De Cubellis M., Workpackage L: Preparing Smart Statistics - Deliverable L3: Description of the findings regarding Task 3, Smart Devices. Final version, 30th October 2019

Type of smart data	Type of smart device	Type of data provision			
		Data collection (Active vs Passive)	**Data tool**	**Frequency**	**Data pre-processing**
				Data streaming or possible updates	**In-app/device vs in-house processing (paradata needed)**
Sensor data: mobile devices	Smartphone/Tablet	Active	APPs (questionnaire)	possible updates	In-app devices
Sensor data: external devices	Smartphone/Tablet	Passive	Sensors (accelerometer, gyroscope, magnetometer (compass), GPS, microphone, camera(s))	data streaming	In-app devices
	Wireless Speakers	Passive	Speakers	data streaming	In-app devices
	Smart TVs	Active	Media APPs (questionnaire)	possible updates	In-app devices
	Others Smart Home devices (Smart thermostats, Kitchen appiances, smoke sensors, alarm control units, etc)	Passive	Sensors	data streaming	In-app devices

Type of smart data	Type of smart device	Type of data provision			
		Data collection (Active vs Passive)	**Data tool**	**Frequency**	**Data pre-processing**
				Data streaming or possible updates	**In-app/device vs in-house processing (paradata needed)**
	Smart devices for Health (Devices that measure glucose levels, arterial pressure, heart rate, oxygen level, brain acitivity, blood alchool level, etc)	Passive	Sensors	data streaming	In-app devices
	Smart devices for Fitness (smart watches, smart wearables, etc)	Passive	Sensors	data streaming	In-app devices
Sensor data: external devices	Smart devices for Mobility (Smart Traffic Light, Smart detection cameras, Smart parking sensors. Smart passengers counter, Smart sensors in vehicles, Car, bike and E-scooter sharing systems, Portable GPS trackers)	Passive	Sensors	data streaming	In-app devices

Type of smart data	Type of smart device	Type of data provision			
		Data collection (Active vs Passive)	**Data tool**	**Frequency**	**Data pre-processing**
				Data streaming or possible updates	**In-app/device vs in-house processing (paradata needed)**
	Smart devices for travel (Smart travel cards, Tracking device, Smart suitcase, etc.)	Passive	Sensors	data streaming	In-app devices
External Providers smart data		Citizens Data donation - Passive	Depending on the agreement with the external Provider		
	Air quality systems	Citizens Data donation - Active	Open Data portals (es: Lufdaten project)	data streaming	In-app devices
	Smart devices for fitness	Citizens Data donation - Active	Open Data portals (es: Strava)	data streaming	In-app devices
NSI internal smart data collected for other purposes		Passive	Depending on the agreement with the internal Provider		

Table 4: Types of smart data, smart devices and data provision

2.3 ESSNet Big data II - Big data reference architecture (BREAL)

The ESSNet Big Data II has also analysed the architecture layers to process big data sources. The executive summary of the Deliverable[18] defines BREAL (Big Data REference Architecture and Layers) as:

"A European reference architecture for Big Data. BREAL serves the purpose of guiding Big Data investments by NSIs and helping the development of standardized solutions and shareable services within the ESS and beyond"

Intended users of BREAL are:

- NSIs that aim to introduce the use of Big Data in their production processes, especially those that plan to use Web data and sensor data.
- In addition to NSIs, public and private organizations that would like to follow a defined and controlled way of producing Big Data-based statistics guided by the Official Statistics expertise.

From a practical point of view, BREAL can be used as follows:

- As an instrument for NSIs top management to plan national investments related to Big Data projects, considering the economies of scale that are offered by European infrastructures and services for Big Data.
- As a 'reference framework' for enterprise architects to be used at national and ESS-level to align business and IT needs.
- As a 'language' for IT/solution architects to describe information systems projects that make use of Web data and sensor data. "

[18] Work Package F- Process and architecture. Deliverable F1: BREAL: Big Data REference Architecture and Layers

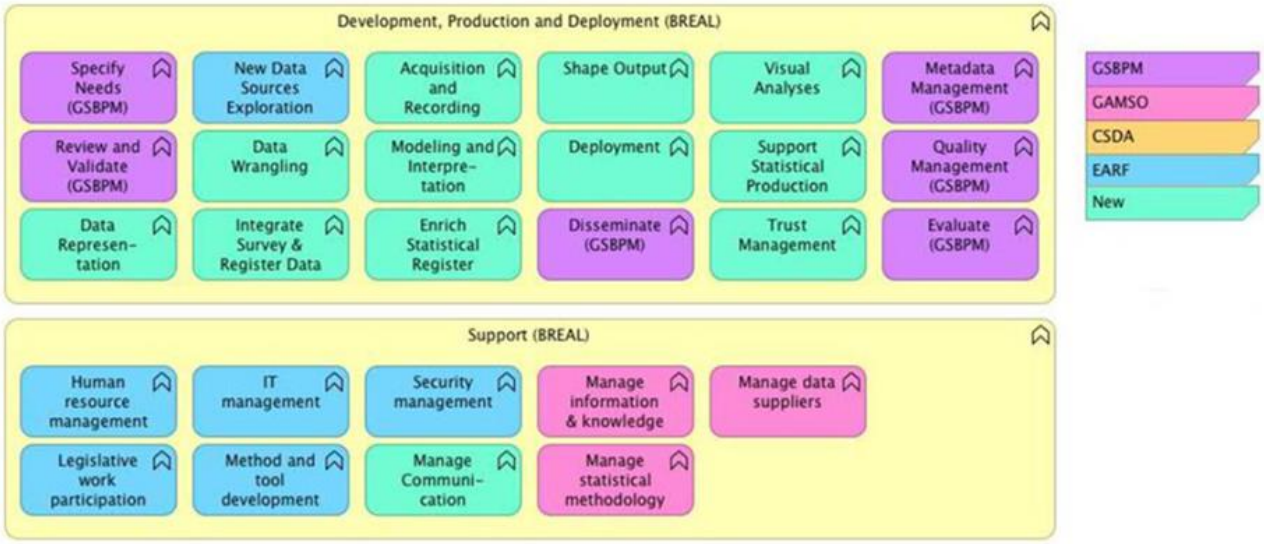

Figure 9- BREAL Business Functions

BREAL is also aligned with the following architectural frameworks:

- NIST Big Data Reference Architecture (NBDRA).
- Cross-industry standard process for data mining (CRISP-DM), as a reference of data mining process.

Adopting BREAL as general reference architecture has underlined the need to focus on specific business functions, to start modelling the platform layers. BREAL groups the business functions in two subsets: “Development, production and deployment” and “Support”. While the first group relates to the abilities to ingest, process and disseminate new data sources, the second subset includes the business functions that support the core activities related to the first group. Following an iterative implementation, the starting point for the design of the TSS_u architecture are the business functions reported below. In BREAL, the descriptions refer to big data, but in the following table, they have been adapted to smart data.

BREAL Business functions	Description
Development-Production and Deployment	
Acquisition and Recording	Ability to collect and store Smart data
Data Representation	Ability to model unstructured data
Data Wrangling	Ability to transform Smart data format for subsequent analysis and processing. It also includes Extraction (retrieving the data), Cleaning (detecting and correcting errors in the data) and Annotation (enriching with metadata).
Metadata Management	Ability to capture, store and manage the relevant information related to data and processes
Modelling and Interpretation	Ability to develop new algorithms for the statistical use of Smart data
Trust Management	Ability to gain reliable statistical products using Smart data sources
Support	

Manage data suppliers	Profiling and management of data suppliers, burden management
Manage statistical methodology	Use of standard statistical methods and practices
Method and tool development for new statistics	Ability to develop new methods and tools to support trusted Smart statistics

Table 5: BREAL Business functions and TSSu

To improve and increase the use of smart data for official statistics, the business function "Manage data suppliers" is of great relevance. The main aspects that affect the statistical process are:

- Management of data sharing agreements;
- Management of data transfer.

Depending on the type of data provider, the management of data sharing agreement may result, as an example, in protocols to apply for data ingestion. These protocols may establish the use of privacy preserving methods, or third parties computation techniques. Depending on the type of data source, the management of data transfer involves all the tasks related to data ingestion, such as data collection tool, software solutions, technology infrastructure.

2.4 Relevant data platforms projects

In addition to the BREAL reference architectures, the following initiatives are particularly relevant for TSS_u and their results should be considered in this analysis:

- Triangulum Platform;
- SOURCE™project, an initiative launched to promote the use of MOTUS platform in several countries. This platform is a key element of our AS-IS analysis. This experience is a relevant use case to analyse in terms of lessons learnt, best practices and challenges due to domain peculiarities.

2.4.1 Triangulum Platform[19]

The Triangulum project is one of the European Smart Cities and Communities Lighthouse Projects, funded by the European Commission through the Horizon 2020 research and innovation programme. The goal of the Triangulum project is to build replicable solutions and frameworks to acquire, process and disseminate big data from smart cities devices. Starting from the grant general requirements, the University of Stavanger (UiS) has developed and implemented two main components that provide cloud services for smart cities data. Although the cloud platform refers to a specific domain (smart cities), the following design principles and requirements are useful to identify some functionalities of the application components of the TSS_u platform.

Above all, the following principles have guided the modules implementation:

- Use of open-source and community-supported technologies.

[19] For an overview of the deliverable, see: https://www.triangulum-project.eu/wp-content/uploads/2019/03/2018-01_D2.2-Cloud-Data-Hub.pdf

- Modular implementation and instantiation (scalability of solutions).
- Agnostic in terms of data structures.

The whole architecture is built on the main direction of data flow. Particularly, the external data sources correspond to the upstream input of the cloud data platform, while processed data represent the downstream output to users or other software clients.

The two components implemented for the platform are: i) Data collection framework and ii) Data processing framework. The Data collection framework contains the following implementation units:

- Data acquisition;
- Data ingestion;
- Data storage;
- Data access.

As collected data can vary considerably, ad-hoc subcomponents called adaptors allow to deal with different data models, formats and unknown volume and velocity. Another type of subcomponents, namely queueing and load balancing, dispatch the data gathered by a specific adaptor to the more general subcomponents of the data collection framework. Concerning data layers, input data have three stages of preparation: raw, sanitized, and filtered. At the first stage, the data are considered raw, and after some pre-storage processing, they become sanitized and then filtered. Concerning data access, it is both internal to the platform, between the implemented units, and external. A specific webservice for registered users will be implemented for external access. Referring to data storage, the main requirement is to separate the storage of processed data from raw data.

The Data processing framework has the following subcomponents:

- Exploratory data analysis.
- Batch processing.
- Modelling.

Compliant with the general requirements, each software element is modular and implements lower levels of abstraction.

The following table describes shortly the main functionalities of each subcomponent.

Module	**Implemented Unit**	**Functionalities**
Data collection framework	Data acquisition	Acquire external data sources Transfer available data in the raw format
	Data ingestion	Ingest acquired data: load pre- storage row data and pre-storage data processing
	Data storage	Store ingested data in a common general framework

	Data access	Make the stored data accessible to the Data processing framework
Data processing framework	Exploratory data analysis	Data processing for exploration, manipulation, analyses, and calculations
	Batch processing	Process off-line large amount of data
	Modelling	Provides developed models to be applied to smaller dataset

Table 6: Main functionalities of platform subcomponents

2.4.2 SOURCE^{TM} project and Motus

The overall goal of the SOURCE^{TM} (Software Outreach and Redefinition to Collect E-data Through MOTUS) project is to describe and share the enhancements of the MOTUS platform[20] developed by the Research Group TOR (Tempus Omnia Revelat) since 2012. The specific objective of the project is to provide evidence of the effectiveness and feasibility of MOTUS solutions in the data collection framework of Statbel and Destatis. Testing MOTUS reuse will provide guidance for sharing and reusing MOTUS in the other European countries, within the main diary-based surveys: Time Use Survey (TUS) and Household Budget Survey (HBS).

Although the project has produced several work packages[21], in this context, the description of MOTUS platform aims at providing an overview of the functionalities implemented from an architectural perspective focused on the business layer. The idea is to benefit as much as possible of the MOTUS experience and results. For this purpose, the description of MOTUS in terms of CSPA layer is particularly relevant. The following analysis focuses on GSBPM phases covered by MOTUS and the correspondent GSIM input and output.

MOTUS is a platform implemented by the Research Group TOR of the Vrije Universiteit Brussel (Belgium). At the beginning, it was developed for collecting online diary-based information, for the Time Use Survey (TUS).

The platform has a front-office and a back-office. The front-office includes, both the mobile and the web application used by the respondents. While the operational logic of the application is embedded in the source code, the front-office is customized during the Design phase. The back-office, called the MOTUS builder, allows to define the content of the data collection application and supports the whole

[20] The platform is available at the following link: https://www.motusresearch.io/en/page/motus-research-platform

[21] For an overview of the project, as well as the content of the single deliverables, see the Methodological and evaluation report, SOURCE^{TM} Software Outreach and Redefinition to Collect E-data Through MOTUS - Towards a Modular Online Time Use Survey. April 2020.

statistical process, from data collection to data dissemination. The back end is composed by 11 modules, called “builders” that allow to run different surveys at the same time, even with the same respondent. MOTUS is based on HETUS-guidelines as reference framework.

To profit of the MOTUS experience, the MOTUS builders are described according to the related GSBPM phases. This analysis is useful to understand the specialization of each module and how the combination of different modules can be easily adjusted for different surveys. Although the MOTUS and TSS_u platforms have many similarities, the latter will be domain agnostic and provide scalable solutions to ingest different types of smart data. Therefore, it is useful to abstract as much as possible from the peculiarities of the survey for which MOTUS was conceived, to focus on the modularity approach. The study of modularity concerns not only the application layer, but the combination of the builders to meet the different requirements of GSBPM phases, or to adapt the initial set up to a specific survey.

The following table reports the builders involved in each GSBPM phase and their tasks.

GSBPM	MOTUS builder	Tasks
Design phase	Device builder	Management of Web app, Mobile app, API
	Questionnaire builder	Questionnaire definition and schedule
	Diary builder	Classification management Setting of diary parameters Definition of quality criteria
	Survey builder	Questionnaire definition (Individual, Household, custom for context peculiarities)
	Communication builder	Management of emails, notifications, letters
	Event builder	Management of plugins
Collect phase	Language builder	
	Research builder	Customize respondent tasks, event based flow to manage passive data, communication
	Invitation builder	Management of respondent participation
	Dashboard builder	
Analyse phase	Data builder	
	Quality builder	Definition of Para data/metrics

GSBPM	MOTUS builder	Tasks
	Computation builder	
Advise phase	Visualization builder	

Table 7: GSBPM phases and related MOTUS builders

The Computation builder and Visualization builder will be implemented in the future. Along with the Dashboard, the Data builder and the Quality builder, the Computation builder will be part of the subset of components for data processing. Once collected data are stored, the invocation of MOTUS R-package allows to clean data and is used by:

- Dashboard builder, for monitoring the fieldwork and assess the early results via an interface.
- Database builder, to download database and for metadata.
- Quality builder, for data quality measurement.
- Computation builder for data access via RStudio, and statistical methods for data processing.

Another outcome of the project is the assessment of ESS shareability of MOTUS, both for TUS and HBS. This analysis, based on the following business requirements, has underlined the need of further developments:

- Functionality & maintainability;
- Reusability;
- Online availability;
- Usability, user friendliness, accessibility;
- Data comparability.

Each requirement is related to key issues and dimensions. The table below describes these aspects according to the MOTUS experience and the elements tested for the reuse of the platform for HBS.

Business requirement	Description	Key issues	Dimensions
Functionality	Ability to perform data collection by improving data collection tools	Support the respondent	Ease of use Accessible Compatible with different devices, platforms, browsers Good performances Security
Maintainability	Ability to maintain and enhance the platform software	Choice of the programming language Track changes in the source code	Application architecture Deployment strategy
Reusability	Ability to reuse tools and software components	Intra reusability within a domain	Adjust easily the core strategy and the components setup for different contents and contexts

Business requirement	Description	Key issues	Dimensions
		Inter reusability within different domains	Implement one application compiled from the code base for each statistical domain, and a unique software environment to serve different statistical domains
		Combine input from different sources	Ingest passive data available in smart devices
Online availability	Ability to have a stable internet connection	Manage and harmonize web technologies and devices	Web and mobile application Devices and responsiveness Offline vs online registration Active and passive data connection
Usability, user friendliness, accessibility	A software easy to be used by all types of respondents	Implement app easy to use and to understand compliant with GDPR principles	Friendly User Interface Privacy management Respondents feedback
Data comparability	European Statistics Code of practice – Principle 14: "European Statistics are consistent internally, over time and comparable between regions and countries; it is possible to combine and make joint use of related data from different sources"	Process, tools, methods standardization Development of verification strategies that include Involve the respondent in the data validation strategy Training programs and best practices to improve data comparability	Sample design, sample handling and interaction with respondents Method comparability Use of the same application components Streamline data verification and cleaning Training courses & workshops

Table 8: Business requirements, Key issues and Dimensions for assessing MOTUS shareability

The privacy issues are one of the key elements for TUS and HBS, due to the respondent involvement is longer and more intensive compared to traditional surveys. According to the MOTUS experience, the assessment of the following activities would help to achieve GDPR compliance:

- Provide Information about the survey and additional use of the collected data;
- Describe the added value of each sensor;
- Ask for consent to use sensors in Smart devices and IoT devices;
- Adapt the consent to use sensors;
- Evaluate the relevance of the consent to use sensors as a prerequisite for Trusted Smart Surveys.

To benefit from the lessons learnt in the MOTUS experience, also the weaknesses and the threats underlined in the Swot analysis are particularly relevant. This analysis, reported in the following table, is useful to identify and prevent potential critical issue that may arise also during TSS_u implementation.

Internal weaknesses	External threats
Web app responsiveness	License strategy not yet defined
Individual-Household cluster	Governance model not yet defined
Not yet a training facility for researchers	System integration/harmonization
Not yet an informative website	National laws
Not yet a well-balanced guideline for respondents	External privacy issues
Not yet a well-balanced guideline for researchers	External security issues
No yet an online help-desk	External ethical issues
No yet an immediate feedback	Changes in the underlying software platform (Ionic, React, Flutter,…)
Growing gap between groups of individuals	Old, not updated devices
Steep learning curve	Old, not updated browsers
Burdensome registration	Stability and size of development team
Length of the participation	Server capacity
	Stress test not yet executed

Table 9: MOTUS Swot analysis

3. TSS_u platform business layer

This section describes some of the elements to consider, for modelling the business layer of the TSS_u platform. The analysis starts from the list of general requirements, and will focus on:

- BREAL business functions suitable for describing the business layer of TSS_u platform (e.g., Acquisition and Recording, Data Wrangling, Data Representation). The analysis aims at detailing also the main business processes that perform the business functions.
- Data life cycle, to track the main data transformations throughout the statistical process. The adoption of BREAL as reference architecture for modelling the business layer entails the alignment of the information layer also. The advantages resulting from the compliance with BREAL are several. Firstly, BREAL has embedded the hourglass model and for each business function, a set of application services allows to perform the associated process steps.

The platform design should fulfill the following constraints:

1. **General requirements stated in the objectives of the ESSNet**. TSS_u platform will be agnostic to specific survey domains and will have modular methodological frameworks to deal with the following general constraints:
    - Statistics with several data sources
    - Data sources relevant for several statistics
    - External data computation
    - Use of data without sharing

- Integration with existing frameworks

2. **Management of the Data Life Cycle of TSSu active and passive data**. In this context, an in-depth analysis of data life cycle is useful to understand similarities and differences between active and passive data. Dealing with a combination of these types of data may require ad-hoc solutions to preserve data integrity and assess data quality. Further, the new data sources can affect data modelling and for instance, to deal with structured and/or unstructured data may require specific methods and software solutions. The idea is to analyse each type of data according to the three data layers proposed by the hourglass model:

   - **Raw Data Layer**
   - **Convergence Layer**
   - **Statistical Layer**

   This analysis will help to:

   - describe for each type of data the activities (procedures, methods and tools) to perform, in order to transform raw data in statistical output. The Convergence Layer will map the transformation of raw units and attributes in core units and attributes for the statistical analysis;
   - describe data at the logical level;
   - align the application and the information layer of the TSS_u platform;
   - integrate existing architectural framework.

3.1 TSS_u platform business functions

Starting from the business functions in BREAL, we need to identify which of them are appropriate for describing the business layer of TSS_u platform (e.g., Acquisition and Recording, Data Wrangling, Data Representation) and specify the main steps of a generic TSS_u process. Particularly, each business function will be analyzed to identify the related process steps.

While a business function relates to the organization of resources and activities and can group different processes, a process describes the flow of tasks to perform to obtain a specific output. The design of the business layer will allow to describe each application component of the platform in terms of:

- types of input and output data;
- tasks to perform (each component performs one or more process steps described in the business layer);
- dependencies with other components (e.g.: metadata, privacy).

The idea is to start from the general requirements and for each sub-process of a generic TSS_u, to model each application component, regardless of the software specification. On a higher level, each sub-process and the corresponding application component will be related to one or more GSBPM sub-phases and GSIM structures.

3.2 TSS_u platform overarching business functions

The following paragraphs focus on the business functions related to the statistical process as a whole, or support the core business functions of the “Development-Production and Deployment” subset.

3.2.1 Metadata management

The overarching business function ‘Metadata management’ covers a wide range of information related to the domain (e.g. variables, classifications), the process, data provenance, data lifecycle, quality. Although these aspects are all relevant and interconnected, the architectural perspective may help to: i) identify priorities; ii) abstract from specific statistical domain; iii) facilitate an iterative implementation.

The suggested approach is to focus primarily on the subset of metadata directly related to each business function considered. In terms of GSIM concepts, in the stage of platform design, the metadata business function should allow to track at least the following aspects:

- Data provenance, particularly tied to Data acquisition and Recording;
- Process description: each process step has an input and an output, and performs different tasks, like applying a method or a data transformation.
- Process control: overview of all types of rules applied in each process step.
- Process documentation, in order to document data layer transformations and build indicators for process auditability and quality assessment.

This approach makes it easier to implement application modules metadata driven and allows to balance the domain agnostic requirement with the harmonisation of concepts and definitions within the same domain.

The following figure shows the metadata repository subsets, grouped according to the above criteria.

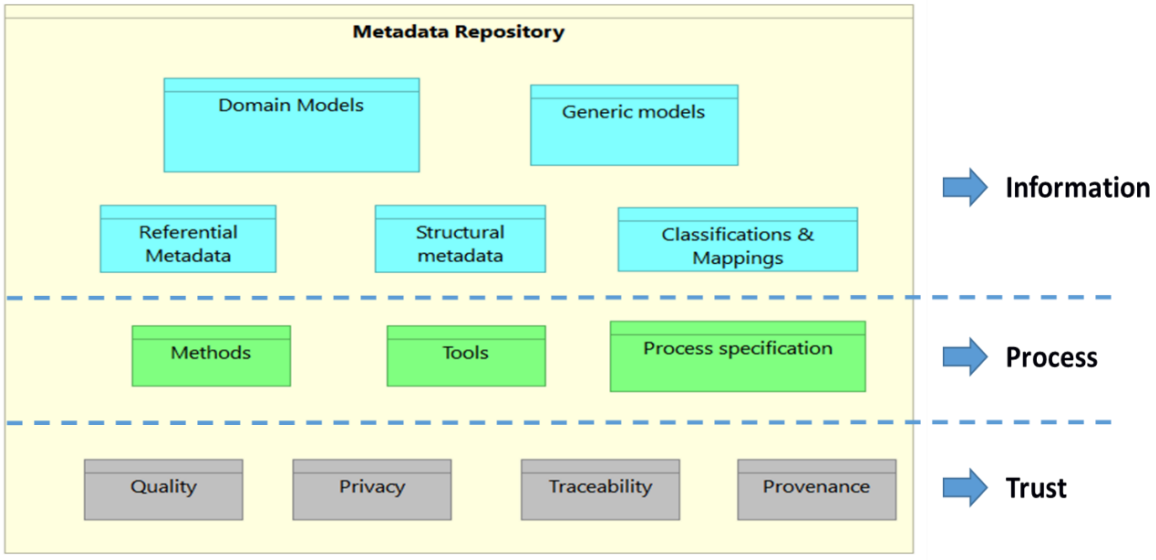

Figure 10: Conceptual description of TSS Metadata Repository

3.2.2 Manage statistical methodology

The methodological framework will result from the combination of a survey design and several methods applied in the process steps executed. These methods, according to a data layer perspective, can be grouped as follows:

- **Methods related to specific business functions**, that allow to transform Raw data in Convergent and Statistical data. This group includes also the methods used to assess data coverage and data reliability.
- **Methods for data integration, data processing and pre-processing**. While different types of data pre-processing may occur in every data layer for different reasons, data integration and some sub-phases of data processing (e.g. 'Review and Validate') may concern primarily the Convergence and the Statistical data layer.
- **Methods for new data sources exploration**: this subset includes all the analysis to assess the value of a new data source in terms of the information provided. Some of these methods could be used during the survey for evaluating on the fly the smart data collected.

The proposed classification of the methods to apply for TSS_u is useful to reach an inventory of the algorithms and statistical functions embedded in each application component. Mapping the methods with the business functions will also prevent overlapping of services and functionalities that execute similar procedures, thus facilitating the reuse and the adaption of implemented components. The reference to the data layer will enhance the generalisation of the application layer in terms of data structures to model, data versioning and data flow. The relevant information concerning these aspects will be managed by the Metadata component.

As the platform must be domain agnostic (as far as possible), the design of a generic survey will help to:

- model the process steps and the workflow to make it easy to set singular instances for specific surveys;
- manage the integration and the peculiarities of new and traditional data sources both during data collection and throughout the whole statistical process.

Concurrently, while the design of TSSu platform may provide an overview of the business and application layers, a more detailed methodological strategy is tied to specific scenarios and hypothesis. Many factors may affect the choice of a method, and focusing on the statistical use of smart data sources, some of these factors are:

- data structure;
- data provider;
- frequency of data provision (stock or longitudinal data);
- known/unknown data model;
- reliable algorithms to extract content from data;
- integration of smart data with traditional sources.

3.2.3 Privacy preserving methods

The use of privacy preserving methods is particularly relevant in a statistical process based on smart data, or dealing with sensitive data. This paragraph is a short inventory of the Privacy Enhancing Techniques (PET) discussed in the UN Handbook on Privacy Preserving[22].

The combination of these techniques may fulfil the general privacy goals of Input privacy, Output privacy and Policy enforcement. While Input privacy refers to data sources, Output privacy concerns the statistical results and the Legal Enforcement affects the whole system. Input privacy prevents Computing Party to access or derive input or intermediate values during data processing. Therefore, input privacy also includes the protection against the mechanisms that would allow the Computing Party the derivation of unauthorised data content. The goal of output analysis is to prevent the identification of input data within the published results unless Input Parties provide explicit consent. Output privacy is related to the measurement of information leakage resulting from a computation, regardless the input privacy provided in some cases by the computation itself.

The third privacy goal aims at improving the mechanism of positive control related to input parties. This control consists of a formal language, used to identify the computing parties and the rules they have agreed to follow. Some Policy decision points translate these rules into machine-readable language, while the Policy enforcement points correspond to the technical means to apply them. Policy enforcement affects the whole privacy-preserving system, providing rules for assuring input and output privacy at the same time.

The mix of the following PET, most of them deriving from cryptography, allows to fulfil the above principles:

- Secure Multiparty Computation (shortened MPC)
- (Fully) Homomorphic Encryption (shortened as HE or FHE)
- Trusted Execution Environments (shortened as TEE)
- Differential Privacy
- Zero Knowledge Proofs (shortened as ZK Proofs)

The first technique, Secure multi-party computation (also known as secure computation, multi-party computation/MPC, or privacy-preserving computation), allows to jointly perform an agreed-upon computation. As input data is divided in random shares, each participant is prevented to acquire any of the inputs provided by other parties. This method relies on secret sharing of both, inputs and intermediate results, so that the final output is correct.

Homomorphic encryption uses groups of encryption schemes, having a particular algebraic structure that allows to compute data without directly accessing to them. Only the party (in most cases, the owner of input data) that knows the secret key is able to decrypt the result.

Trusted Execution Environments (TEEs) is a capability for secure computation, resulting from the combination of special-purpose hardware and software. A mechanism implemented in the hardware creates a protected execution environment, where a process runs without processor memory, or being visible to other processes. In this environment, computation is not performed on encrypted

[22] https://marketplace.officialstatistics.org/privacy-preserving-techniques-handbook

data. The security issue is shifted on protected memory spaces, where code entries and exit points are tightly controlled.

Differential privacy (DP) is based on a general concept of privacy, related to data anonymization and to the models used for statistical disclosure control. This technique provides a theoretical framework of Output Privacy, as its goal is to measure and limit the quantity of individual information leaked, due to data aggregation. DP is also a privacy standard that states the property of an algorithm for data analysis, to preserve the input data privacy. This property is satisfied by removing a particular record from input data. To fulfil the DP requirement, the results of the algorithm applied in the real and alternate world, must be statistically indistinguishable, regardless of the type of database, or the removed individual. Thus, DP relates to the algorithm and does not concern input data, or the output of a computation. In general, only random algorithms satisfy the DP requirement of indistinguishability.

Among the cryptographic technologies, Zero Knowledge Proofs allows one party (the prover) to prove statements to another party (the verifier) using secret information known to the prover without revealing those secrets to the verifier. The main properties of this technology are:

- **Completeness**: If the statement is true and both the prover and the verifier apply the protocol, the verifier accepts the proof.
- **Soundness**: If the statement is false, and the verifier applies the protocol, the proof will not convince him.
- **Zero-knowledge**: If the statement is true and the prover applies the protocol, the confidential information is protected during the interaction between the prover and the verifier.

The ongoing work on this topic is huge and there are many enhancements in progress. Therefore, the privacy issue for TSS_u platform must be analysed from different perspectives, to consider the legal and technical aspects. The deliverable of task 3.1.4 will explore more in detail the different aspects of privacy preservation within TSS_u context.

3.3 Modeling TSS_u platform business layer

The analysis of a general statistical process and the alignment with existing framework and with BREAL for the smart component of TSS_u have resulted in the design of the platform business layer. The main objective of this model is to provide an overview of the main sub-processes executed in the platform. The architectural PoC will allow to specialize this general model for a specific use case (corresponding to a pilot survey chosen according to WP2 suggestion), thus specifying the steps of each sub-process. Further, the process design will highlight the relationship between process steps, data, methods and tools, providing technical requirements for the design of the application layer. ArchiMate[23] language has been used to model the business layer according to GSBPM standard (see the model shown in Figure 11).

The basic assumption is that regardless of the national context, the production pipeline of TSS_u is similar to traditional surveys, starting with data collection and ending with output dissemination. More precisely, the first sub-process relates to the design and implementation of apps or web-questionnaires and more in general to all types of tools to be used for data gathering. The next sub-

[23] ArchiMate is an open and independent language for modeling enterprise architecture, available from: https://www.archimatetool.com/.

process refers to the sample selection followed by the collection set-up. Concerning data collection, some relevant differences between traditional and TSS_u are highlighted by the sub-process GSBPM: Run and Finalise Collection. This sub-process is composed by a subset of steps performed to collect questionnaire data, regardless of the technology used to implement it, and by the subset corresponding to Sensor Data Throughput Activities, analysed more in detail in the deliverable of Task 3.1.6. These activities are connected to the smart part of the survey and may differ according to the type of sensor used for data collection. Therefore, they are associated to the following BREAL business functions: Acquisition and Recording, Data Wrangling, Data Representation and Modeling and Interpretation.

Considering the different data objects and more in general the data flow, TSS_u Survey Questionnaire Source provides input data to the first subset, while TSS_u Sensor Data Source provides input data to Sensor Data Lower Throughput Activities. Once all type of data has been gathered, GSBPM: Process performs data cleaning, while GSBPM: Analyse delivers a better understanding of data, enhancing the need of further processing. Finally, GSBPM: Disseminate refers both to the dissemination of TSS_u results, provided by TSS_u Survey & Sensor Data Information Set and to additional Trusted Smart Statistics (TSS) produced by ad-hoc analysis of sensor data and provided by Sensor Data Information Set. This sub-process serves BREAL: Shape Output business function that is the ability to format and present the statistical output.

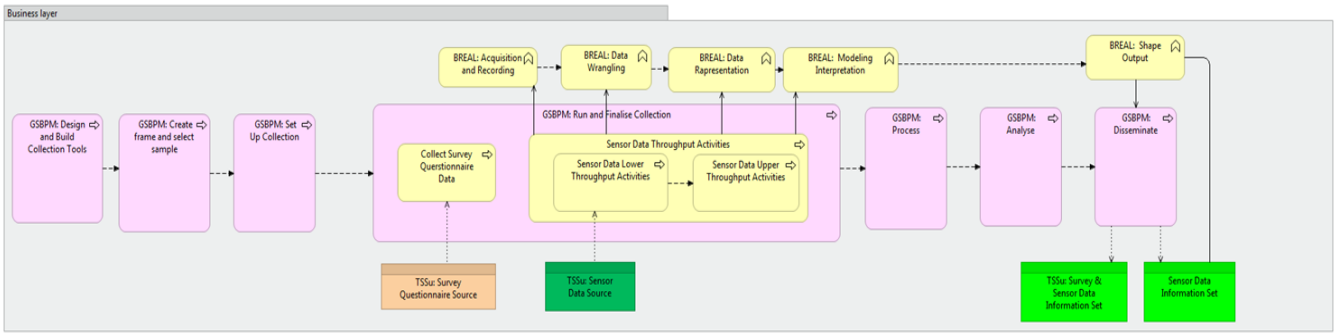

Figure 11: TSS_u platform business layer

3.4 Next steps

The overview of preparative work is the starting point for a deeper analysis of: i) data collection process steps, ii) application components involved in data processing and metadata management, iii) data transformations due to in-app or in-house data processing. Therefore, the metadata and architectural PoC to be implemented in the second phase of the ESSnet project will facilitate: i) the specification of the platform requirements, ii) an enhancement of the initial architectural framework. More precisely, the PoC should investigate the integration between sensor and traditional data, to highlight potential challenges during process execution due to the combination of different data sources. The expected results of this task is the design of the business process of a generic TSS_u. The analysis of the process steps of a pilot survey in terms of input/output data, methods, and metadata will allow to test the consistency between the theoretical framework and the actual survey implementation.

References

Bosch O., Quaresma S., De Cubellis M., Workpackage L: Preparing Smart Statistics - Deliverable L3: Description of the findings regarding Task 3, Smart Devices. Final version, 30th October 2019

Minnen J., Nagel E., Sabbe K., SOURCE TM: Software Outreach and Redefinition to Collect E-data Through MOTUS. Towards a Modular Online Time Use Survey. Main Report. April 2020

Mussmann O., Schouten B.: Final methodological report discussing the use of mobile device sensors in ESS surveys. MIXED MODE DESIGNS FOR SOCIAL SURVEYS - MIMOD, WP5 deliverable (2019). Edited by EUROSTAT

Ricciato F., Wirthmann A., Giannakouris K., Reis F., Skaliotis M.: Trusted smart statistics: Motivations and principles. Statistical Journal of the IAOS (November 2019)

Schouten B., Bulman J., Järvensivu M., Plate M., Vrabič-Kek B.: Report on the action @HBS - Version 1

Scannapieco M., Bogdanovits F., Gallois F., Fischer B., Georgiev K, Remco P. et al.: BREAL. Big Data Reference Architecture and Layers. Version 2019-12-09. Edited by EUROSTAT

Annex 1: ESSnet Mimod

The objective of the ESSNet MIMOD[24] - Mixed MOde Designs in Social Surveys - project was to rationalize theoretical knowledge, to gather experiences and good practices for NSIs and to provide indications on how to use mixed-mode survey designs ensuring the quality of results.

The Mixed Mode (MM) designs, i.e. the use of different data collection techniques in the same survey, are adopted by NSIs both, to contrast declining response and coverage rates and to reduce the cost of the surveys, for example introducing web mode. However, MM designs introduce several issues that must be addressed both at the design phase, by defining the best collection instruments to contain the measurement error, and at the estimation phase, by assessing and adjusting the bias effects (mode effect). The accuracy of the estimates, in fact, has to be preserved from possible increase in the total survey error, due to extra measurement error introduced by the additional data collection modes. MM design can generate differences between the outcomes of modes due to differences between the respondents, or by differences in measurement error. The selection effect refers to differences in the population coverage and in nonresponse processes, while the measurement effect refers to mode features that can influence the answers to survey questions. The main problem of MM design is that selection and measurement effects are confused. This involves a complication of the inferential process in which methods to disentangle the two effects are needed, to obtain unbiased estimates of measurement error.

The introduction of the web mode in surveys that adopt mixed mode designs (WP1)[25] and the consequent implications, both in the data collection phase (WP3, WP4) and in the estimation phase (WP2) have been the main focus of the ESSNet MIMOD project. The web mode has been analysed not only with reference to the more traditional case in which respondents use a desktop personal computers (PCs) or notebooks, but also considering the possibility of using other devices that have Internet access (WP5).

WP5 of the MIMOD project investigates the employment of mobile devices in ESS surveys. In particular, it explores fitness of ESS surveys for smartphones (WP1, WP4) and the utility of mobile device sensors to replace and/or supplement survey data.

WP5 provides an initial inventory of potential sensor measurements. First, it presents criteria for sensor data, considering the point of view of *surveyor* (survey topics may be candidates for enrichment or replacement with sensor data when they satisfy at least one), *sensor* and *respondent* (sensors may vary in their intrusiveness).

These criteria are summarized in the following table.

[24]For an overview of the project, see the Final report, available from: https://ec.europa.eu/eurostat/cros/system/files/final_report_CP_rev1.pdf (Accessed: 20 October 2020)

[25] The deliverables of the project are available from: https://www.istat.it/en/research-activity/international-research-activity/essnet-and-grants

Point of view	Criteria
Surveyor	**Burden**: the survey topic(s) are burdensome for a respondent, in terms of time or cognitive effort **Centrality**: the survey topic(s) are non-central to respondents, i.e. the average respondent does not understand the question or does not know the answer **Non-survey type**: the survey topic(s) do not lend themselves to a survey question-answer approach to begin with
Sensor	**Omnipresence**: the sensor(s) are available in most, if not all, contemporary devices (Smartphone sensors are examples of omnipresent sensors, whereas sensors in wearables have a much lower population coverage and pose challenges with regard to data access) **Data access**: data generated by the sensor(s), as well as metadata about the properties and accuracy of the sensor data, can be accessed and processed (it is not always clear what sensor, GSM, Wi-Fi or GPS, produced the data and how accurate the data are) **Quality**: the sensor data is comparable, reproducible and accurate (statistical objective to derive accuracy of statistics and to be able to compare statistics between persons and in time) **Costs**: any costs associated with the sensor(s) are affordable in most surveys
Respondent	**Respondent willingness**: respondents are willing to consent to provide the sensor data **Data handling**: respondents can retrieve, revise and delete sensor data on demand **Burden**: respondents are willing to devote the effort needed to collect and handle the sensor data **Feedback**: respondents may retrieve useful knowledge about themselves

Table 10: Sensor data criteria from the surveyor, sensor and respondent perspective

Other important overviews presented in the deliverable concern:

- a list of elements and sensors supported by many contemporary smartphones/tablets and sensors that are included in wearable devices;
- the distinction between active and passive data collection and secondary sensor data;
- the analysis of some ESS surveys - Labour Force Survey (LFS), European Statistics on Income and Living Conditions (EU-SILC), European Health Interview Survey (EHIS), Information and Communication Technology survey (ICT), Household Budget Survey (HBS) and Harmonized European Time Use Survey (HETUS) - considering the sensor data criteria and the potential sensor measurements for each survey.

The following table shows the potential sensor measurements for ESS surveys.

Survey	Sensors	Secondary sensors
LFS	Time-location, mobile device use	Social media data, Mobile phone provider data, Internet provider data

EU-SILC	Camera, microphone, time-location, mobile device use	Social media data, Mobile phone provider data, Internet provider data, Smart energy use meters data (electricity, water)
EHIS	Time-location, motion, heart rate, wearables, camera	Wearable sensor data
ICT	Mobile device use, mobile device properties	Social media data, Mobile phone provider data, Internet provider data
HBS	Time-location, camera, mobile device use	Scanner data from shops, Bank transaction data, Loyalty card data
HETUS	Time-location, motion, mobile device use, NFC, Bluetooth, wearables	Wearable sensor data

Table 11: Sensor measurements for ESS surveys

Task 3.1.4:
Preservation of privacy and transparency

Prepared by:

Mirela Causevic (CBS, The Netherlands)

Joeri van Etten (CBS, The Netherlands)

Task leader:

Rob Warmerdam(CBS, The Netherlands)

Outline

Executive summary

Trusted Smart Surveys aim to provide means for NSI's to leverage the data collected by smart devices which have become ubiquitous in modern society. When novel data collection strategies are proposed and data from new sources is collected, concerns with regards to security and privacy are inevitable, and rightly so. NSI' should take great care for the data they collect so as to not let the privacy of respondents be compromised.

However, at the same time, NSI's are obligated to serve the public by producing accurate statistics. NSI's should hence look for ways to enable leveraging new data-sources, while preventing privacy being breached. To assist in this process, this chapter provides the reader with a set of tools that might help with understanding and dealing with the privacy considerations involved in the aforementioned new forms of data collection.

Specifically, in the first section of this chapter, an overview is given of the Privacy-by-design framework for handling private information. This framework is subsequently converted into a set of guidelines which should assist in process of making correct decisions with regards to how private data should be dealt with by an application which collects such data.

The second section is dedicated to techniques for preserving privacy while maximizing data utility. An overview of a number of such techniques (e.g. differential privacy, homomorphic encryption, secure multi-party computation, etc.) is given. The benefits and drawbacks are highlighted, as well as typical use-cases. Finally, a discourse on what these techniques could mean TSS applications is presented. Also, a number of scenario's in which these techniques could be applied in the smart survey process are described.

In the final section, the issue of user authentication is tackled. Naturally, the identity of TSS participants should be verified. However, this should be done using a minimal amount of information to reduce privacy impact. A protocol for user authentication using a minimal set of characteristics is therefore presented in the form of IRMA, short for 'I Reveal My Attributes'. Finally, an example implementation of IRMA in the context of a TSS application is given.

1. Introduction

Data is part of more and more aspects of our lives and can be potentially very valuable in offering insights that are useful for society. This data can be generated in different ways, through using the world wide web, smartphones and their sensors, IoT devises and their sensors, and through actively participating in research.

Statistical offices are simultaneously noticing a decline in response on questionnaires from the general population. In this sense, it is valuable to explore the more automatic collection of data through sensors, as this might help to compensate for the decline in response, as well as possibly bring new possibilities for statistical output.

However, in all data collection there is an inherent concern regarding privacy, with varying levels of possible concern. Much research has been, and is being done to enable data mining while ensuring at least some level of privacy.

If viewed from the perspective of wanting to increase this data mining, one would need to work on implementing privacy restrictions from the perspective of the people who generate the data in order to protect them, now and from future usage. In the sharing of data, the objective would thus be to not only store, but also collect and process these in a privacy preserving manner.

Given that new forms of data are increasingly continuous, dynamical and diverse, it may also be useful to think about (partial) automatic and decentralized collection and processing.

Many researchers are looking for valuable data. Not in the least place are researchers at National Statistical Offices. In this context, the ESSnet has devised a vision for deploying smart surveys, in which participants can provide information in a more specific and less burdensome way. The goals is to collect more accurate and rich data sources for NSI's, and as such that it brings a better representation of a society. [1]

This document aims to develop a thinking framework around guidelines, requirements and methods for the (partial) automatic and decentralized collection and processing of data.

The requirements are focused on mutual agreement on what privacy means in a given context, not just what regulations state at a given moment in time.

An important part that is also considered, is in what way and how well this is communicated to users and how much control they can have. It should be as clear as possible, and collection should be controllable.

Guidelines for the smart survey context could be developed by:

1. Using solid, existing privacy guidelines.
2. Specifying them to assist the users' understanding.
3. Assessing both privacy risks, and research value, aiding in informed decisions between trade-offs.
4. Suggesting methods and technologies to control processes based on the guidelines and assessments.
5. Providing a worked through example by looking at a mobility application for a household study.
6. Serving as a handbook for reference usage and further developing in the context of collecting and processing (sensor) data for NSI's, specifically for using in smart surveys.

The overall goal of this component within Work Package 3 is to describe and asses the practical applicability of the intended Trusted Smart Survey (TSS) platform. The elements of data collection within a TSS platform

are all measurement modalities that use the sensors which are present in the device. It's specifically intended to leverage this data, in order to de-burden traditional data collection in official statistics, like traditional surveys. A hybrid form of data collection like surveys and combined with sensor-measurement is seen as part of the TSS platform.

The aim is to contribute towards theory, as well as a minimal viable product. The elements the platform should contain are among other things:

IT-infrastructure, methodology, incentives for participation, privacy-preserving examples, and considerations regarding meta-data descriptions.

Ideally, all countries involved in ESSnet could adapt and copy the methods and technologies explored in this documentation for their own specific use-cases and respecting their own restrictions and context.

2. Guidelines

2.1. Context of the Guidelines

These guidelines aim to ensure privacy in the context of smart surveys, which can mean usage of mobile phone sensor data as well as opt-in surveys as triggered by sensor data. Privacy should be at the core of thinking about this and based on solid research and implementation where possible.

For this, it is useful to look at the original seven principles of privacy-by-design [2] and where possible, extend them into the context of what is needed for ensuring privacy for smart surveys.

Part of this translation, is aiming to somewhat specify privacy, especially from the user perspective. All too often, privacy agreements are not as understandable to everyday users as they should be, however correct in adhering to the law these agreements may be. The result is that they often don't get read or are misinterpreted because they are read only partially, and selectively.

The seven privacy-by-design principles are the basis, and listed below:

1. Proactive not reactive; preventive not remedial
2. Privacy as the default setting
3. Privacy embedded into design
4. Full functionality – positive-sum, not zero-sum
5. End-to-end security – full lifecycle protection
6. Visibility and transparency – keep it open
7. Respect for user privacy – keep it user-centric

Following this, the interpretation for (sensor)data collection and smart surveys results in:

1) These guidelines and methods are pro-actively specified, given the expected context and estimations of possibilities. In addition to that, to remain preventative, they should be periodically reviewed and changed if needed.

2) Privacy is the default setting, and all alterations to these settings are viewable and adjustable by the data producer (or, user of the data collecting application). The users should control not only portions of the application, ideally also granular levels like individual processes.

3) Privacy is at the core of the design, and as such the definition of privacy itself is specified as detailed as possible, to increase mutual understanding, and the modes in which it could be (transparently) changed.

4) Full functionality should be available, but with the knowledge of risks, an informed decision could be made on the of elements of this functionality at different moments.

5) Privacy, and the options thereof, are part of the core of the approach, and as such should be part of the full lifecycle of a given application.

6) To fully respect user privacy, in addition to providing transparency and adjustability, a risk-scale is used, which can be used to trade-off with an indication of potential research value.

7) In general, these guidelines are based on personal preferences, ideally also informed by a (short version of) a risk-scale, based on which user can adjust their privacy settings.

2.2. The Three Guidelines

1 – Clear definitions and distinctions between elements:

Clarity about the used definition of privacy, given the context, should be ensured for the party who generates the data.

Given these clear definitions, there should be an exact distinction between:

1. Which processes are being described?
2. The tone of the description should provide clarity regarding potential privacy related risks, adhering to the previously mentioned scale.
3. The variation in description methods in terms of detail, length, update frequency of the terms, should all be as clear as possible,
4. How will the level of clarity be tested? And what will be done with feedback regarding the given topic?
5. Clarity on how will be dealt with changes of usage, or unintended usage for added analyses.

2 – View-ability:

The processes required for i) collecting data, and ii) processing data, including any underlying inter-dependencies, should always be easily viewable and available for the further inspection of their specific access and working, by the generator of the data and user of the given tool, as well as whoever is collecting and processing the data. Methods should be in place to control the gradation of complexity when viewing these aspects. Through a form of layered complexity a data generator can select what level of depth the explanation should be, tuning the way in which they wish to view it at a given time.[26]

26 Special attention should go to making this truly accessible and interpretable. This is increasingly challenging as many users don't read terms of service or privacy guidelines. One could regard the agreement as not that useful for the party generating the data, if they are not aware of the content. A possible approach could be to select the most relevant elements, sum them up in easy to understand pieces of texts and (moving) illustrations, which the data generator (or user of a given application) should go through, clicking "OK" but which are so simple and easy it would be more likely that they would have read and understood the content. A few images and short texts which remain on screen for a few seconds after which "OK" can be selected to move forward to the next brief explanation, resulting in a full, albeit summary-based, consent to the policy.

3 – Access to processes:

The party who is providing the data, (the 'Data Collector) should have as much as possible access to the processes involved, preferably at any given point. These processes should preferably enable their switching off. If this is not possible, during installation of the application this should be clearly communicated and asked for consent in this regard. Wherever possible, the application should use modular design and compartmentalization to aid in enabling this.

3. Privacy Preserving Techniques

Attempts of protecting the privacy of individuals whose data is contained in a dataset usually starts by removing all personally identifiable information. For instance, a dataset containing medical records might contain insurance data unique to a given individual. Removing these data, or replacing them with, for example, randomly generated id's, would prevent identifying the medical record of a given individual. This process is called anonymization and is usually the first step in privacy preservation.

Naively, one might expect anonymization to be a sufficient means for preventing a security breach to compromise the privacy of the individuals whose information is contained in the leaked dataset. However, when taking into account the possibility of combining different data sources, anonymization alone appears to be insufficient. For example, in 2007 researchers were able to recover 99% of all personal information contained in a Netflix dataset, which had been made public for a machine learning competition. Supposedly, all personally identifiable information had been removed, but researchers were able identify the individuals by combining the Netflix dataset with public data from the Internet Movie Database [3]. Specifically, the individuals were uniquely identified by their ratings of three movies and the approximate date on which these movies were watched. Similarly, Latanya Sweeney [4] has shown that, for example, 87% of Americans could be uniquely identified by gender, ZIP code an date-of-birth and sometimes. This illustrates that datasets might contain more unique identifiers than one might suspect and that anonymization at the level of individual datasets is not always sufficient.

Another way of preventing sensitive data from being exposed is by encryption, the digital equivalent of storing precious items in a safe. This method can very effective in preventing individual records from leaking. However, in order for the data to be used, it has to be decrypted, leaving it, once again, susceptible to attacks.

These considerations highlight the two main factors under consideration when talking about privacy preserving techniques. Firstly, we should think about the level of protection a given technique provides, taking into account the possible linkage of data to all other possible sources. Secondly, the usability of the data should be considered. A trade-off between these two factors should be made while considering the status of the data. When data is in use, the privacy preserving technique used should provide maximum usability, possibly at the cost of some security. However, when the data is at rest, usability can be sacrificed in order for more robust and computationally cheaper protection to be possible.

3.1. Advanced Techniques for Privacy Preserving Data Analysis

Both previously mentioned techniques, anonymization and encryption, are perfectly valid ways of protecting privacy. However, as discussed, anonymization alone might not provide enough protection, while encryption

limits usability too much for it to be applied to data in use. For this reason, researchers have been looking for new ways of preserving privacy, while retaining usability. In what follows, a discussion of these techniques is provided, in the light of the aforementioned considerations.

3.1.1 Differential Privacy

In order to have meaningful discussions about privacy preserving techniques, it is useful to have a rigorous definition of privacy and a means of quantifying the amount of privacy a given technique provides. For these purposes, the concept 'differential privacy' was introduced. In words, the statement of differential privacy can be expressed as the following promise made by a database owner to a subject who appears in the database[5]:

"You will not be affected, adversely or otherwise, by allowing your data to be used in any study or analysis, no matter what other studies, data sets, or information sources, are available."

Differential privacy thus tries to ensure that the outcomes of any analysis based on a given dataset are independent of whether any individual is included in the dataset or not.

Mathematically, the simplest form of this statement can be formulated as follows: given two databases D and D' which differ by only a single entry, an algorithm A and a set of outcomes $A(D) \in S$, then A is said to provide ϵ-differential privacy if the probability for a given outcome to occur when the algorithm takes D as input differs at most by a factor e^{ϵ} from the probability for that outcome when D' is taken as an input.

$$Pr[A(D) \in S] \leq e^{\epsilon} Pr[A(D') \in S]$$

The parameter ϵ captures the amount of privacy provided by the algorithm. Note that, for example, when ϵ is equal to zero the datasets are indistinguishable as far as the algorithm A is concerned, i.e. it is impossible to tell whether the record relating to a single individual is in the dataset or not.

For further mathematical background, the curious reader is referred to [5].

To understand how an algorithm might provide differential privacy, let us consider the case of randomized response a research method proposed to grant survey participants plausible deniability [6]. Say we conduct a survey on cheating. We ask the survey participants whether they have ever cheated, but before they answer, they are asked to flip a coin. If heads, they are asked to toss again and answer the question truthfully. If tails, however, they are asked toss again and answer "yes" if heads, or "no" if tails.

This way, half of participants will answer truthfully, while the other half will answer randomly. Given the ¼ probability that a single answer is incorrect, the participants have plausible deniability about their individual response. However, given enough participants, we are able to make statistically significant claims about the population as a whole.

Differentially private algorithms work in a similar way. Privacy is created by adding some form of noise, like the coin-toss does in the previous example. This can be done at two levels, the input level: noise is added before data is stored in a dataset, or the output level: noise is added whenever the data is queried. Given addition of a similar amount of noise, the former offers more robust privacy protections, while the latter retains higher data fidelity. Also, when sharing of data might be desired either now or in the future, local differential privacy should be used, as privacy needs to be preserved before the data is shared.

There are a few caveats when it comes to differentially private algorithms however. Firstly, adding noise to the data obviously reduces the quality of the data. The amount of noise correlates inversely with the quality of the data, so when implementing a differentially private algorithm, the usual trade-off between privacy and usability still has to be made. In situations where high precision is required and data is very sensitive, these algorithms might therefore not be the best option. Secondly, it is important to take into account that when global differential privacy is used, repeated querying of the same dataset reduces the privacy budget. This happens because aggregating multiple query results enables one to filter the noise by averaging over multiple queries. Given a specific application, a suitable amount of privacy, represented by the epsilon parameter, should be chosen based on that application. These caveats illustrate the need to have a clear understanding of the data and its uses before differentially private algorithms are implemented.

Options for implementing differential privacy algorithms include libraries for various programming languages like

- TensorFlow Privacy for differentially private machine learning Python[7]
- IBM's differential privacy library for Python[8]
- Google's differential privacy library for C++[9]

Many of these libraries are open-source, hence implementing differential privacy does not have to be a costly solution for privacy preservation. In fact, the use of differential privacy is already quite common. Examples of current day usage include:

- iOS and macOS keyboard statistics[10]
- Private data Sharing Interface(PSI) developed for Harvard University's Privacy Tool's project[11]
- Application usage statistics in Windows 10[12]
- OnTheMap: tool for visualizing US commute patterns[13]

However, the use of differential privacy has not been without any critiques ([14, [15], and [16]). It seems that either implementation is not always as secure as it is supposed to be, or the reduction in data quality is too high. To determine whether these critiques invalidate the practical applicability of differential privacy as a concept, or just specific implementations thereof, further investigations are necessary.

3.1.2 Homomorphic Encryption

Regular encryption has been around for a long time and its use is wide-spread. However, as noted before, it has one big downside: data has to be decrypted before it can be used. This is where homomorphic encryption comes in. Namely, a homomorphic encryption scheme enables the possibility of performing computations on encrypted data without the need for it to be decrypted first. Thus, homomorphically encrypting data retains all the benefits of regular encryption, while doing away with the negative.

There is a downside however: homomorphic encryption schemes are computationally very expensive. To save on computational cost, partially homomorphic encryption schemes are available as well, which allow only a subset of operations to be performed on the data. These schemes provide enhanced performance at the obvious cost of reduced operational flexibility.

To understand how homomorphic encryption works, let us once more look at a simple example. Say we have a binary message, i.e. the message is either 0 or 1. We encode the message as follows:

$$C_p(m) = m + 2r + qp$$

Here, $C_p(m)$ is the encrypted message, r a noise term, p a large prime, which acts as the encryption key and q a large integer. The noise term is added for protection purposes; without it, extracting the encryption key p is simply a matter of finding the greatest common divisor of two messages. Decrypting is now as simple as taking mod p of the encrypted message.

To see that this encryption scheme is homomorphic, consider adding the encryptions of two messages:

$$C_p(m_1) + C_p(m_2) = m_1 + m_2 + 2(r_1 + r_2) + (q_1 + q_2)p$$

Again, by taking $mod(p)$ we decrypt the message. The result is the sum of both messages, so we have performed addition without revealing the individual contents of the messages. Multiplication works intuitively as well and one can check that decrypting the result of multiplication of two encrypted messages produces the correct result. Hence, we have achieved homomorphic encryption of the binary message.

Unfortunately, adding to many numbers will generate a noise term that grows larger than the p and the encryption breaks. An additional step therefore has to be added to the encryption scheme to reduce the noise. These noise reduction steps are the cause of the high computational complexity of homomorphic encryption schemes and much research is being done on reduction thereof.

Note that the above encryption scheme is symmetric, i.e. there exists just one key with which data is both encrypted and decrypted. Most practical homomorphic encryption schemes are, however, antisymmetric. That is, one requires a public key to encrypt a message, but another, private key to decrypt. In the case of homomorphic encryption, sometimes an additional key is needed to perform computations.

There exist various options for implementing fully homomorphic encryption. Among others, these are:

- Microsoft's SEAL[17]
- HElib by IBM[18]
- PALISADE by New Jersey Institute of Technology, Duality Technologies, Raytheon BBN Technologies, MIT, University of California, San Diego and others[19]
- HEAAN by the Seoul National University[20]

Like with differential privacy, these libraries are open-source, so homomorphic encryption can be implemented without any cost in terms of software. However, since computational cost is high, the overall price of homomorphic encryption is much higher than that of, for example, differential privacy.

3.1.3 Secure Multi-Party Computation

Combination of data sources often enables the extraction of information that would not have been available had the data sources been kept separate. However, when multiple data owners are involved, combining different sources of data involves exposing the source data to one or more of the other involved parties. In an ideal world, a trusted party would exist who could compute the function given the inputs of the other parties. However, in general, we should assume such an idealized is not realistic.

Secure Multi-Party Computation attempts to simulate the trusted third party cryptographically. The goal of SMPC is therefore to create a protocol for multiple parties to compute an arbitrary function on their inputs, without revealing the actual value of the input to one another. As such, secure multi-party computation is similar to homomorphic encryption.

Similar to homomorphic encryption, the goal of SMPC is to enable computation using values that remain hidden. A fully homomorphic encryption scheme could hence be used to enable SMPC. However, to enable SMPC, it is possible to use protocols that are less computationally expensive. In this sense, secure multiparty computation can be easier to implement. However, the requirement of multiple parties brings architectural difficulties at various levels such as software, hardware and judicial.

3.1.4 Trusted Execution Environment

Whenever software based solutions for providing secure data handling are deemed unsuitable, another possibility would be the implementation of a hardware based approach. Trusted Execution environments enable such a hardware based approach. Such an environment can exist within physical CPU or be a dedicated subsystem within a larger hardware system. The main benefit of trusted execution environments is that they are secure against any attack at the software level. The downside however, is that dedicated hardware is required. Also, whenever the data leaves the environment, it is as susceptible to attack as ever. Hence, additional measures will always be required.

Trusted execution enabling technologies are provided by companies like AMD, Intel, ARM and IBM. Currently, these technologies are being used for biometric identification methods, mobile financial services, digital rights management. In the context of TSS, trusted execution environments could prove to be useful for shielding TSS gathered data on the participant's device from possibly malicious software. Whether such an implementation is within the realm of current possibilities should be investigated.

3.1.5 Federated Learning

Machine learning involves feeding a model large amounts of data to 'teach' the model something about the set of data that is fed. Each time a model is fed a new batch of data, parameters and weights are updated to yield a more accurate one than the previous iteration. Traditionally, this process takes places centrally. That is, the model is stored at a given location and the data is brought to the model for training. This approach is dependent on the possibility of the data being brought to the model, which might not always be possible, especially when data is considered private. For this reason, federated learning was proposed as a means of training models on data that might, for some reason. In general, this is achieved by distributing a model across the sites where training data is located and training the model locally on the local, heterogeneous datasets. The resulting updates to the parameters and weights are then shared with the other sites to combine training results with other locally trained models. The final model is then a combination of the many locally trained models. Federated learning thus provides a way to build a model that benefits from various private data sources, without the need to have full access to these.

At this time, the number available libraries for enabling federated learning (some examples are TensorFlow Federated[21], PaddleFL[22] and PySyft[23]) is quite small and all have their shortcomings in terms of features and/or support. Implementation of federated learning in practice might therefore not be feasible currently. However, much work is being done in this sub-field of artificial intelligence, so practical implementation might be more feasible in the future.

3.2. Comparison of Different Techniques

To summarize, a table identifying strengths, weaknesses and potential usage scenarios is given below in table 1.

	Benefits	Downsides	When to use
Differential Privacy	- Cost-effective - Easy to implement - Quantifiable amount of privacy protection(although never absolute)	- Trade-off between usability and protection necessary - Reduced accuracy	- Aggregate statistics
Homomorphic Encryption	- Protection on par with regular encryption - Enables secure use of untrusted cloud service	- Computationally expensive	- Secure processing of confidential data in untrusted environments
Secure Multi-Party Computation	- Protection on par with encryption	- Multiple parties required - Computationally expensive(although less so than homomorphic encryption)	- If combining sensitive data form various sources is desired
Trusted Execution environments	- Secure, even with respect to malicious software	- Dedicated hardware required	- Protection against malicious software
Federated Learning	- Enables machine learning on private data	- Sufficient local processing power necessary - Complicated infrastructure - Less efficient and less accurate than centralized training	- When machine learning on distributed, private data is desired

Table 1: Comparison of Various Privacy Preserving Techniques

3.3. Advanced Privacy Preserving Techniques for TSS

The privacy guidelines proposed in the first section force NSI's to take privacy very seriously when designing smart surveys. In fact, privacy-by-design principles imply that NSI's should strive for complete privacy of respondents. In the design of these applications, there should hence be the desire to keep respondent data as private as possible. Privacy preserving techniques play a crucial role in achieving this goal. However, as has become apparent from the discussion of the different privacy preserving techniques, implementation of these comes at a cost. Most often, this cost is related to the computational complexity of PPT, which inhibits the scalability of these solutions. The question of how to solve privacy issues can therefore be, in part, transformed into the problem of reducing computational complexity.

A way to achieve this reduction in computational complexity is by decoupling the different steps needed to produce a given statistical result into a part that relies on processing of local data present on a single smart device and a global part, which involves aggregation of data collected from different respondents. For example, say we have trained a machine learning model to determine whether people are at home, based on data from various sensors. We would like to use the predictions of this model for the production of time-commitment statistic. Production of this statistic involves two steps: (1) inference based on sensor data and (2) aggregation to produce statistical results.

Given the computational complexity of the first step, application of PPT would not be feasible. However, seeing as this step uses solely local data, PPT would not enhance privacy of the respondent. The second step does involve sharing the data, as it needs to be aggregated with data from other respondents to produce statistical results. Of course, sharing the respondent's data poses a privacy risk. Luckily, given the considerably lesser computational complexity, even the most complex PPT could be feasible. Of course, statistical operations more complex are fathomable, limiting some complex PPT to specific use cases.

Therefore, providing robust privacy protection for respondents should involve reducing computational complexity by locally processing data and subsequently implementing state-of-the-art privacy preserving techniques to enable the production of aggregate statistics without any compromise to privacy.

On the other hand, one could also consider a different point of view. If we consider a smart survey as simply being a replacement for traditional surveys, additional privacy preserving techniques might not even be warranted seeing as they are not always that different from a privacy perspective.

To illustrate this similarity, consider the following example. Say we have an application that gathers geo-spatial data for a mobility survey. This application would replace a traditional mobility survey in which participants are asked to fill out a form detailing their movements throughout the day. At this time, these data would be protected by regular encryption when the data is transmitted to the NSI. Once it arrives at the NSI, the data is considered safe by virtue of being within the confines of the NSI, an environment that is considered to be secure.

If we now consider the 'smart' survey, the method of data collection changes, however, from a privacy perspective the situation remains the same. Data is collected on a participant's device, sent to the NSO and analyzed within the digital confines of said NSO. Given this consideration, the current data protection model should suffice for the smart survey as well. Additional privacy preserving techniques would therefore not provide any added benefit and be a waste of computational resources.

So which of the two points of view described above should we adopt? The answer to this question depends on multiple factors. First of all, the interpretation of the privacy guidelines and the way they relate to current privacy protection measures plays a big role. If we interpret the guidelines in such a way that obligates the NSO to do all it can to protect a respondent's privacy, PPT is essential. However, if current privacy protection measures are deemed sufficient, then why would we add more if, from a privacy perspective, there is no significant difference between them?

Secondly, the future vision for TSS should be considered. Should the goal of TSS be to solve the growing problem of non-response by replacing existing surveys with smart counterparts, so as to gather just enough data to keep producing currently produced statistics? Or should we use TSS to leverage the ever growing data-sources for the production of more and increasingly accurate statistics. In case of the former, current data protection measures should be sufficient. However, the latter vision requires a desire to increase the number of available data-sources by increasing the number of respondents as well as the amount of data they agree to share. Fulfillment of this desire is likely enabled by the use of privacy preserving. Both to cover legal bases, as well as nurturing the relationship of trust required between respondents and NSI in order for respondents to be willing to share increasing amounts of data.

In the end, NSI's should probably answer the question on whether or not advanced privacy enhancing measures are necessary on a case-by-case basis. Also, since the trade-offs involved in application of privacy

preserving measures are always evolving due to changes in legal, technological and civil considerations, continuous re-evaluation will probably be necessary.

3.4. Example Scenarios for Application of PPT in TSS

Homomorphic encryption for processing of single respondent data in public cloud

If the amount and complexity of preprocessing that has to be done on the respondent's device exceeds the computational capability thereof, homomorphic encryption could be used to enable secure computation in the public cloud. In this scenario, the respondent holds both the encryption and decryption key, such that both the input and output of the computation stay private.

Private Machine Learning on Respondent Devices using Federated Learning

Implementation of federated machine learning within TSS apps could enable training of machine learning models using sensor data (of course, companies like Google and Apple are already doing this). If sensor data could be combined with respondent input to label training data, effectiveness would be considerably increased, allowing for building robust models.

Secure Multi-Party Computation for Aggregate Statistics

After local pre-processing to turn high-dimensional, unstructured sensor data into low-dimensional structured data, aggregation is necessary to produce statistical results. For this step in the statistical production process, data needs to leave the respondents device so concerns regarding privacy are warranted. Secure Multi-Party Computation could provide a solution for this use-case, without compromising quality of the output. Concerns regarding computational cost are mostly mitigated the fact that local pre-processing reduced the complexity of the input data, so scalability should not be a big issue. Therefore, an SMPC solution with a significant number of parties could be feasible.

Differential Privacy for Aggregate Statistics

Similar to the way Apple [10] uses differential privacy for the collection of user data from various apps within certain privacy boundaries, trusted smart surveys could employ differentially private algorithms to learn just the right amount to produce aggregate statistics, while not learning enough to risk the privacy of respondents. In this scenario, the respondent should be considered an output party who does not want certain things to be discovered within his/her data. This approach is simpler in terms of computational complexity than Secure Multi-Party Computation, so given a scenario where local pre-processing is insufficient in reducing dimensionality of the output and/or when statistical operations are too complicated to be performed in an SMPC setting, differential privacy could provide a solution. Note however, that in contrast to secure multi-party computation, differentially private mechanisms necessarily decrease the accuracy of the data, so this solution might not always be the right choice.

Differential Privacy for ensuring Private Final Output

Techniques like homomorphic encryption and secure multi-party computation can provide guarantees in regards to the privacy of input data. However, statistics that are produced using this private input data should preserve privacy as well. In particular, published statistics should be protected from re-identification by linkage with public data. Differential privacy provides this protection, so somewhere along the processing chain, this technique could be implemented to mitigate these risks. Note however, that concerns with

regards to output privacy are not special to TSS, so policy that is currently in place for protection of output privacy at NSI's might suffice.

4. Authentication and Verification

In the context of privacy, an extended need is also the verification of identity, and related attributes. The Trusted Smart Surveys (TSS) applications might benefit from an authentication which is as strong as is technically and legislatively feasibly, and as easy to use for the users as possible. Ideally, it would be open source, to ensure complete verification for the application provider and all parties involved.

In this sense, it might be useful to look at the IRMA protocol, or set of open source tools. IRMA is short for 'I Reveal My Attributes', and it can be can be used for authentication [24]. The main advantage of IRMA is that the attributes are selectively revealed, based on specific need. In other words, if a situation only requires the verification of age, no other identification attributes will be shared, like in other cases. In the context of the TSS, the main elements which needs verification would be someone's identity and possibly a few personal details, like age, location, gender, occupation. These elements are referred to as attributes in this document. In short, *IRMA is a set of free and open source software projects implementing the Idemix attribute-based credential scheme, allowing users to safely and securely authenticate themselves as privacy-preserving as the situation permits.*

The referred to *Idemix attribute-based credential scheme* uses public-key (asymmetric) cryptography to sign messages. And credentials are sets of a few attributes, which can be different based on implementation [25]. In the context of the TSS applications, and using the IRMA implementation, a National Statistical Office (NSO) would have to collaborate with an official institute which can verify and issue attributes such as identity, age, and so on. A municipality could be such a trusted issuer. Schematically, such a process is illustrated in figure 1.

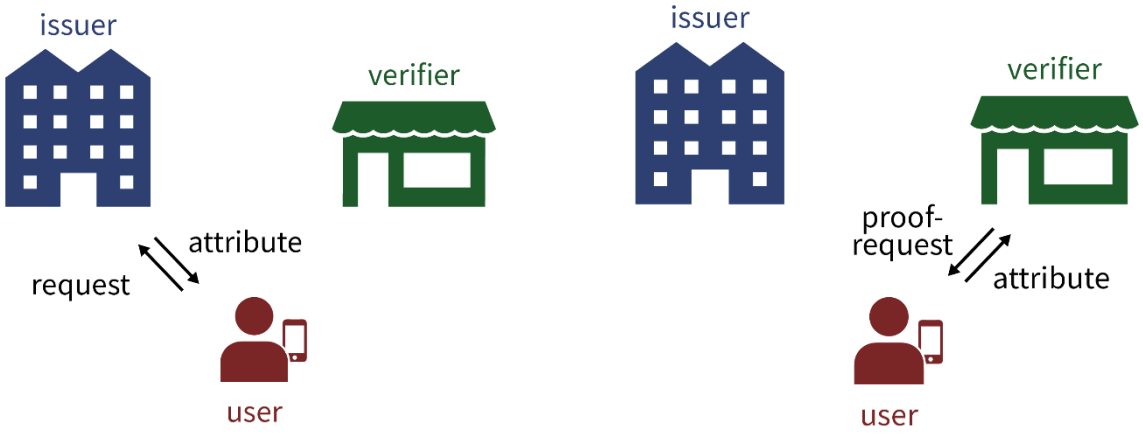

Figure 1: Using the issuer's digital signature over the attributes the verifier can verify that the attributes were given to the user in the past, and that they have not been modified since.

4.1. Technical Overview

The protocol uses a structure of three participants, and public-key cryptography for verification of messages related to attributes. Below is a list of participants, looked at from a TSS perspective, and with the according IRMA terminology [26].

→ This could be a user of a TSS application.

- *IRMA app: (mobile) application that receives attributes, and can disclose them. Also called client as it acts as the client in the IRMA protocol.*

→ A Statistical Office, wishing to verify identities of people who use a TSS app.

- *Verifier or service provider: a party wanting to verify someone's attributes (in order to provide some service).*

→ The 2 participants below could both be a municipality, in a context such as that of the Netherlands, or any entity that is legally entitled to issue and verify identity-related attributes.

- *Issuer or Identity provider: a party wanting to issue attributes to someone.*
- *Issuer: uses an Idemix private key in order to issue credentials to a client, when instructed to by an identity provider*

→ Both a statistical office (provider of a TSS app) as well as a municipality (or other identity verification & issuing institute) can act as a requestor.

- *Requestor: the service or identity provider that wants to, respectively, verify someone's attributes or issue attributes to them.*

Although an implementation of the IRMA credential system might look somewhat different in a TSS and NSO setting, the general flow of an IRMA session is depicted in figure 2.

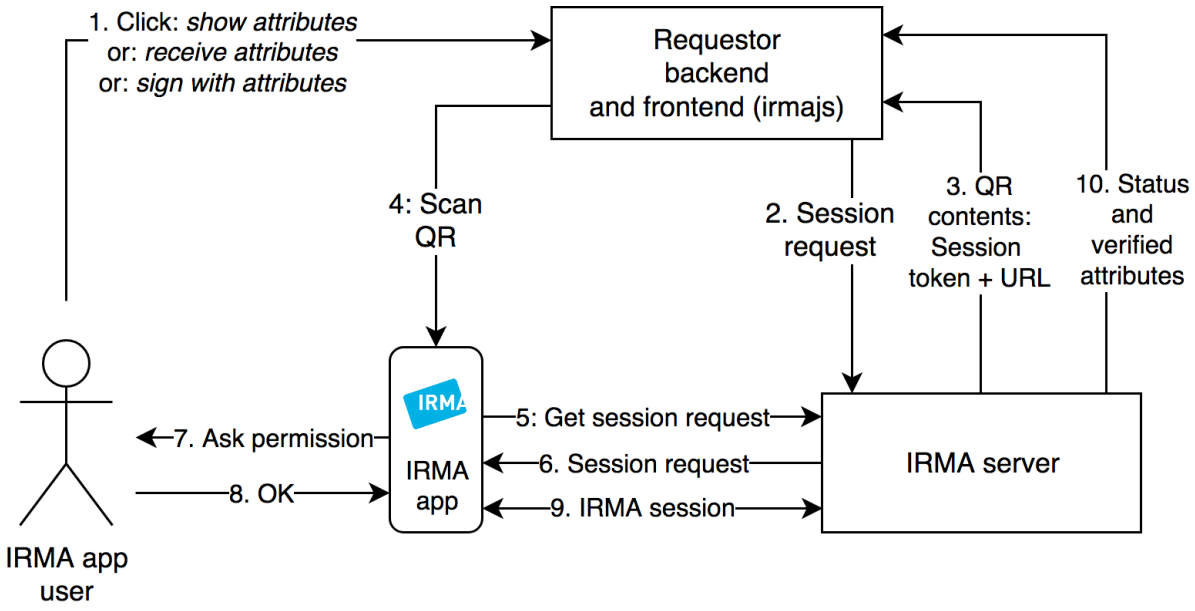

Figure 2: General flow of IRMA session[26]

The protocol uses cryptography, and as such there are a few different entities it distinguishes.

In the below specification, per aspect the roles within the TSS context might be:

- An NSO (providing a TSS) would be involved in 1) and 2) viewing them for verification of a user's identity and other information. The verification within 4), 5) and 6) is also used by the NSO.
- An issuer, likely to be a municipality, would have as its main goal to provide in aspect 1) and 2), by using 4), 5) and potentially 6).

1) *Attribute: a small piece of data, generally containing a statement about the attribute owner (e.g., '> 18 years old').*
2) *Credential: a group of attributes, jointly signed by the issuer using an Idemix private key, in an interactive protocol (called the issuance protocol) between the issuer and client.*
3) *Credential type: each IRMA credential is an instance of a credential type, which determines the names of the contained attributes, its validity period, and by which issuer the credential is issued.*
    - *Singleton credential type: users can store at most one instance of such credential types in their IRMA app.*
4) *Idemix private-public keypair: a pair of related keys:*
    - *Idemix private key: used by the issuer to sign a credential in the issuance protocol.*
    - *Idemix public key: used by a verifier when attributes are disclosed to it, in order to establish that the disclosed attributes have been signed using the corresponding Idemix private key.*
5) *Disclosure proof: a set of disclosed attributes, along with a proof of knowledge showing that these disclosed attributes originated from a credential that was validly signed by the issuer.*
6) *Attribute-based signature: a digital signature, with IRMA attributes cryptographically attached to it, on some document or message.*

An important element in the above list, is the idemix protocol, which uses three main elements: a user, an issuer and a verifier.

It provides authentication in a privacy-preserving manner, most importantly meaning it's anonymous. The idemix *private-public keypair is shown in figure 3[25]*

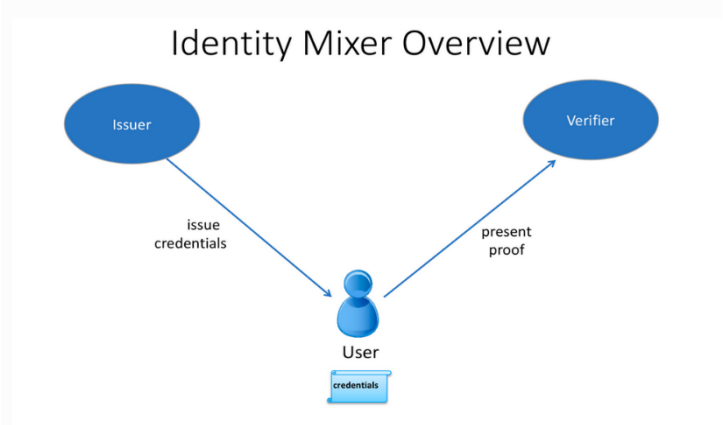

Figure 3: Idemix protocol

4.2. Implementation within TSS context

An NSO, or TSS provider, might consider two general implementations of the IRMA protocol, which might also be used together simultaneously. One option is to use an adapted version from the IRMA protocol to verify the identity of participants. A second option is to use Attribute-based signatures to sign surveys, as if they were documents or messages. These signatures contain a verification step of the identity of the user of the app, and might be less burdensome to implement.

The two options might provide an advantage if used simultaneously in that it may provide extra layer of verification, not only of identity but also of each survey. However, this may be too burdensome on the app and provide only a limited amount of added security.

Option 1 – General implementation of IRMA

In a straight-forward way, one could consider using the IRMA protocol for understanding the identity of a user.

1. A given NSO wants to use verification for a TSS app, and collaborates with a municipality to provide identity verification.
2. The municipality becomes the issuer of this identity verification, and it can a) verify a users' identity, as well as b) sign this using cryptography, specifically a private key.
3. The NSO implements a verification layer, based on a public key, provided by the issuer.
4. A potential user wants to participate in a TSS using its app, and wishes to get their identity (including some other personal details) verified. It may be assumed that identity verification is only needed once, during the installation of the TSS app.
5. The user installs the app and a few steps of regarding verification follow. These may vary depending on the intended implementation and technical restrictions within a TSS app. They involve the user of the TSS app and the IRMA server. Generally, the steps should follow this order:
    a) The installation of the app, and its user, are now functioning as the 'requestor' which sends its session request (including the attributes to be disclosed, or message to be signed) to the IRMA server.
    b) The server accepts this and assigns a session token (being a random string) to it.
    c) The implementation of the IRMA protocol requests the session request, receiving the attributes to be disclosed or issued, or message to be signed. And the IRMA server returns the session request.
    d) Optional, but a possibly useful addition from a PbD perspective: the IRMA app shows the attributes to be disclosed, or message to be signed, to the user, and they can decide whether to proceed.
    e) If the user accepts, the IRMA server performs the IRMA protocol, issuing new attributes to the user, or receiving and verifying attributes from the user, or receiving and verifying an attribute-based signature made by the TSS app.
    f) The session status (DONE, CANCELLED, TIMEOUT), along with disclosed and verified attributes or signature depending on the session type, are returned to the requestor.

Option 2 – Attribute-based signatures

Either separately or in addition to the first option, NSO's could consider implementing Attribute-based signatures, which are used for signing documents and messages. This could be considered if the general implementation of the attribute- or credential-based verification is not suitable or too burdensome.

In the TSS context, a survey might be seen as a document or message. *Attribute-based signature sessions: [27]*

Similar to disclosure sessions, but the attributes are attached to a message digitally signed into an attribute-based signature. The attribute-based signature can be verified at any later time, ensuring that the signed message is intact, and that the IRMA attributes attached to it were valid at the time of creation of the attribute-based signature.

4.3. General Considerations

There are few general things to consider, from legal, usability and technical perspectives, as listed below.

1 – Issuance of attributes

For the verification of attributes, a NSO needs to decide which attributes are most suitable to use. In order to be able to make this decision, it is important that for each credential type it is clearly documented how the attributes are obtained, and how it is ensured that they indeed belong to the person that receives them.

2 – The role of NSO and the IRMA server

The NSO should think about the role of a Scheme manager, which [27]:

distributes Idemix public keys, credential types and issuer information to clients and requestors; also decides which issuers may join its domain and what credential types they may issue.

Also, the role of the participants and the IRMA server should be considered, including its potential legislative restrictions, potentially differing per country.

3 – IRMA pincode protection

There is a specific verification step from the user's perspective, pincode protection. It should be considered in which step this could implemented within the TSS app, to provide ease of usability, and not to conflict with possible other pincode protection steps used within this app.

4 – Implementation and choice of language

It may be the case that different NSO's use different programming languages for their TSS apps, and as such also for the implementation of the IRMA protocol.

In general, it may be useful to consider that there is a Go implementation of IRMA, called GABI

5 – Generally, it can be good to take a look at IRMA's (advanced) security properties [28]

5. Conclusion and Outlook

Given the increasing production and collection of data in modern society, concerns with respect to the privacy risks involved will continue to grow. Considering the depency of NSO's on the willingness of the public to share their data, there should be a desire to take measures for the protection of privacy of respondents. In the case of smart surveys, which aim to leverage the wealth of information present in smart device sensor data, this consideration is even more apparent, given that the greater the amount of information present in

a dataset, the greater the possible harm when the is used for the wrong reasons. In the development of smart surveys, privacy should thus be an important consideration, not an afterthought.

In particular, we propose that development of smart surveys should follow the privacy-by-design principles described in section 2. Application of these principles in the context of the TSS app leads to three guidelines that should be considered when such an app is developed. Following these guidelines would make sure that respondents remain in control of their data, have a clear understanding of what their data is used for and allow them to selectively change which of their data is shared.

The measures taken when following these guidelines do not however, protect respondent data from being exposed whenever a leak occurs. Classically, protection from such an event is provided by anonymization and encryption. However, for statistical analysis, exposure of individual records is required at some point. Given the wealth of information that might be extracted from sensor data, respondents might not be comfortable with having these data exposed, be it within the confines of the NSO. Also, abiding by the privacy-by-design principles implies that NSO's should do everything in their power to protect the privacy of respondents. So, even if the respondent is willing to share all their sensor that, the NSO should not want to collect more information than is absolutely necessary. In fact, NSO's should strive to produce statistics without ever gaining access to individual respondent data.

Advanced privacy preserving techniques, like secure multi-party computation and homomorphic encryption, could make this a reality and should therefore be considered to be an important part of the future of smart surveys. However, implementation of these techniques is technically involved due to computational complexity. Application of these techniques to complex high-dimensional data originating from smart device sensors is hence not feasible. We therefore propose that application of PPT should involve two steps: (1) local pre-processing on the device, to reduce dimensionality of the data and complexity of subsequent computations and (2) private computation to produce aggregate statistics, using low-dimensional output of local pre-processing. This way, computational complexity of analysis is an issue only when privacy is not, while privacy is merely an issue when computational complexity of analysis is not.

Of course, whether this approach is feasible, depends on the amount of pre-processing that can be done on the device, as well as the efficiency of a given PPT. At this time, Secure Multi-Party Computation seems the most practical solution for application in the TSS context. However, this technique requires an architecture in which there are multiple trusted computing nodes. Various scenarios could be imagined, but we leave it for further investigations to make this more concrete.

The final privacy issue in the design of smart surveys applications regards authentication of users and verification of user identity. Ideally, identity of the respondent should be verified using as little information as possible. The 'IRMA' authentication framework provides a solution for this issue. In the Netherlands, this framework could already be implemented. Users can add certain attributes, like age and gender, to their digital passport by connecting with various institutions. Say a survey requires, in addition to some sensor data, merely age and gender of the participant, using IRMA then allows the respondent to share only their age and gender, without having to reveal their whole identity.

While IRMA is available internationally, there is a lack of institutions outside the Netherlands who can provide verification of useful attributes. Whenever this situation changes, it will probably be worthwile to look into the application of IRMA to allow for attribute based verification of user identity. Perhaps, NSO's could even try to contribute to the process of IRMA becoming a viable authentication framework outside the Netherlands.

References

1- ESSnetSmart SurveysDIME/ITDG Steering Group, Luxembourg, 12 February 2020 https://ec.europa.eu/eurostat/cros/system/files/07_-_essnet_smart_surveys.pdf

2 - Privacy by Design, The 7 Foundational Principles Implementation and Mapping of Fair Information, 2011, Ann Cavoukian;https://iapp.org/media/pdf/resource_center/pbd_implement_7found_principles.pdf

3 - How To Break Anonymity of the Netflix Prize Dataset, *Arvind Narayanan, Vitaly Shmatikov*, https://arxiv.org/abs/cs/0610105

4 - Simple Demographics Often Identify People Uniquely, *Latanya Sweeney*

5 - The Algorithmic Foundations of Differential Privacy, *C. Dwork, A. Roth*, Foundations and Trends in Theoretical Computer Science. Vol. 9, 2014

6 - Warner, S. L. “Randomised response: a survey technique for eliminating evasive answer bias", Journal of the American Statistical Association. March 1965

7 - www.tensorflow.org

8 - www.github.com/IBM/differential-privacy-library

9 - www.github.com/google/differential-privacy

10 - Differential Privacy Team (December 2017). "Learning with Privacy at Scale". *Apple Machine Learning Journal*. **1** (8)

11 - Gaboardi, Marco; Honaker, James; King, Gary; Nissim, Kobbi; Ullman, Jonathan; Vadhan, Salil; Murtagh, Jack (June 2016). "PSI (Ψ): a Private data Sharing Interface"

12 - Ding, Bolin; Kulkarni, Janardhan; Yekhanin, Sergey (December 2017). "Collecting Telemetry Data Privately". *31st Conference on Neural Information Processing Systems*: 3574-3583. arXiv:1712.01524. Bibcode:2017arXiv171201524D

13 - Machanavajjhala, Ashwin; Kifer, Daniel; Abowd, John; Gehrke, Johannes; Vilhuber, Lars (April 2008). "Privacy: Theory meets Practice on the Map". *2008 IEEE 24th International Conference on Data Engineering*: 277–286. doi:10.1109/ICDE.2008.4497436. ISBN 978-1-4244-1836-7

14 - McSherry, Frank (25 February 2018). "Uber's differential privacy .. probably isn't"

15 - Lyu, Min; Su, Dong; Li, Ninghui (1 February 2017). "Understanding the sparse vector technique for differential privacy". Proceedings of the VLDB Endowment. **10** (6): 637–648. arXiv:1603.01699. doi:10.14778/3055330.3055331

16 - Mironov, Ilya (October 2012). "On Significance of the Least Significant Bits for Differential Privacy" (PDF). Proceedings of the 2012 ACM Conference on Computer and Communications Security (ACM CCS). ACM: 650–661. doi:10.1145/2382196.2382264. ISBN 9781450316514.

17 - www.github.com/Microsoft/SEAL

18 - www.github.com/homenc/HElib

19 - www.palisade-crypto.org

20 - www.github.com/snucrypto/heaan

21 - https://www.tensorflow.org/federated

22 - https://paddlefl.readthedocs.io/en/stable/

23 - https://github.com/OpenMined/PySyft

24 - What is IRMA? https://irma.app/docs/what-is-irma/

25 - IDEMIX, https://hyperledger-fabric.readthedocs.io/en/release-2.2/idemix.html
26 - IRMA Session Flow, https://irma.app/docs/what-is-irma/#irma-session-flow
27 - IRMA Technical Overview https://irma.app/docs/overview/
28 - https://irma.app/docs/overview/#irma-security-properties

Task 3.1.5 – Incentive Schemes

Version: February 2021

Prepared by:

……..

Nils Meise (Destatis, Germany)

Task leader:

Nils Meise (Destatis, Germany)

Outline

Executive Summary

The activity of Task 3.1.5 aims at exploring incentives for Trusted Smart Surveys. Manual data entries can be cumbersome for respondents and potentially lead to missing entries and drop outs. This activity investigates how gamification could be used as a strong incentive scheme to engage respondents more in the data collection process.

It consists of the following parts: First, an overview of common incentives schemes (chapter 2) and an introduction into gamification (chapter 2). Second, how gamification can be applied to surveys (chapter 4) and how to tailor it to specific groups of respondents. And third, a discussion of implementation (chapter 6) and risks (chapter 7) of gamification. The summary (chapter 8) concludes with the next steps in line of this activity.

1. Introduction

Surveys are dull, offer no particular benefit for a respondent and – if there is no penalty involved – easy to quit. This is particularly true for voluntary surveys that require respondents to follow a repetitive regime of categories to answer over the course of multiple days or even weeks (e.g. TUS or HBS). To engage people in surveys is by increasing the benefits and at the same time decreasing the costs (or burden) of participation for which e.g. Dillman et al. (2014, pp. 28–32) collected the following criteria to achieve this goal:

- "specify how the survey results will be useful
- ask for help or advice
- ask interesting questions
- utilize sponsorship by a legitimate organization
- stress that opportunities to respond are limited
- convey that others have responded
- use cash and material incentives to encourage (but not require) reciprocity
- recognize that benefits both have additive effects and can reinforce one another
- do not deny the existence of benefits"

All these criteria aim at the intrinsic motivation of a respondent in order to participate in a survey. However, they do not fully cover burdens introduced by participating in the survey itself. Even users that are highly motivated and support the goals of a survey do drop out, because they do not stay engaged. A powerful incentive to keep respondents engaged in a survey or how to engage people into activities is to gamify them. Where gamification "[…] is used to describe those features of an interactive system that aim to motivate and engage end-users through the use of game elements and mechanics." (Seaborn and Fels 2015). In short, to apply game mechanics to non-game settings.

Having a fun experience and staying engaged in an activity is not a trivial task to accomplish. Even games can fail or get uninteresting over time. Thus, a growing community of designers and entrepreneurs entered the field and publish on implementations of gamification, how to create gamified settings and moral hazards involved (being engaged vs. being addicted) (Chou 2016; Goethe 2019; Marczewski 2018). Gamification is used in a applied sense to accomplish personal goals in life (Chou 2016) or in growing numbers in educational settings (Smith 2017). Surveys and in particular online surveys are in the scope of gamification (Harms et al. 2015; Oliveira and Paula 2020) and research shows that gamified surveys have benefits in regard to "[…] user experience, motivation, participation, amount and quality of data" (Harms et al. 2015, p. 219).

Trusted Smart Surveys offer the potential to include gamified parts or full gamification. Especially mobile devices are a gateway for smart data and questionnaires are prone to that implementation. Using mobile devices with gamified survey apps fits the every-day use of mobile devices by respondents, because it is similar to using other apps with gamified elements. However, not only context, also users are important. Not all mechanics appeal in the same way to people, because user preferences and personality does matter (Tondello and Nacke 2020; Triantoro et al. 2020). Trusted Smart Surveys should be engaging, but not addictive in nature.

A proper implementation of gamification as a strong incentive scheme for Trusted Smart Surveys needs to consider different user types or initially developed personas to appeal to respondents. Where a "[…] persona is description of a fictitious user" (Nielsen 2019, p. 2) that allows to develop a product or in this case a survey for a specific target audience. The task inventories gamification based incentive schemes and suggestions for implementations based on personas.

This part of the deliverable is structured as follows: first, we elaborate alternatives to gamification and why gamification might fit trusted smart surveys well. Second, we give an introduction into gamification. Third, we discuss the gamification of surveys. Fourth, we focus on user types and personas as a proxy for respondents' personalities. Fifth, we provide outlines for an implementation of gamification in trusted smart surveys and underline particular challenges that these types of surveys have. Sixth, will be a short assessment of gamification related risks. We will close with a summary and guidelines for best practices when it comes to incentives applicable for trusted smart surveys.

2. Incentive Schemes for Trusted Smart Surveys

Why should trusted smart surveys or any survey be gamified? Aren't there alternative incentives to consider? Anyone who ever conducted a (voluntary) survey experienced the problem that return rates are usually low. The introduction already laid out the filed by discussing the burden of surveys and criteria to increase the benefit. A blunt action taken for a respondent's benefit is a monetary incentive, which can be *prepaid* or *promised*. Hence, independent or dependent on actual participation in a survey. Also, a combination of prepaid and promised monetary incentives are a common practise. Additional common incentive schemes are listed in table 1 below.

Table 12: Common Incentive Schemes

Incentive	**Description**
monetary incentive	cash, money transfer, gift card
coupon or discount	Discount for a product or service of the survey agency.
free sample	Access to a free product or service. Often part of the survey
giveaways	Free branded items from the survey agency. Like notebooks, pencils, etc.
charitable donation	Survey agency donates to a charity for each returned/ finished survey
raffle	Each respondet enters a raffle upon completing the survey.
access to survey results	Respondents get mandatory access to survey results or reports.

Despite *access to survey results* all listed incentives require additional action by a survey agency that is out of scope of the content of the survey.[27] Especially finding suitable ways to transfer money or voucher can be tricky for some agencies.

Therefore, we suggest to use the main resources of a survey to build an incentive. First, the application itself, which can be altered to make the interaction with the active parts of a survey more enjoyable for a respondent. Second, the data actively collected by user entries or passively collected by sensors. Surveys that combine active and passive data collection depend on an application and that application delivers a gateway to allow not only an access to (preliminary) survey results, but to put individual results into a context during the survey. These contextualised, aggregated and shared information not only offer an additional insight, but can also be used to gamify elements of the survey. Further, they could offer an insight for the respondent how the passively collected data is used.

In the next chapter we take a general perspective on gamification. We will come back to the issue of passive and active data collection in gamified surveys when we discuss the actual implementation.

3. Gamification

In our context we do understand gamification like Marczewski (2018) as "[t]he use of game design metaphors to create more game-like and engaging experiences". This definition tells us that we can learn from games to create an experience that feels like a game, but actually results in a setting that is engaging for a user. Focussed attention is also found in other definitions like those of Seaborn and Fels (2015), who argue that gamification "[…] is used to describe those features of an interactive system that aim to motivate and engage end-users through the use of game elements and mechanics." Or by referring to game like experiences as something that influences the behaviour of a user and gamification is the way to design such systems (Triantoro et al. 2020). It could also be argued that this focussed attention is similar to what is described as *flow* or *flow state* in psychology. This kind of mental state does not only include a focus on an activity, but also a positive attitude towards the activity itself. Flow experiences could even be considered the key element for a successful implementation of gamification and provide the necessary motivational support to stay engaged in the (gamified) activity (Blohm and Leimeister 2013, p. 278).

Game elements and mechanics are the building parts of a game – or simply the rules of what players can do or not do and direct or indirect consequences of their actions. To put it in a more operationalized way, game mechanics are "*[a] distinct set of rules that dictate the outcome of interactions within the system. They have an input, a process and an output*" (Marczewski 2018, emphasis in the original). Any game design works within these boundaries and gamification is the hard task to not only apply new rules to a former non-game setting, but also rather alter existing rules to be in line with a gamified experience or vice versa. An overview of common game design elements collected by Deterding et al. (2011, p. 12) is listed in Table 13. What is important to note is that although game design elements are at the core of gamification, it is not the goal to design a game, but gamify a non-game setting.

Table 13: Levels of Game Design Elements

Level	**Description**	**Example**
Game interface design patterns	Common, successful interaction design components and design	Badge, leaderboard, level

[27] An exception to this would be free samples of a product that is a part of a survey. In other words: a product test.

	solutions for a known problem in a context, including prototypical implementations	
Game design patterns and mechanics	Commonly reoccurring parts of the design of a game that concern gameplay	Time constraint, limited resources, turns
Game design principles and heuristics	Evaluative guidelines to approach a design problem or analyze a given design solution	Enduring play, clear goals, variety of game styles
Game models	Conceptual models of the components of games or game experience	MDA; challenge, fantasy, curiosity; game design atoms; CEGE
Game design methods	Game design-specific practices and processes	Playtesting, playcentric design, value conscious game design

Source: Deterding et al. 2011, p. 12

At first glance, gamification looks like something that depends on or needs technology to be used. Which, on one hand is true, because information technology delivered the tools to gamify the use of products and services (Blohm and Leimeister 2013, p. 276), but on the other hand gamification is also a mind-set to approach all kind of tasks in our daily life (Chou 2016). The question at hand is, if games can be found everywhere, what kind of games are we talking about? Games are commonly just perceived as some sort of structured activity that is defined by a fixed set of rules, goals to accomplish and a competitive element for leisure or recreational activities. It stands in opposition to play, which is an unstructured activity that is not bound by fixed rules or a goal. Categories of interest for gamification are what are called *serious games* and *games with a purpose*. Serious games are games designed for other purposes than entertainment. Usually, these games are designed to meet a given learning objective (Blohm and Leimeister 2013, p. 277). Objectives are based on educating an audience on a particular topic, but in a more fun and competitive way than simulations would do. Games with a purpose are a category of games that use human interaction to solve problems that are trivial for humans, but hard for computers to solve, which often is about classifying data (Ahn 2006; Ahn and Dabbish 2008). One recent example is “Project Discovery,” a mini-game to categorize cells for Covid-19 research that was rolled out as part of the online game “Eve Online.” The users were highly engaged and produced an amount of completed categorizations that would have taken decades to complete otherwise. Beside gamified categorization, also in-game incentives were part of the project. Other problems that were addressed by games with purpose in the past are by now trivial for machines too, due to advances in computational power and machine learning. An example for this are also CAPTCHAs, which are set up to ensure that an actual human and not a machine is interacting with a website. Simple text based CAPTCHAs are no challenge for a machine anymore and they make less errors than humans do. Current CAPTCHAs, which are often image recognition oriented, do not only try to verify a human operator, but also train Googles image recognition algorithms, which makes them a dual-use game. CAPTCHAs also show us that a game-like mechanic – find images that contain a certain thing, select them and sent your results to reach your goal of sending your initial request to the server – could also be a new burden and even have an opposite effect.

Gamification of smart surveys falls into hybrid category of serious games and games with a purpose. This has consequence on how the content and goals of such a gamified survey are communicated. We will address this issue later, but first focus on the gamification of surveys as our field of interest.

4. Gamification of Surveys

Surveys are dull. We already made this assumption earlier, but what does it actually mean and is it true? To fully answer that question, we also need to look at our users, which we will do in the next chapter, but first we will look at the particular elements of gamification and surveys. First of all, gamification should make responding to surveys an emotionally appealing and fun activity (Dillman et al. 2014, p. 24). However, gamification is not solely about fun or creating a flow state for the user as we discussed already, it also has an impact upon how a survey and the survey organization is perceived by a respondent. Triantoro et al. (2020) did a study on the impact of gamification on online surveys and show that gamified surveys: "[…] drive attention and enjoyment, and subsequently increase the attractiveness of a surveying organization." Harms et al. (2015, p. 219) do also point out that gamification leads to an improved accuracy of survey results. These promising advantages come with new costs, because a gamified survey presses for a different look and feel. Harms et al. (2015) also tested a gamified survey that was created by transferring a legacy survey into a gamified format. The result was a complete visual overhaul, new elements of choice that were only implemented to individualize the experience for the user and customization options for look and feel of the survey. The benefits of gamification came with an increase of cost in UI design, prototyping and testing (Harms et al. 2015, p. 227).

Choices do matter, they have consequences and shape how we perceive or frame an activity. Gamification is also storytelling, because its mechanics show us that our choices do matter and we are told why and how. Not only for the greater good of participating in providing value data for important choices that will impact whole societies, but also in a smaller scale, if we do not get our reward for staying engaged in a survey every single day. Having a reward or not, making one or another choice does not necessarily mean actual consequences for a user. Though, it is important that a user feels that choices have a meaning, but not necessarily that the outcome changes based on choices made, it is the user's experience that changes due to choices made (Marczewski 2018). Story based items and other game design decisions are giving problems to the user to solve – the activity is enriched and given purpose, fun, or an interesting challenge.

In order to successfully gamify our surveys, we will look at our respondents or to use a more general term – our users – next.

5. Gamification with a Focus on Respondents

5.1 User Types

A common technique in software development to get an idea about typical users and their way of interacting with the product is thinking of user types or personas. There is no one size fits all and how users might use a product can be vastly different from initial design ideas. We will start our journey by looking at a scheme for user types by Marczewski (2018) that was developed for the use in gamification and which was also already put to an empirical test by Tondello et al. (2019). Marczewski developed six basic user types: *socialiser*, *free spirit*, *achievers*, *philanthropists*, *player*, and *disruptors* in a *User Type Hexad* (Figure 12). The figure does also show the main motivation of that user type while being engaged in the game or their willingness to play. *Players* are in the game for the reward (Marczewski 2018). For example, they participate in a gamified survey to get the monetary incentive and will complete all tasks to achieve this goal. *Disruptors* don't want to play, if they are involved in a game they want to change it, disrupt the system, which drives their motivation (Marczewski 2018). Such a user in a gamified survey would e.g. give false information or even motivate others to also give false information. A disruptor could also be someone, who gives socially desired answers and thus disrupting the survey. *Philanthropists* are motivated by purpose, they want to give to other people and expect no reward (Marczewski 2018). Abstract goals of civic research, bettering society, and a general call for volunteering in surveys, which aim at giving dependable information for informed political decisions, may resonate with this type. *Achievers* are motivated by complete mastery off all challenges thrown at them and tokens that show their accomplishment are not sufficient, though welcome (Marczewski 2018). Completion of tedious daily tasks are no problem for this user type, if the setting is right for them. *Socialisers* are looking to create new social connections and are motivated by systems that promote that (Marczewski 2018). *Free Spirits* are motivated by autonomy, which means system that allow to exploration or ways to express creativity (Marczewski 2018). This user type might be the hardest for gamified surveys as surveys favour closed questions and exact answers. Appeal for this user type must come from the whole interaction experience with a survey app.

Figure 12: User Type Hexad

Source: https://www.gamified.uk/user-types/

Marczewski categorizes the user types into two groups: *intrinsic* and *extrinsic* user types. Intrinsic user types are the philanthropist, achiever, socialiser and free spirit. Extrinsic user types are players and disrupters. It is important to note that user types are not fixed and game mechanics could and should allow a transition of user type. Especially if motivation and engagement should be placed on a stable base: "The Player User Type is important to recognise, as most people coming to a gamified system are likely to enter initially due to rewards (points, prizes, etc.). The trick is to try to convert them from being reward oriented into intrinsically motivated users [...]" (Marczewski 2018, p. 107). To translate such shift to surveys could mean to attract survey respondents with a monetary incentive (a prize), but it is crucial to shift their experience to be intrinsically motivated to stay engaged.

5.2 Personas

Another way to think about the potential users of an application or in our case a survey, are personas. Where user types are rather set categories and one has to figure out in which categories one's users would fit, personas are closer representations of actual users and unique for their use case. Nielsen (2019, p. 2) defines them as follows: "[…] a persona is a description of a fictitious user. A user who does not exist as a specific person but is described in a way that makes the reader believe that the person could be real. A persona is based on relevant information from potential and real users and thus pieced together from knowledge about real people". Usually, personas are based on data gathered from a domain, but some models also go for personas based on designer's assumptions.

How many personas are required for a project depends on how different the potential users are and whom a project is targeting, which further emphasizes the role of personas as a *design tool*. However, more than six personas are often not feasible, because it might be difficult to remember all details about all personas within a given project (Nielsen 2019, p. 9). Nielsen (2019, p. 12) came up with a process model that consists of four parts: a data collection and analysis part, a part of persona descriptions, a part on creation of scenarios for problem analysis and idea development, and a part on acceptance of the personas within the organisation and involving the actual design team.

5.3 How to decide on User Types or Personas

Using user types or personas is not specific for gamified surveys. They are a usual tool for software development and when thinking about designing web or app based surveys. The challenge for gamified surveys is to declare the gamified setting as the design principle and investigate the appeal for different personas.

Trusted Smart Surveys are thought of with an empowered user base in mind. A public that actively participates in the production of official statistics to greater the good for the whole society. When considering user types, it becomes clear that this is just one of several possible ways of engagement. Also personas are thought from different perspectives, which allows for more ways of creating an engaging survey. Though, even in this short overview it becomes apparent that extra work from the ground up or in the direction of gamification is needed, if it is to be employed in a Trusted Smart Survey.

In the next chapter, we will shift our focus from users to context. Trusted Smart Surveys offer a new potential to include a wide variety of game mechanics, but also ethical, legal and technical boundaries.

6. Implementation of Gamification in Trusted Smart Surveys

One of the main guiding principles when implementing gamification in Trusted Smart Surveys is that it must not violate the trust of the user. Trust is a crucial element for all surveys and by establishing trust it does also make people more engaged in surveys (Dillman et al., pp. 37–41). Gamification has the potential to violate trust and pose ethical problems, if it aims at manipulating users to adapt a certain behaviour or lock them in a product or activity, which users will get ware of and not just withdraw from an activity but also develop a negative opinion towards the organization responsible (Goethe 2019, pp. 32–33). By ensuring that the principles of a Trusted Smart Surveys are acknowledged, communicated, and correctly implemented, a Trusted Smart Survey already has a high potential for stronger engagement of users. Part of the trusted nature is that the survey is conducted by official government institutions and the results have an impact on political decisions. In short, official surveys are a sincere business. Any form of gamification used in Trusted Smart Surveys should not jeopardise the attitude towards the survey or the survey organisation. Thus, being

engaged in a Trusted Smart Survey should be a sincere, but fun driven enterprise, which is built on a foundation of trust.

Before starting to think about the actual implementation of gamification, there are a couple of questions that need to be answered. First, what do I know about my users? Second, what are my resources (time, personal, technical, etc.) that I can devote on developing a full or partial gamification of a survey? Third, does my smart survey consists of passive data collection only or does it heavily rely on passive data collection?

In an ideal situation, information about a user base could be gained e.g. via surveys, interviews or focus groups. This would allow for a detailed construction of personas and tailored gamification modules. In addition, an assessment of user types via a survey is a viable option and could be done with former participants of surveys that are planned to be gamified. Information on users could also come from available data and sampling of participants be done from an existing pool (e.g. participants of a panel study) like Kreuter et al. (2020, p. 4) did for study on smartphone sensor data for surveys. If no data is available, also assumptions about personas are a first start. The concept of personas is process oriented and future change part of the concept.

In survey settings that are completely build around passive data collection, gamification is harder to implement. Gamification builds around an engaging experience. The crucial question is, if the respondent is required to interact with the passive data collection process. If the respondent is required to use a wearable to passively collect data, a strong incentive is needed in order to nudge the respondent to use that wearable at all required times. If the passive data collection is done by devices that are just around in a respondent's home for a period of time, e.g. air quality sensors that automatically send their data to the survey agency, the incentive is not focussed on nudge the respondent to *do something* but instead to just not remove the devices. In short, surveys that need strong active engagement need stronger incentives.

Implementing incentive schemes falls into wider methodological design considerations of trusted smart surveys and of course the business architecture of official statistics. Incentive schemes and their use in the field have an impact on the official statistics business process. From a business layer perspective GSBPM of course comes to mind, but due to the wider scheme of big data sources, it makes sense taking the BREAL architecture into account, which fuses it all together. The BREAL business layer architecture targets at the whole big data lifecycle and the underlying business functions that would also cover incentives as part of the overall process (cf. Scannapieco et al. 2019). In particular, during *Acquisition and Recording* (BREAL) phase data production in the incentive-focussed modules as well and of course during *Deployment* (BREAL) as they may require new infrastructure that is not available at an NSI. One example for a missing infrastructure or unfamiliar process is the lack of data streaming capabilities to process data in real time. The data collected as part of the incentive scheme is not a part of the statistical data set that results from a survey, but it is part of the data collected to improve a survey or parts of it. If any data is stored, it should be stored separately from the statistical data and solely for the purpose to improve the inventive schemes. During or because of a production phase they have an impact on *Continuous Improvement* (BREAL) and thus on *Evaluate* (GSBPM) and *Quality Management* (GSBPM).

The whole process depends on an intertwined design and production process involving actors from statistical and IT domains. Other particular actors that should be involved are data protection officers to ensure that none of the extended data collection or analysis for incentives is violating any privacy or legal restrictions in place.

7. Risks of Gamification

The decision and the implementation of gamification is not free of risks. This part will discuss which risks are on the side of the survey agency, the respondent and the results of the survey.

The decision to gamify a survey holds risks for the survey agency that are by majority due to the fact that gamification is still uncommon terrain in the survey domain. A bad survey design wastes money and time. Especially as a gamified survey takes more time to design and program than a standard survey. There could be no needed experts available at the survey agency (e.g. UI and UX experts for gamified apps), non-realistic expectations about the impact of gamification on surveys or a general lack in IT infrastructure for required features (e.g. data streaming).

A gamified survey poses a risk for respondents as well. Game mechanics are used to make respondents stay engaged in a survey. Respondents might become too fixated on achieving a reward or status and neglect other duties. This could even lead to ethical problems, when the amount of engagement, competition and reward system is not managed right. In addition to a manipulate aspect, gamified surveys should be entertaining and not overly competitive. Where is the fun to continue of you are always last on the leader board?

In regard to survey results different new biases are introduced, which can be tight to particular user types or general game mechanics. Socially desired answers or completion of a survey with random answers to help a researcher or to complete a task is already problems survey agencies are dealing with. However, gamification aiming at *achiever* and *philanthropist* user types could be more prone to such problems. Even if a gamified survey does not aim to include a *disruptive* user type, such a user type has malicious intend and could temper with the passive data collection as well. A *player*, who is in for the reward, will try to maximize the reward. If such a reward depends on completion, it also introduces a risk of random biased answers for the sake of completion. In sum, the risk tied to user types are not unfamiliar, but need to be addressed during the design, testing and production phase of a survey. Passive data collection might be able to mitigate some of the risks. However, sensors and devices could also be tempered with or not used as intended.

8. Summary

Incentives make surveys a fun and engaging activity, thus reducing the burden for respondents. However, it should not turn into a huge burden on the side of the NSI, which has to allocate more resources of time and personal for design and during deployment. Even in a full-automated environment incentives push to further development of personas or adjustments of user types for future surveys. Also new experts in UI or HCI design are needed for a successful implementation.

Gamification is a proven tool to foster engagement in a given activity and widely used in different fields of business' and tested in online surveys and app-based scenarios. Also, respondents are used to gamification elements during leisure activities and increasingly in education and work settings. However, a wide range use of gamification is ethically challenging, because it aims at adapting to socially expected behaviour and rewards conformity (cf. Kim and Werbach 2016).

User types in gamified settings are an orientation for development and allow to adapt for different types of respondents not only in app design, but also in communication in order to appeal to specific audiences. But user types are a simplified abstraction of motivation only. We also discussed personas, which offer a similar variety of types with a more in-depth and focussed attention on specific and specified ideas of users.

Implementation of incentives fall into wider methodological considerations, bound to the business logic of the production of official statistics (GSBPM) and big data in official statistics (BREAL) and are a challenge for continuous development, integration, privacy and ethics. Some of these challenges we discussed in regard to particular risks of gamification.

The next steps in line of this activity are a Proof of Concept that will explore how gamification-based reports could be implemented as an incentive scheme for app based smart surveys and an integration of incentives schemes into the general smart survey methodology suggested in this deliverable. The goal of the Proof of Concept is to test what kind of gamified incentives can automatically be derived from survey data. Functional templates that use a gamified theme will be developed in the context of this Proof of Concept. Thus, it will also provide valuable information on how to mitigate some risks related to gamification that are mentioned above. However, it will neither cover respondents' interaction with gamified survey elements nor does it research potential biases due to gamification or incentives in general. The closer integration into the suggested smart survey methodology will be done in context of the improved conceptual framework for the European Platform for Trusted Smart Surveys.

References

Ahn, Luis von (2006): Games with a Purpose. In *Computer*, pp. 96–98.

Ahn, Luis von; Dabbish, Laura (2008): Designing games with a purpose. In *Communications of the ACM* 51 (8), pp. 58–67. DOI: 10.1145/1378704.1378719.

Blohm, Ivo; Leimeister, Jan Marco (2013): Gamification. In *Business & Information Systems Engineering* 5 (4), pp. 275–278. DOI: 10.1007/s12599-013-0273-5.

Chou, Yu-Kai (2016): Actionable gamification. Beyond points, badges, and leaderboards. Fremont, CA: Octalysis Media.

Deterding, Sebastian; Dixon, Dan; Khaled, Rilla; Nacke, Lennart (2011): From game design elements to gamefulness. In : Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments. Tampere, Finland, pp. 9–15.

Dillman, Don A.; Smyth, Jolene D.; Christian, Leah Melani (2014): Internet, Phone, Mail, and Mixed-Mode Surveys. The Tailored Design Method. 4.th ed. Hoboken: Wiley.

Goethe, Ole (2019): Gamification Mindset. Cham: Springer International Publishing (Human-Computer Interaction Series).

Harms, Johannes; Biegler, Stefan; Wimmer, Christoph; Kappel, Karin; Grechenig, Thomas (2015): Gamification of Online Surveys. Design Process, Case Study, and Evaluation. In Julio Abascal, Simone Barbosa, Mirko Fetter, Tom Gross, Philippe Palanque, Marco Winckler (Eds.): Human-Computer Interaction - INTERACT 2015. Cham: Springer International Publishing, pp. 219–236.

Kim, Tae Wan; Werbach, Kevin (2016): More than just a game. Ethical issues in gamification. In *Ethics Inf Technol* 18 (2), pp. 157–173. DOI: 10.1007/s10676-016-9401-5.

Kreuter, Frauke; Haas, Georg-Christoph; Keusch, Florian; Bähr, Sebastian; Trappmann, Mark (2020): Collecting Survey and Smartphone Sensor Data With an App. Opportunities and Challenges Around Privacy and Informed Consent. In *Social Science Computer Review* 38 (5), pp. 533–549. DOI: 10.1177/0894439318816389.

Marczewski, Andrzej (2018): Even Ninja Monkeys Like to Play: Unicorn Edition.

Nielsen, Lene (2019): Personas - User Focused Design. 2nd ed. 2019. London: Springer (Human-Computer Interaction Series).

Oliveira, Kayque Willy Reis; Paula, Melise Maria Veiga (2020): Gamification of Online Surveys. A Systematic Mapping. In *IEEE Transactions on Games*, p. 1. DOI: 10.1109/TG.2020.3004366.

Scannapieco, Monica; Bogdanovits, Frederik; Gallois, Frederic; Fischer, Bernhard; Georgiev, Kostadin; Paulussen, Remco et al. (2019): BREAL. Big Data Reference Architecture and Layers. Version 2019-12-09. Edited by EUROSTAT.

Seaborn, Katie; Fels, Deborah I. (2015): Gamification in theory and action. A survey. In *International Journal of Human-Computer Studies* 74, pp. 14–31. DOI: 10.1016/j.ijhcs.2014.09.006.

Smith, Tamarah (2017): Gamified Modules for an Introductory Statistics Course and Their Impact on Attitudes and Learning. In *Simulation & Gaming* 48 (6), pp. 832–854. DOI: 10.1177/1046878117731888.

Tondello, Gustavo F.; Mora, Alberto; Marczewski, Andrzej; Nacke, Lennart E. (2019): Empirical validation of the Gamification User Types Hexad scale in English and Spanish. In *International Journal of Human-Computer Studies* 127, pp. 95–111. DOI: 10.1016/j.ijhcs.2018.10.002.

Tondello, Gustavo F.; Nacke, Lennart E. (2020): Validation of User Preferences and Effects of Personalized Gamification on Task Performance. In *Frontiers in Computer Science* 2 (29). DOI: 10.3389/fcomp.2020.00029.

Triantoro, Tamilla; Gopal, Ram; Benbunan-Fich, Raquel; Lang, Guido (2020): Personality and games. Enhancing online surveys through gamification. In *Information Technology and Management* 121 (2). DOI: 10.1007/s10799-020-00314-4.

Task 3.1.6:
Metadata and Process auditability

Version: February 2021

Prepared by:

Raffaella Aracri (ISTAT, Italy)

Mauro Bruno (ISTAT, Italy)

Michele Karlovic Riccio (ISTAT, Italy)

Giuseppina Ruocco (ISTAT, Italy)

Task leader:

Franck Cotton (INSEE, France)

Outline

Executive summary

The main objective of task 3.1.6 is the design of the TSS_U platform metadata component, which is a challenging task that requires an in-depth analysis of several aspects, namely:

- statistical concepts related to the survey theme and objectives;
- collection instruments, variables and codes;
- statistical methodology used in data processing;
- process and lineage metadata;
- quality metadata.

Starting from a preliminary analysis of standards and frameworks (i.e., ontologies, BREAL Information Architecture, GSBPM and GSIM) related to metadata concepts, this document focuses on the following general aspects:

- Data structures
- Sensor data
- Process steps
- Privacy issues
- Generic functionalities for reuse in different domains
- Specific characteristics of national contexts.

The approach adopted to fulfill these general objectives is based on the following principles:

- Rely on standards and align with existing frameworks;
- Use active/passive metadata;
- Use-case-driven approach. The first use case selected for testing the metadata model is the Time Use Survey, combining GPS data and other variables (e.g., “type of activity”) collected through an app on

In order to overcome specific domain requirements, the main goal of this task is to model a repository with the minimum set of metadata for process standardization and application components reuse. The implementation stage should focus primarily on the description and integration of sensor and traditional data used in TSS_u.

1 TSS_U Metadata: objectives and design principles

The term 'Metadata' refers to many concepts and dimensions, interconnected and with specific features at the same time. In order to contribute to the TSS_U platform design, the main objectives of the Metadata task relate mainly to:

- developing a common semantic representation of the domain covered by the project;
- providing recommendations and guidelines regarding the capture of metadata all along the operation lifecycle. The whole set of metadata will cover:
  - statistical concepts related to the survey theme and objectives;
  - collection instruments, variables and codes;
  - statistical methodology used in data processing;
  - process and lineage metadata;
  - quality metadata.

This list of metadata relates to different dimensions: domain concepts, collection tools, regulations, methodology, output description, process tracking and quality.

The priorities established to investigate a subject so vast, focus on the following general aspects:

- Data structures
- Sensor data
- Process steps
- Privacy issues
- Generic functionalities for reuse in different domains
- Specific characteristics of national contexts.

The approach adopted to fulfill these general objectives is based on the following principles:

- Rely on standards and align with existing frameworks;
- Use active/passive metadata;
- Use-case-driven approach. The first use case selected for testing the metadata model is the Time Use Survey, combining GPS data and other variables (e.g., "type of activity") collected through an app on the smart phone.

As for the architectural task, the reference to standards and existing frameworks is essential for the compliance to metadata reporting in official statistics. Furthermore, this alignment avoids overlapping between different metadata systems and the reuse of available metadata. The preliminary analysis of metadata concepts is based on the following standards and frameworks:

- Ontologies
- BREAL Information Architecture
- European structural metadata and quality framework
- GSBPM process description
- GSIM structures

Due to the combination of different reference frameworks, the proposed approach is intended mainly for readers who are used to deal with metadata concepts and related standards.

In order to overcome specific domain requirements, the main goal of this task is to model a repository with the minimum set of metadata for process standardization and application components reuse. The implementation stage should focus primarily on the description and integration of sensor and traditional data used in TSS_u.

2 TSS_U Metadata repository

Following an empirical approach, the design of the Metadata repository has started from a bottom-up perspective, to facilitate defining the set of metadata concepts for TSS platform. At this stage, the repository is designed as a collection of relevant information for:

- data description and processing;
- process standardization, control and documentation;
- tools and methods inventory.

The whole set of metadata can be divided in different subsets:

- Domain models, related to survey main concepts and objectives.
- Generic TSS models, that refer to the common concepts related to TSS. This subset includes the Semantic Sensor Network Ontology, and the models (Information metadata model, Process metadata model) derived from the BREAL architecture.
- A core subset that contains the specification of variables, units, data structures, classifications.
- Methods and tools used throughout the statistical process and process specification.
- Monitoring and assessment of the different trust dimensions. Dealing with TSS, this subset groups all types of metadata that allow to assess Trust throughout the process.

While the content of the first subset is tailored to survey characteristics, the content of the other two groups can be standardized, in order to build generalized tools. The following figure provides an overview of the subsets, grouping several metadata according the above criteria. The Provenance dimension is overarching, due to its connection across all layers.

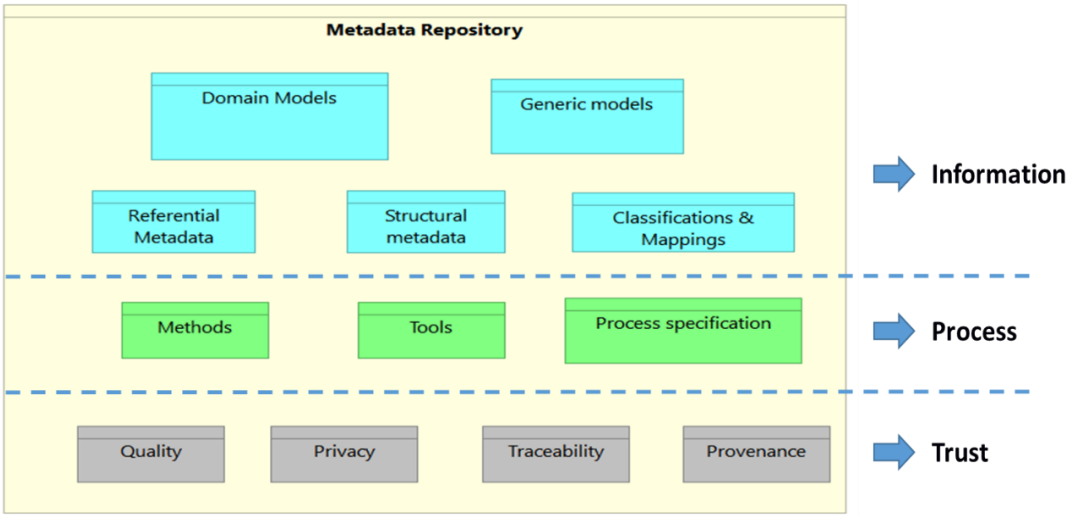

Figure 13: Conceptual description of TSS Metadata Repository

In order to achieve a conceptual overview of metadata, these subsets can be referred to three main areas: Information, Process and Trust. While the Trust issue is overarching, connected to every aspect of the statistical process, the Information area includes the blue subsets in the figure. These objects concern the whole set of information treated throughout the process. Some of these objects, like Referential metadata or Classifications and mappings will be modelled according to the official standardized European systems, so for these subsets no further exploration is needed. The analysis for this area will start from the TSS Generic models. The third area (Process) groups all the activities performed to achieve a statistical output, including process specification and process execution (Traceability).

3 TSS_u Information model

The alignment with existing frameworks and tools has contributed to identify and describe the elements of each metadata subset. Particularly, ontologies classes have been very useful not only for the description of specific statistical domains, but also for specifying the items belonging to each metadata subset. At this stage, the focus is:

- inventory of the key metadata concepts, regardless any technical specification;
- modelling data transformations through the process.

To this aim, the following paragraphs focus on the Information and Process metadata models, starting from BREAL (Big Data REference Architecture and Layers). BREAL is the reference architecture fostering the development of standardized solutions and applications for Big data. While BREAL business layer describes the production process, the actors and stakeholders involved, BREAL Information layer deals with Big Data Life Cycle in terms of data models.

Assuming the BREAL Information architecture as reference standard for tracking data transformation, in the TSS data model, the data layers are specialized to describe sensor data sources. The benefits of adopting BREAL concern both the data perspective analysis, and the integration with other frameworks (namely, PROV ontology and GSIM) that support process and data standardization. The following figures report the TSS Information architecture adapted from BREAL. Particularly, the main data transformations of structured sensor data in motion across the three layers (Raw, Convergence and Statistical) have been modelled, using Archimate28 language. The description of each layer is based on the following data objects: (i) BREAL data entities (blue color); (ii) specific TSS data entities (green color); (iii) GSIM entities (pink color); provenance metadata entities (yellow color).

The starting point is the Raw layer, which is connected to BREAL "Acquisition and Recording" business function. The proposed model considers the information acquired by sensor data, corresponding to a specialization of Big data sources. For TSS, Provider activities become relevant only when they affect the tasks following data acquisition and performed in-house by NSIs, or when the application of a privacy preserving technique (e.g. Pushing computation out) is needed. In Figure 14, on the left side, BREAL: Acquired Entities is subclass of Prov: Entity, while BREAL: Acquiring Agents is subclass of Prov: Agent. Then, TSS: Acquiring Agents is subclass of BREAL: Acquiring Agents and provides TSS: Sensor Data source, and TSS: Acquired Entities is subclass of BREAL: Acquired Entities. The TSS subclasses, colour-coded in dark green and describing sensor data, derive from the SOSA ontology. SOSA is the core part of the Semantic Sensor Network Ontology. In the central part, TSS: Sensor Data source is subclass of BREAL: Structured, and BREAL: in Motion. Considering the compliance with GSIM, TSS: Sensor Data source is also subclass of GSIM: Data Resource.

[28] Archimate is an open and independent language for modeling enterprise architecture, available from: https://www.archimatetool.com/.

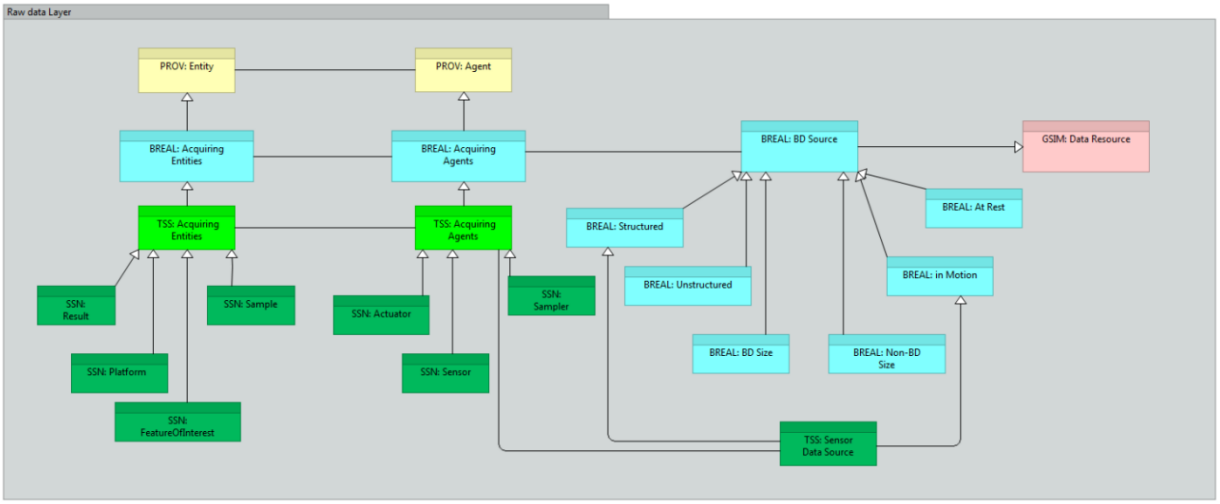

Figure 14: TSSu Raw layer

The Convergence Layer (see Figure 15) refers to acquired data, represented in terms of units of interest and variables for the following analyses. This layer is connected to BREAL business functions “Data Wrangling” and “Data Representation”. Analyzing the convergence layer in the figure below, TSS: Lower layer is subclass of BREAL: Lower layer, and TSS: Troughput activity is subclass of BREAL: Troughput activity. In the convergence layer, Troughput activities refer to all the tasks performed to process raw data, to identify unit data types and derive statistical variables.

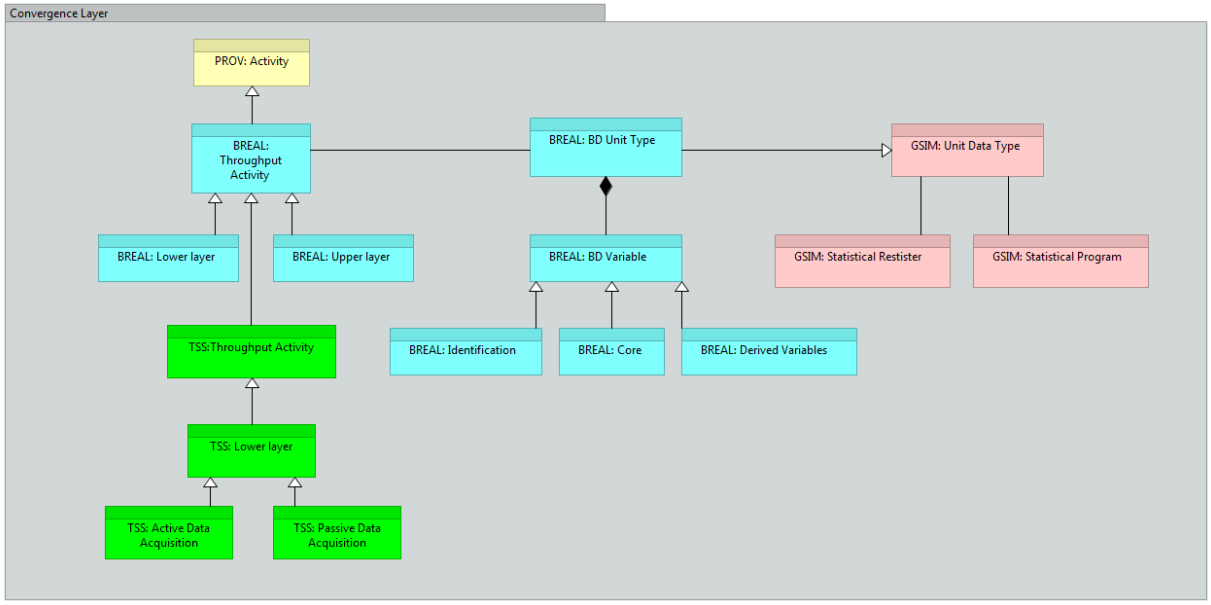

Figure 15: TSS$_u$ Convergence Layer

The Statistical Layer (see Figure 16) reports the core statistical concepts resulting from the analysis. From a process perspective, these data objects are connected to the following business functions: “Modeling and Interpretation”, “Integrate Survey and Register Data”, “Enrich Statistical Registers” and “Shape Output”. In the figure below, on the left side, TSS: Output Quality, TSS: Output Process Auditability and TSS: Output Privacy are specializations of TSS: Lineage of TSS Output Entities, that is subclass of BREAL: Lineage of Output Entities. On the higher level, BREAL: Lineage of Output Entities is a specialization of provenance metadata entities, providing an overview of the tasks performed from the output perspective. Considering the information achieved, TSS: Survey & Sensor Data Information Set is a specialization of TSS: TSS Information

Set, that corresponds to a subclass of BREAL: BD Information Set. TSS Information Set is also a specialization of GSIM: Information Set.

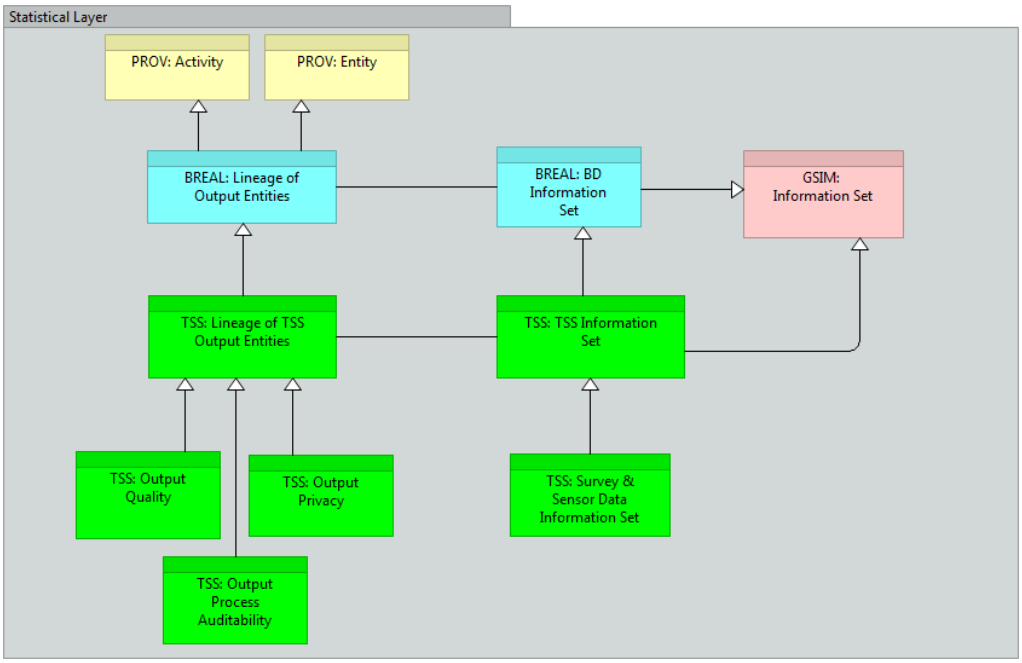

Figure 16: TSS$_u$ Statistical Layer

4 TSS$_u$ Process model

The subset of metadata that refers to Process area includes both the process specification and execution. For the process specification, at the moment, the focus is on the description of the throughput activities that transform raw sensor data in convergent data. This analysis is important to harmonize traditional survey steps and sensor data processing. The process execution allows to track data transformations and to complete the analysis of sensor data processing, an ontology for the workflow management will be examined.

4.1 Defining throughput activities

The throughput activities carried out to transform sensor data in statistical data correspond to the following BREAL business functions: Acquisition and Recording, Data Wrangling, Data Representation, Modeling and Interpretation. The following figure provides an overview of these activities that may differ in each survey, according to the type of data acquisition, the agreement with the data provider, the type of sensor data, in-app or in-house data processing. The main process steps that realize each business function are grouped in two subsets: Lower Throughput Activities and Upper Throughput Activities. The green colored objects correspond to the data objects associated to the process steps, thus connecting the business layer describing the process to the data layer describing data transformation. From the process perspective, Lower Throughput Activities refer to raw sensor data (TSS: Sensor Data Source), while the output of Upper Throughput Activities corresponds to the whole set of survey data (TSS: Survey & Sensor Data Information Set), collected combining different modes.

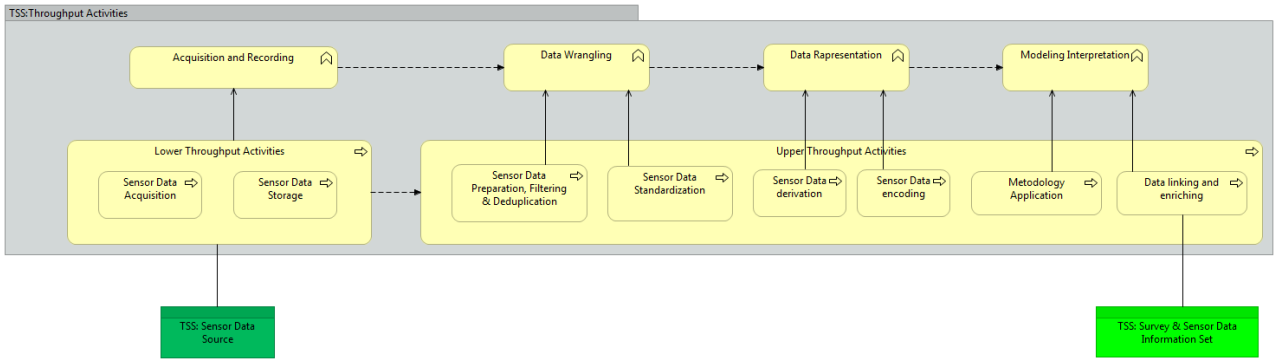

Figure 17: TSS$_u$ Throughput Activities

The following table summarizes the Business Functions, their process steps and the Data Layer involved during sensor data processing. In this context, metadata should help to document and monitor the main activities performed throughout the process. The analysis carried out in the following paragraphs, aims at defining the minimum set of metadata and dimensions to consider, in order to achieve an acceptable level of trust.

Business functions	Process Steps		Data Layer
Acquisition and Recording	Lower Throughput Activities	Data acquisition	Raw Data layer
		Data storage	
Data Wrangling	Upper Throughput activities	Sensor Data Preparation, Filtering & Deduplication	Convergence Data layer
		Sensor Data Standardization	
Data Representation		Sensor Data derivation	
		Sensor Data encoding	
Modeling and Interpretation		Methodology application	Statistical Data layer
		Data linking and enriching	

Table 14: Business Functions, process steps and the Data Layer involved in sensor data processing

4.2 Process model description

The proposed analysis fosters a process specification that allows to fulfill the following requirements:

- Provide an executable description of the process, in terms of process steps, data and metadata. A process description suitable for operational use enhances an active metadata management.
- Preserve the workflow integrity, to guarantee the process reproducibility and track process execution;
- Support the resilience to process, data, or environment changes.

In order to achieve the workflow preservation, the process model description has adopted the conceptual approach developed in the wf4ever Research Object Ontologies29. The basic idea of this group of four ontologies is to enrich the workflow description in natural language with information related to data, metadata and tools. This auxiliary information that allows to run and track the process, saving the intermediate results, is stored in special containers, called Research Objects. The concepts of these ontologies can be specialized to the TSS context, to identify and describe the main elements that belong to the subsets connected to the Process model. The Research Object approach is based on an extendable model for describing data aggregations, enriched with semantic annotations and supporting metadata that can be published and exchanged as a single artifact. The published artifact includes the workflow specification and the related data and metadata required to preserve and reproduce the process described.

In brief, the four ontologies and their main features are:

- **ro**: The Research Object Ontology provides a domain agnostic framework, to describe aggregated resources and their annotations,
- **wfdesc**: The Workflow Description Ontology allows to describe the Workflow specifications listed in a Workflow Research Object.
- **wfprov**: The Workflow Provenance Ontology aims at describing the provenance resulting from the Workflow execution.
- **roevo**: The Research Object Evolution Ontology is useful to describe the evolution of Workflow Research Objects and to track and describe the changes of a Workflow according to different levels of granularity.

Although these ontologies address specific issues of the scientific domain, this approach could be applied in the statistical context for describing the process steps and tracking process execution.

The following analysis skips the ro ontology, due to the use of GSIM framework to describe data and metadata and other information objects in the statistical domain.

The wfdesc ontology30 provides a structure for describing the workflow specifications stored in a Workflow Research Object. The main terms used to describe a workflow are the following:

- **wfdesc:Workflow** that allows to represent workflows, and is a subclass of the prov:Plan;
- **wfdesc:Process** that represents a step in a workflow;
- **wfdesc:DataLink**, used to specify data dependencies between different processes. A data link specifies the connection between the output of a given process and the input of another process.

The wfprov ontology allows to link the workflow descriptions to the provenance tracking of a workflow execution. The figure above describes wfdesc classes and their properties.

[29] Belhajjame K., Zhao J., Garijo D., Gamble M., Hettne K., Palma R., Mina E., Corcho O., Gómez-Pérez José M., Bechhofer S., Klyne G., Goble C.: Using a suite of ontologies for preserving workflow-centric research objects. Journal of Web Semantics (May 2015), available from: https://www.sciencedirect.com/science/article/pii/S1570826815000049

[30] The main characteristics of the ontology have resulted from the analysis of core and common concepts used in the following data driven workflow systems: Taverna, Wings and Galaxy. The ontology documentation is available from: https://wf4ever.github.io/ro/2016-01-28/#wfdesc

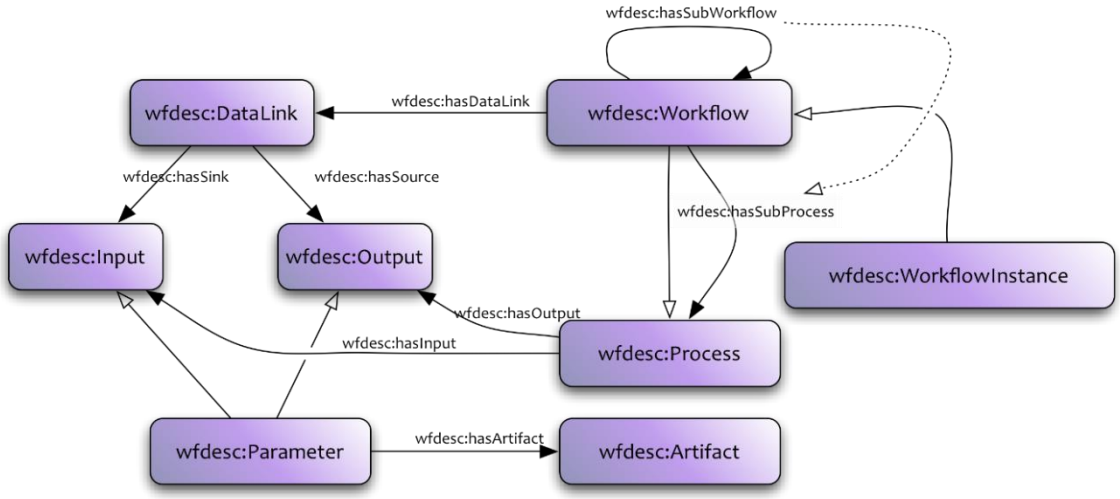

Figure 18: Workflow descriptions using the wfdesc ontology[31]

A Workflow (wfdesc:Workflow) is a graph in which the nodes are wfdesc:Process instances and the edges (wfdesc:DataLink instances) correspond to data dependencies between the main processes (wfdesc:Process). A wfdesc:Workflow includes several wfdesc:Process instances, grouped by the wfdesc:hasSubProcess property. The workflow and each process instance have input wfdesc:hasInput, output wfdesc:hasOutput and some parameters wfdesc:Parameter. An artifact (wfdesc:Artifact) can be associated with a wfdesc:Parameter, using wfdesc:hasArtifact. A process can have 0 or several wfdesc:Parameter instances, connected using wfdesc:hasInput and wfdesc:hasOutput, according to the kind of parameters required and returned by the process. Wfdesc:Workflow has wfdesc:hasDataLink with several wfdesc:DataLink instances, representing the connection between parameters. More precisely, wfdesc:DataLink instances specify data/parameters/settings represented in the wfprov:WorkflowRun.

The following table reports the classes of wfdesc ontology whose concepts can be mapped to the Metadata repository subsets.

wfdesc Ontology		**Metadata Repository Subset**	**Mapping with GSIM**
Classes	**Description**		
wfdesc:Artifact	Information about a class of artifacts (e.g. datatype of a dataset, structure of a document)	The related Subset depends on the specialization of the artifact	
wfdesc:DataLink	Data dependencies between wfdesc:Process descriptions	Process specification	
wfdesc:Input	Input parameter to a wfdesc:Process (e.g. function parameter, command line argument, files read, parameter set by a user)		
wfdesc:Output	Output parameter from a wfdesc:Process (e.g. functional return values, stdout/stdin, files written, or results shown in a user interface)		

[31]Source: https://wf4ever.github.io/ro/2016-01-28/#wfdesc

wfdesc:Parameter	Parameter of a wfdesc:Process (e.g. wfdesc:Input, wfdesc:Output, wfdesc:Configuration)		
wfdesc:Process	Class of actions to execute		
wfdesc:WorkflowInstance	Specialisation of a wfdesc:Workflow defining data/parameters/settings included in the wfprov:WorkflowRun		

Table 15: Wfdesc classes mapped to the Metadata repository subsets

4.3 Modeling process execution

The wfprov ontology describes the provenance resulting from workflows execution. The description of workflows and the traces of their execution allow to track data transformation throughout a process, linking input data sources, intermediate output and final results. Such information is essential for process auditability, assessing efficiency and effectiveness of final results in terms of activities, resources and behaviors. Wfprov ontology is also aligned with the prov-o ontology, more precisely: wfprov:ProcessRun representing a process performance is a subclass of prov:Activity, while wfprov:Artifact representing an artifact used or generated by a specific process run and it is a subclass of prov:Entity.

The figure below (Figure 19) shows wfprov classes and their properties: wfprov:WorkflowRun describes the running of a wfdesc:WorkflowInstance and groups the wfprov:ProcessRuns included in the execution with the relationship wfprov:wasPartOfWorkflowRun. Each wfprov:ProcessRun is a step of the workflow execution and may have dependencies to several wfprov:Artifacts, which are represented by the relationship wfprov:usedInput. Each result produced by one of the wfprov:ProcessRuns is connected by the relation wfprov:wasOutputFrom. Different workflow engines may execute several wfprov:ProcessRuns, thus the wfprov:wasEnactedBy relationship specifies the engine performing each step. In the end, each wfprov:Artifact, wfprov:ProcessRun and wfprov:WorkflowRun can be linked to their correspondant wfdesc description, provided by the following relationships: wfprov:describedByParameter, wfprov:describedByProcess and wfprov:describedByWorkflow.

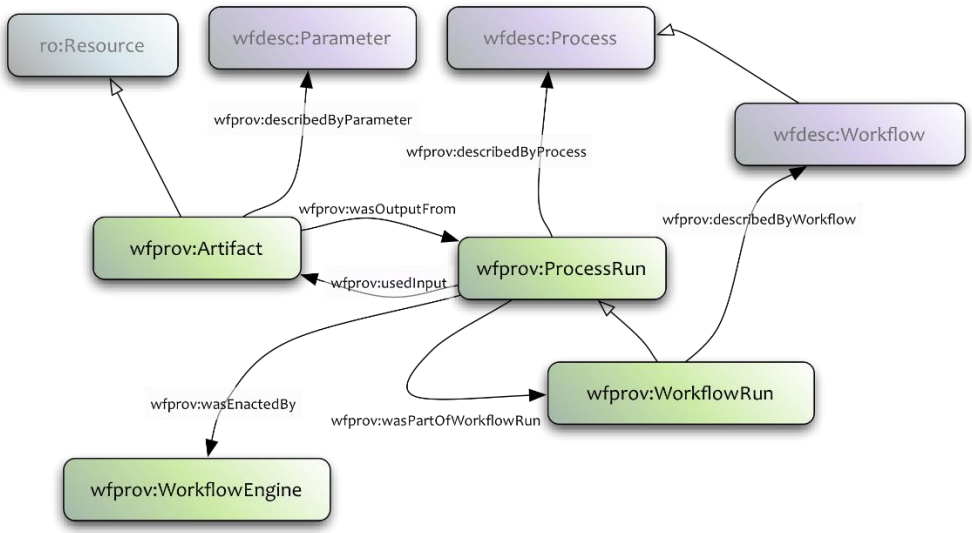

Figure 19: Wfprov ontology classes[32]

The following table reports the description of wfprov classes and the correspondence with Metadata Repository subsets.

wfprov Ontology		Metadata Repository Subset	Mapping with GSIM
Classes	**Description**		
wfprov:Artifact	Data value or item which wfprov:wasOutputFrom of a wfprov:ProcessRun or used as input (wfprov:usedInput) during process execution	The related Subset depends on the specialization of the artifact	
wfprov:ProcessRun	Particular execution of a wfdesc:Process description (wfprov:describedByProcess), having wfprov:usedInput, some wfprov:Artifact instances and producing new artifacts (wfprov:wasOutputFrom)	Traceability	
wfprov:WorkflowEngine	foaf:Agent responsible for executing a workflow definition (described in a wfdesc:Workflow). The result of workflow execution gives rise to a wfprov:WorkflowRun		
wfprov:WorkflowRun	Specialization class of wfprov:ProcessRun, executed by a wfprov:WorkflowEngine, following a workflow definition. A process may include several subprocesses (wfprov:wasPartOfWorkflowRun) according to wfdesc:Process descriptions		

Table 3: Wfprov classes mapped to the Metadata repository subsets

The next figure provides an overview of the relationship between wfdesc and wfprov ontologies. It also highlights the connections with the prov-o ontology.

[32] Source: https://wf4ever.github.io/ro/2016-01-28/#wfdesc

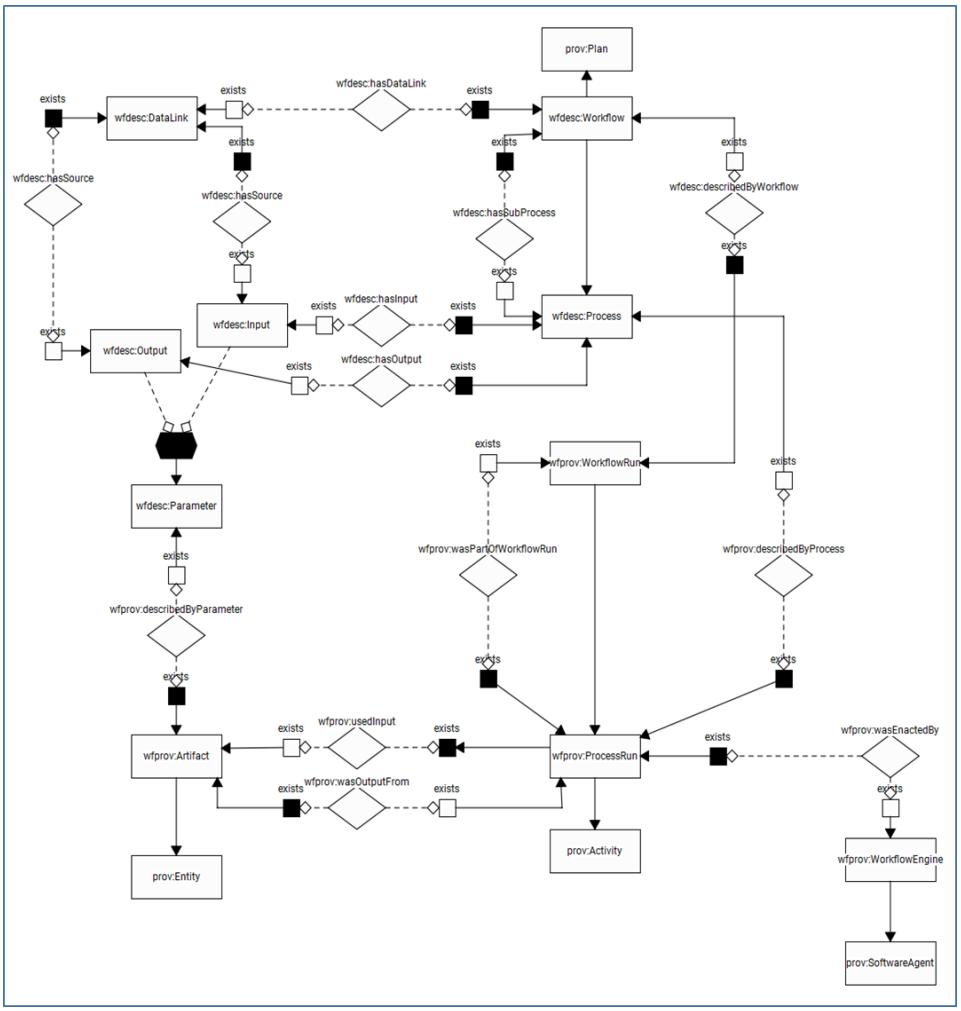

Figure 8: Relationship between wfdesc, wfprov and prov-o ontologies

5 TSS_u Trust model

The following paragraphs focus on some of the main aspects of Trust, corresponding to specific subsets of Metadata Repository, namely: Quality, Privacy and Traceability. The combination of these dimensions with Provenance and Traceability correspond to process auditability that provides an assessment of the main statistical steps and their results.

5.1 Modelling quality metadata

The objective of the proposed model for quality metadata is to provide synthetic indicators to assess the reliability of the results achieved in the main process steps. According to the Big Data Quality Framework (BDQF)33, developed from existing statistical data quality frameworks, the main phases of the statistical process to consider for quality assessment are:

- Input, that includes all the activities related to data collection stage, regardless of the type of data source;
- Throughput, that refers to data transformation performed during process and analyze stages;
- Output, corresponding to data evaluation and dissemination stage.

Regarding the quality dimensions, BDQF has adopted the hierarchical structure composed of three hyperdimensions, developed by Statistics Netherlands[34] and integrated into the administrative data framework released by the Statistical Network (SN). The three hyperdimensions are:

- Source, that refers to general aspects, such as the type of data, the characteristics of data provider, and the source governance and delivery. The Source hyperdimension includes five quality dimensions: Supplier, Relevance, Privacy and security, Delivery, and Procedures. These dimensions are assessed mainly by qualitative methods, resulting in scores and rankings.
- Metadata, that relates to the source meta information and particularly to clarity of the definitions and completeness of the source metadata. The Metadata hyperdimension contains four dimensions: Clarity, Comparability, Unique keys, and Data treatment performed by the data provider. All the dimensions of this subset are measured by qualitative methods.
- Data, that examines the quality aspects of the data, acquired or processed.

As Source and Metadata hyperdimensions relate to quality aspects of a data source provided by data owner, they can be used to assess the suitability of a source for specific purposes before its acquisition, during the so called “discovery phase”. The quality assessment provided by the Data hyperdimension, can be achieved only after the data is actually acquired.

Considering TSSu, the first two stages of the statistical process may combine traditional and smart data, methods and tools. Since the present analysis addresses mainly sensor data directly acquired by NSIs, the quality issues related to traditional data, as well as sensor data provided by third-parties are out of scope for the moment. In order to detect and overcome relevant issues arising during data acquisition and processing, in this context, the main quality dimensions to focus on are related to the Data hyperdimension and more precisely to:

[33] M. Signore, *et al.* “A Suggested Framework for the Quality of Big Data” (2014)

[34] Daas, P., Ossen, S., Vis-Visschers, R., Arends-Toth, J. (2009), Checklist for the Quality evaluation of Administrative Data Sources. Statistics Netherlands, The Hague/Heerlen

- Accuracy (and its sub-dimension Selectivity) allow to assess the target population coverage, or detect selectivity effects, occurring when specific sub-populations are under/overrepresented or totally excluded.
- Coherence, expressed in terms of consistency and linkability. As a whole, this dimension measure the compliance of a data source, or a dataset with internal standard definitions and its consistency over time or with other data sources (Consistency). Further, it aims at measuring the quality of linking variables (Linkability).
- Validity that provides a measure of the coherence between processes, methods and collected data values.

The following table reports an initial set of quality indicators for each process step related to data acquisition and throughput activities of sensor data. Another dimension added to the quality dimensions specified above is Accessibility, as suggested in the Quality check list for the acquisition phase – Data, one of the deliverables of the MIAD35 project by the Statistical Network. This list refers to acquired administrative sources and contains some key concepts that can be adjusted for measuring the quality of sensor data acquisition.

Business functions	**Process Steps**		**Quality dimension**	**Indicator**
Acquisition and Recording	Lower Throughput Activities	Data acquisition	Accessibility	Number of acquired observations
		Data storage		Number of stored observations
Data Wrangling	Upper Throughput activities	Sensor Data Preparation, Filtering & Deduplication	Accuracy	Brief description of the whole data preparation task
				Filtering rules and Percentage of filtered observations
				Percentage of duplicated observations
		Sensor Data Standardization		Brief description of the data standardization task
Data Representation		Sensor Data derivation		Mapping between sensor and statistical units
				Mapping between sensor data and statistical variables
		Sensor Data encoding		Percentage of items derived from sensor data successfully coded
Modeling and Interpretation		Methodology application		Brief description of the methods applied to model sensor data
				Percentage of missing items derived from sensor data
		Data linking and enriching	Linkability	Percentage of units successfully linked with reference datasets
				Percentage of units erroneously linked with reference datasets

Figure 20: Measures of sensor data quality during Throughput activities

[35] For an overview of the activities and the related deliverables see: https://ec.europa.eu/eurostat/cros/content/miad-methodologies-integrated-use-administrative-data-statistical-process_en

The proposed quality indicators provide an initial assessment of the reliability of acquired sensor data. Further analysis is needed to validate the completeness of the above quality measures[36]. A broader quality assessment will result from the integration of the results provided by the methodological task. Following an evidence based approach, a TSS_u use case based on WP2 feedback and the planned Proof of Concepts (POC_s) will be essential to test and improve the suggested quality metadata model.

5.2 Privacy preservation throughout the process

The privacy issue analyzed from the metadata perspective is tightly connected to process description and execution. The basic idea of this approach is to identify the process steps that require the execution of one or more tasks related to privacy preservation. On a higher level, these tasks can be grouped in the following subsets:

- **Privacy assessment**, related to all the activities carried out for assessing potential privacy risks. In most cases, this assessment is particularly relevant during the design phase of a process, when the principles of privacy by design and by default are applied to define an overall strategy for privacy preserving. Data protection impact assessments (DPIAs) are part of this subset. Concerning these activities, the role of metadata could be the documentation of the assessment and the results achieved by DPIA. Due to the iterative implementation of the platform, it would be useful to specify also the conditions that trigger an analysis to prevent and avoid any privacy violation. These specifications would assist in monitoring the impact of developing activities on privacy preservation.
- **Privacy management**: this subset is mainly related to data consent management and third-party data protection. A respondent may revoke the initial consent provided during data collection, or special privacy protection is for particular variables (e.g., personal data, medical data) or vulnerable individuals. In this case, the role of metadata is to assist in an effective management of data consent, tracking the initial consent and any following modification for each respondent. In addition, metadata should highlight the subset of data having a special protection in the privacy Regulation. Referring to third-party data protection, metadata could allow to document and monitor the agreements with third-parties for data processing. In addition, metadata should document and report the results of the controls planned to monitor and assess the activity performed by third-parties. This approach is consistent with one of the design principles of TSS_U: Pushing computation out instead of pulling data in.
- **Privacy preserving techniques** includes all the tasks of secure private computing, performed by the application of a privacy preserving technique, to be added in the subset of the metadata repository that lists the methods implemented in the platform components.
  At first glance, the application of these techniques during data acquisition and dissemination is a reasonable assumption, to be confirmed by real use cases.

In order to improve this first analysis to monitor the privacy requirements throughout the statistical process, further investigation is needed37. The results of task 3.1.4 “Preservation of privacy and transparency”, as well as the feedback of POCs will be very useful to model metadata taking into account the different privacy issues.

[36] Future enhancements of this work will take into account the results of the UNECE – HLG-MOS Machine Learning Project ‘WP2 - Quality Framework for Statistical Algorithm’

[37] Future enhancements of this work will take into account the ouputs of the UNECE- HLG-MOS project "Input Privacy-Preserving Techniques". Valuable feedback will also be provided by works conducted at the W3C community level, in particular the Data Privacy Vocabulary (https://dpvcg.github.io/dpv/)

5.3 Tracking process execution

The objective of roevo ontology is to track the evolution of Workflow Research Objects, considering different levels of granularity. More precisely, it provides general information about changes (creation and current status), as well as the type of changes made to an aggregation of individual resources (additions, modifications and removals). The roevo ontology is also an extension of the prov-o ontology, whose classes provide the main concepts for describing the evolution of Research Objects. Particularly, the following sub-classes of prov:Entity have been created to provide provenance and track the different states of a Research Object:

- roevo:LiveRO, that corresponds to a Workflow Research Object created and to be populated. As the content or state of their resources are not stable, Live ROs are unfinished objects and may change over time.
- roevo:ArchivedRO is a production Workflow Research Object to be preserved and archived. It represents the last stage of a RO, having reached a stable version to be released or, otherwise to be deprecated. As a consequence, they are immutable and no other change or version are accepted.
- roevo:SnapshotRO provides representation of a liveWorkflow Research Object at a given point in time. As they record the state of the Live RO in a specific moment, they are not susceptible to change.

In the following figure, roevo:VersionableResource represents the evolution of a resource, which can be a roevo:SnapshotRO, a roevo:ArchivedRO, a ro:Resource, or ro:AggregatedAnnotation. As it allows to track the provenance of a roevo:VersionableResource, this class can be considered a specialization of prov:Entity.

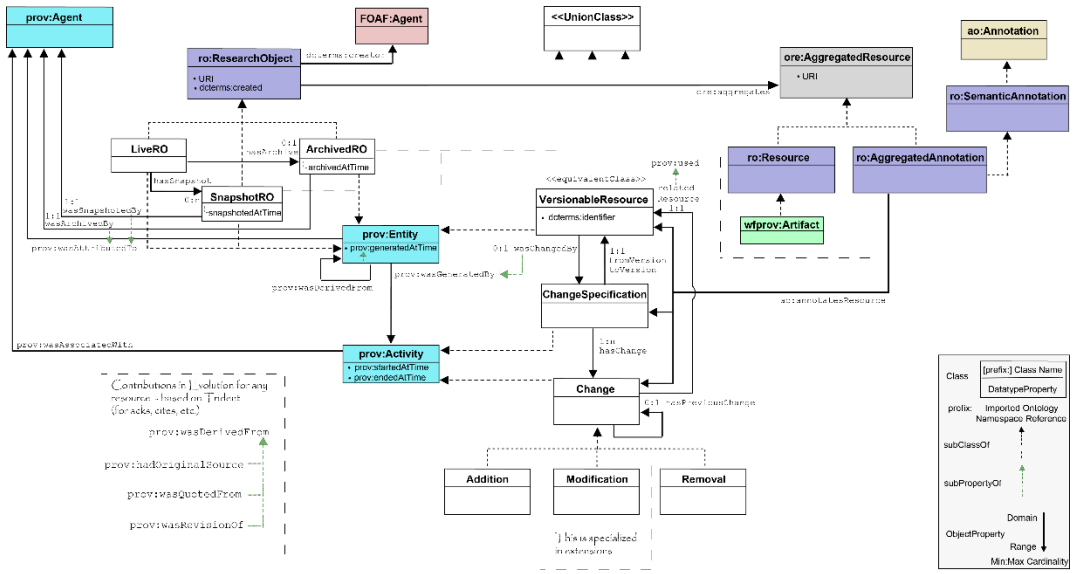

Figure 21: Classes of roevo ontology and relationship with prov-o ontology[38]

[38]Source: https://wf4ever.github.io/ro/2016-01-28/roevo/

wfprov Ontology		Metadata Repository
Classes	**Description**	**Subset**
roevo:Addition	Type of roevo:Change that changes a Research Object by adding to the aggregation some resources, annotations, content and/or structure	Traceability
roevo:ArchivedRO	Subclass of ro:ResearchObject specifying a type of Research Object in archival state	
roevo:Change	A prov:Activity that changes the Research Object aggregation, content, structure, its annotations or aggregated resources	
roevo:ChangeSpecification	Group of roevo:Changes activities corresponding to the changes between two different versions of a Research Object	
roevo:LiveRO	Subclass of ro:ResearchObject specifying a type of Research Object in live state	
roevo:Modification	Type of roevo:Change that represents an update of an aggregated resource content or an existing annotation, thus changing the aggregation, content and/or structure of a Research Object	
roevo:Removal	Type of roevo:Change that changes a Research Object by removing a resource or an annotation	
roevo:SnapshotRO	Subclass of ro:ResearchObject specifying a type of Research Object in snapshot state	
roevo:VersionableResource	A prov:Entity representing an artifact to be versioned and for which the track of different versions is required	

6 Next steps

In order to validate the metadata repository described in this document, a PoC will be implemented in the second phase of this ESSnet research project. The analysis will allow to model i) data collection process steps, ii) application components involved in data processing and metadata management. Therefore, the PoC will facilitate the specification of the platform requirements, providing an enhancement of the architectural framework.

More precisely the analysis of the process steps should investigate the integration between sensor and traditional data to highlight potential challenges during process execution due to the combination of different data sources.

The following roadmap describes the main PoC steps:

Business process modelling: describe each process step of a pilot survey in terms of input/output data, methods, and metadata, to test the consistency between the theoretical model and survey implementation.

Modelling of platform application components: detailed analysis of one or more software components implemented by WP2 for a pilot survey.

Metadata modelling:

- Input/output data structures
- Methods
- Sensor data transformations throughout the process steps
- Quality assessment

- Privacy issues.

IT platform requirements:

- Frontend: the platform should provide a data collection tool that allows to design questionnaires for mobile devices, web browsers, etc
- Backend: the backend should be designed and implemented according to modern cloud-native architectures. The infrastructure hosting the backend could be on-premises, in the cloud (Amazon, Google, Azure) or both
- Data storage: data should be stored in Relational databases (MySQL, PostgreSQL, etc) in NoSQL databases (MongoDB, Redis, Couch DB, etc) or both? Where should data be stored (on-premises, at national level, in the cloud)?
- Security issues: privacy preserving techniques to be implemented in ad-hoc components? Privacy by design should be a guiding principle in the design and implementation of the platform.

Annex 1: Semantic Sensor Network Ontology

The Semantic Sensor Network Ontology39 provides an overview of the key concepts that allow to describe, model and ingest sensor data. This ontology has different conceptual modules and includes a self-contained core ontology called SOSA (Sensor, Observation, Sample, and Actuator). The following figure provides the different conceptual modules of SOSA.

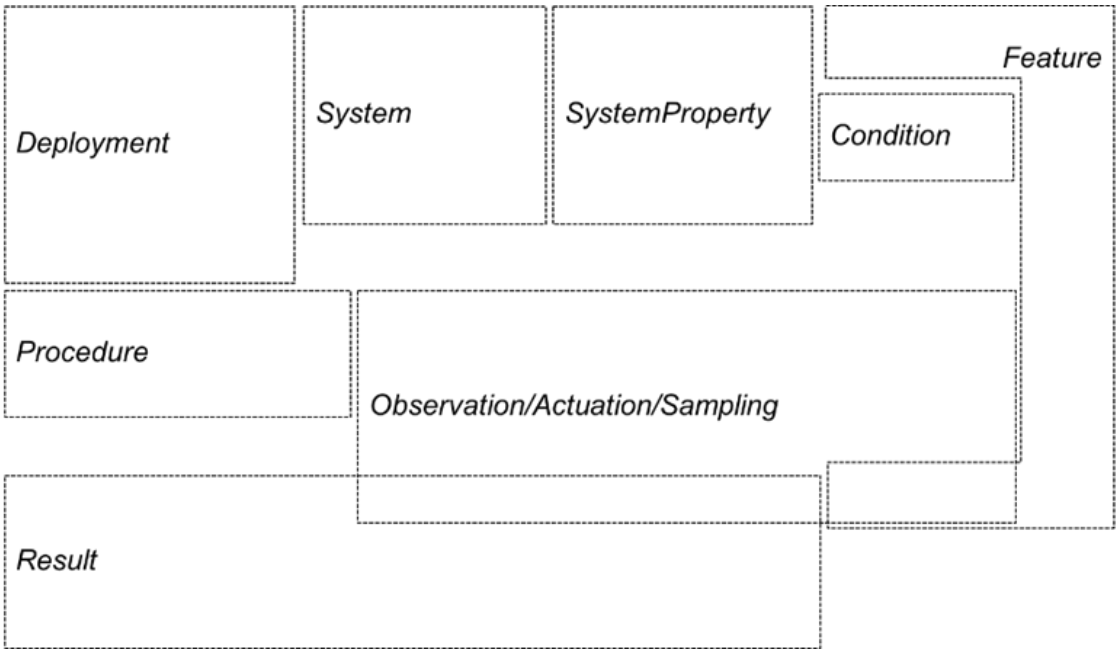

Figure 22: Overview of the SOSA/SSN ontology modules

The following figures provide an overview of the main classes and properties belonging to different modules, from the perspectives of Observation, Actuation and Sampling.

[39] The ontology is available from: https://www.w3.org/TR/2017/REC-vocab-ssn-20171019/

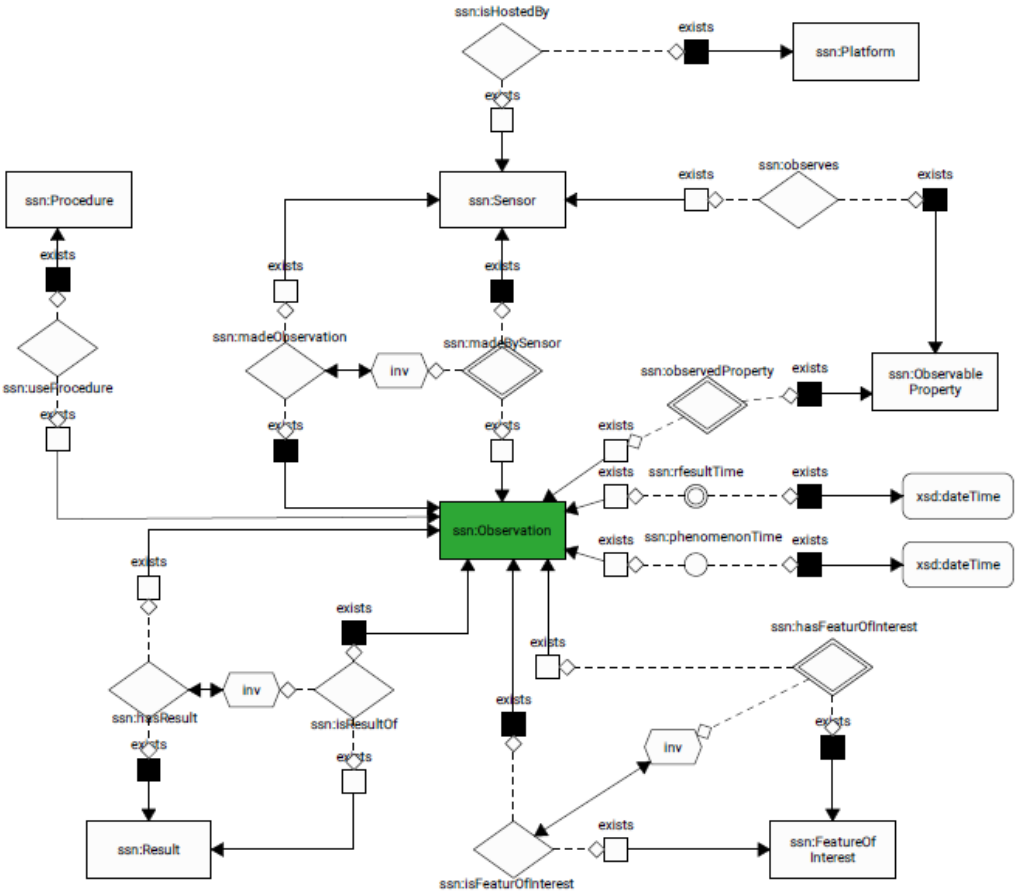

Figure 23: SOSA Observation perspective

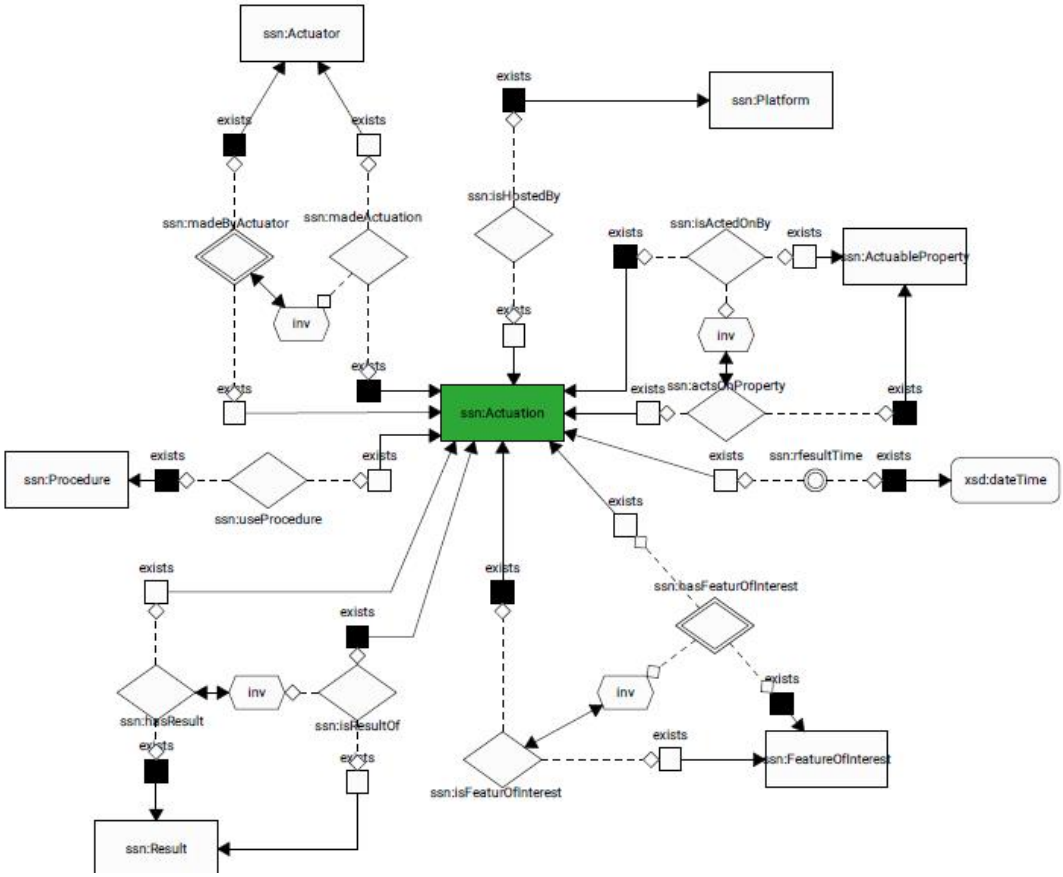

Figure 24: SOSA Actuation perspective

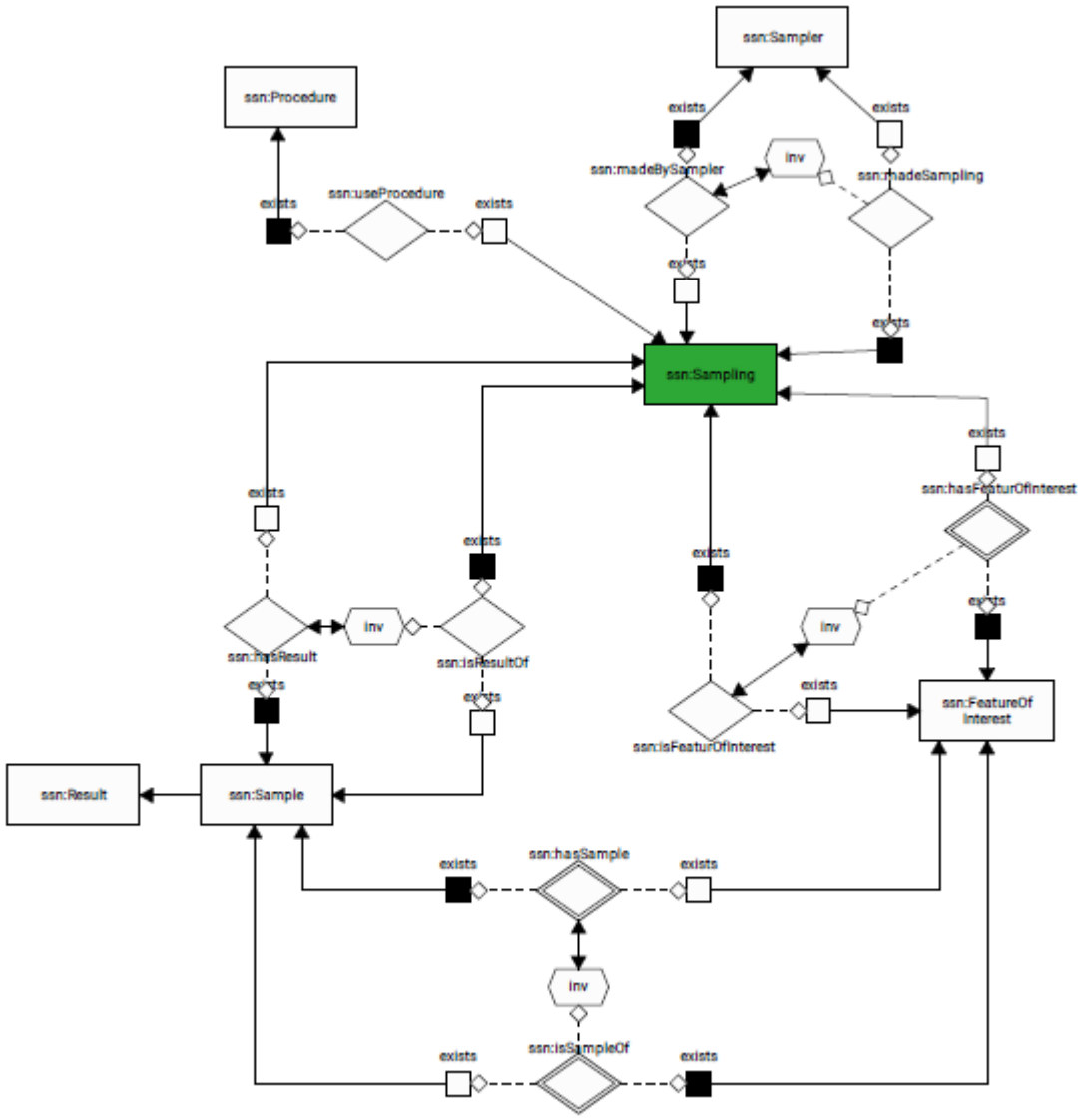

Figure 25: SOSA Sampling perspective

Alignment between SOSA and PROV

The primary classes from SOSA are aligned with the PROV classes[40]. PROV has three main classes:

- Entity, the class that includes physical, digital, conceptual, or different types of things with predetermined features;
- Activity is everything that occurs over a period of time, acting upon or with entities. It may refer to consuming, processing, transforming, modifying, relocating, using, or generating entities;
- Agent, the class of things responsible for an activity taking place, for the existence of an entity, or for another agent's activity.

SOSA	Relationship	PROV
sosa:Observation	subclass of	prov:Activity
sosa:Actuation		
sosa:Sampling		
sosa:Sensor	subclass of	prov:Agent
sosa:Actuator		
sosa:Sampler		
sosa:FeatureOfInterest	subclass of	prov:Entity
sosa:ObservableProperty		
sosa:Platform		
sosa:Result		
sosa:Sample		

SOSA main Classes and Provenance metadata

In order to track sensor data provenance, only the SOSA Classes aligned with the PROV ontology have been analyzed in detail. The following table reports the description of the Classes corresponding to the elements grouped in the Provenance subset, within the Metadata repository. The analysis of these classes is helpful to define the list of metadata that relate to the sensor data observations.

sosaOntology		**Metadata Repository**
Classes	**Description**	**Subset**
sosa:Observation	Act of carrying out an (Observation) Procedure to estimate or calculate a value of a property of a FeatureOfInterest. Links to a Sensor to describe what made the Observation and how; links to an ObservableProperty to describe what the result is an estimate of, and to a FeatureOfInterest to detail what that property was associated with.	Provenance - Activity
sosa:Actuation	An Actuation carries out an (Actuation) Procedure to change the state of the world using an Actuator.	
sosa:Sampling	An act of Sampling carries out a (Sampling) Procedure to create or transform one or more Samples..	
sosa:Sensor	Device, agent (including humans), or software (simulation) involved in, or implementing, a Procedure. Sensors respond to a Stimulus, e.g., a	Provenance – Agent

[40] Prov ontology is available from: https://www.w3.org/TR/prov-o/

	change in the environment, or Input data composed from the Results of prior Observations, and generate a Result. Sensors can be hosted by Platforms.	
sosa:Actuator	A device that is used by, or implements, an (Actuation) Procedure that changes the state of the world.	
sosa:Sampler	A device that is used by, or implements, a (Sampling) Procedure to create or transform one or more samples.	
sosa:FeatureOfInterest	The thing whose property is being estimated or calculated in the course of an Observation to arrive at a Result, or whose property is being manipulated by an Actuator, or which is being sampled or transformed in an act of Sampling.	Provenance - Entity
sosa:ObservableProperty	An observable quality (property, characteristic) of a FeatureOfInterest.	
sosa:Platform	A Platform is an entity that hosts other entities, particularly Sensors, Actuators, Samplers, and other Platforms.	
sosa:Result	The Result of an Observation, Actuation, or act of Sampling. To store an observation's simple result value one can use the hasSimpleResult property.	
sosa:Sample	Feature which is intended to be representative of a FeatureOfInterest on which Observations may be made.	

ANNEX: Planning of Proof-of-Concept

A parallel activity carried out in 2020 was the definition of the planning for the Proof-of-Concept, to be conducted in the second part of the ESSNet, in 2021.

The objective is the development of proofs-of-concept in the form of modular prototype elements for essential aspects of the architecture such as:

- active and passive data collection
- use of machine learning for the identification of activities, identification of parallel activities, missing data, etc.
- privacy-preserving computation solutions
- full transparency and auditability of processing algorithms
- integrating of incentive schemes into the platform
- front-ends for configuration (allowing survey managers to instantiate and run new surveys with full support for multilingual needs)

The activities concerning the Proof-of-Concept are organized in Task 3.2, according to the following list of sub-tasks:

3.2	Development of Proof-of-Concept	**Istat**
3.2.1	Data collection and survey methodology	**Istat** Destatis Cbs
3.2.2	Use of machine learning for evaluating collected data	**Destatis** Cbs Insee Istat
3.2.3	Privacy-preserving computation solutions	**Cbs** Istat Insee
3.2.4	Process auditability solutions	**Insee** Cbs
3.2.5	Integrating of incentive schemes into the platform	**Destatis**

The proposed PoCs have been grouped into two sets, according to the perspective of the issues addressed: one group has a more methodological point of view, while the second concerns more technical and architectural aspects. Furthermore, this distinction follows the structure of the two working groups set up within ESSNet with the dual purpose of linking the activity of Work Package 2 and Work Package 3 and of linking the activities of the sub-tasks within WP3.

According to these two general areas, the Proof-of-Concept is described in the following paragraphs.

Methodological Proof of Concepts – Clustered Task 3.2.1, 3.2.2, 3.2.5

We do take an integrative approach for our Proofs of Concepts (PoCs) of sub-task 3.2.1, 3.2.2 and 3.2.5. All three PoCs aim at different aspects of a broader smart survey methodology, which makes them distinct to each other but also allows them to contribute to each other. A high-level goal of this integration is to close the gap between the high-level domain agnostic description of the European Platform for Trusted Smart Surveys and the actual components such a platform could provide to help implementing smart surveys. In preparation of these PoCs several key features of a smart survey were addressed from a methodological perspective: Active and passive (sensor) data collection, machine learning to assist in analysis and reducing the burden for respondents with smart assistance, and incentives to further reduce the burden for

respondents or to even make a survey a fun and engaging experience. The identified key features, the overreaching methodological arch and the planning of a platform resulted in an idea to produce a small-scale example of a domain agnostic pipeline for smart data and smart data processing. It was clear immediately that all aspect should be covered in a smart way, which meant not only data but also the respondents' ease and incentives should be handled smart.

The combined PoCs of sub-task 3.2.1 and 3.2.2 will pick up the pipeline at the moment sensor data is available in a smart survey setting and develop a generalized machine learning process to evaluate the quality of collected data. The process will be separated in different software modules that will allow it to be applied to different contexts and survey needs. A general assumption is that sensor data shares many common features, because it is basically signal data and the main problem when dealing with signal data is anomaly detection. In addition, sensors do not often measure directly variables of interest and the PoC will explore ways to assist and in best cases automatically find hidden insights.

Insights from data and generally information flowing back to a respondent are of interest in the creation of incentives. In particular incentives that use a gamified approach and apply additional meaning to a report. The PoC of sub-task 3.2.5 compliments the work done in the other methodological PoCs by re-using meaningful data for gamified incentives in forms of reports. The goal of this activity is to identify how the smart survey itself can help to create its incentives. The outcome will consist of software modules that can integrated in various app-based smart surveys.

All PoCs will use the iLog data set provided by the University of Trento, which is similar to time use survey data and consists of diary and sensor data. This forms a baseline for each developed module of the PoCs in terms of modularity of the components. For further testing of (raw) sensor data sub-tasks 3.2.1 and 3.2.2 will also use health diary and accelerometer sensor data collected in Work Package 2.3.

In a later phase, more data coming from Work Package 2 will be included in all PoCs as additional data sources. In particular, data from tasks that deal with HBS and TUS. Although, at this moment it is not absolutely clear what these data packages will contain and how we will deal with them, we want to include them to further connect the work packages and bridge the gap between the hands-on tests of Work Package 2 and high-level concepts of Work Package 3. Creating these links will add additional insight on how to design the European Platform for Trusted Smart Surveys.

Task 3.2 – Sub-task 3.2.1 & 3.2.2 "Treatment of sensor data for Smart Surveys through Machine Learning Generalized Module."

<table>
<tr><td>Driver (Rationale behind the PoC, High level Goal)</td><td colspan="2">Trust Smart Surveys Platform.
Generalizing a machine learning module capable of operating on different surveys.
Proving effectiveness and feasibility of using ML algorithms on sensor data for Smart Statistics and implementing procedures to evaluate the quality of collected data.</td></tr>
<tr><td>Expected outcomes and impact</td><td colspan="2">To produce a simple example of a domain agnostic pipeline that processes sensor data and infers variables of interest.
Minimization of the statistical burden for users through smart technologies.
Data enrichment.
Evaluation of sensor data quality.</td></tr>
<tr><td rowspan="6">PoC description</td><td>Overview</td><td>In a Smart Survey platform, it is fundamental to use software modules that can be applied to different contexts and surveys.
Machine learning is an essential phase of sensor data processing and the fact that its implementation is domain agnostic draws great interest.
Our goal is designing and developing a generalized machine learning module apt to being used on different surveys.
Moreover, procedures and methods for data quality evaluation will be tested.</td></tr>
<tr><td>Scope</td><td>Trust Smart Statistics & Machine Learning.</td></tr>
<tr><td>Assumptions</td><td>A machine learning module can be generalizable to different surveys if it deals with data provided by the same type of sensor (accelerometer).</td></tr>
<tr><td>Methods</td><td>Signal analysis.
Machine Learning Models (Supervised Classifier).
Data anomaly detection.</td></tr>
<tr><td>Data sources</td><td>- WP2.3 Health diary and accelerometer sensor data.
- iLog data (provided by University of Trento, available 02/2021); similar to time use survey with questionnaire, diary and sensor data.</td></tr>
<tr><td>Success criteria</td><td>Designing and developing a generalized machine learning component with satisfying accuracy on all surveys taken into account.</td></tr>
<tr><td>Risks</td><td>The resulting accuracy can be affected by the generalization requirement.</td></tr>
<tr><td>Roadmap</td><td colspan="2">1) Identification of best practices for the processing of accelerometer data
2) Data acquisition from iLog (University of Trento) e WP3.2
3) Evaluation of the quality of acquired data and error detection
4) Designing the Generalized ML module
5) Data preparation
6) Dimensionality reduction of sensor data
7) Training of supervised classifier model for both surveys
8) Model performance evaluation
9) Designing the possible integration of the trained ML model in all TSS process (e.g. use of the algorithm to give feedbacks and reports to the user.)</td></tr>
</table>

Synergies and Interdependencies between tasks/WP	Incentive scheme to provide users with a report. Possible Privacy WP task for Privacy Preserving Input ML module.
Deliverables	Report on PoC - Planning - Results

Task 3.2 – Sub-task 3.2.5 "PoC on Integration of Incentive Schemes into the Platform"

Driver *(Rationale behind the PoC, High level Goal)*	- Incentives made out of survey data - Using gamification as a design principle to create a strong incentive - Explore generic gamification based on active and passive survey data - Explore how gamification can be implemented with little change to an app	
Expected outcomes and impact	- automatic reports from sensor data - easy to create and to implement incentive for quantitative survey data	
PoC description	**Overview**	
	Scope	
	Assumptions	
	Methods	- UI and (minimal) back-end programming
	Data sources	- iLog data (provided by University of Trento, available 02/2021); similar to time use survey with questionnaire, diary and sensor data.
	Success criteria	- Mockup and at least three reporting screens that use a gamified theme
	Risks	- lack of ressources: only one participant involved - lack of infrastructure
Roadmap	- EDA to define categories for automated reporting - Automatic reports from survey data - Automatic reports from sensor data - Mockup - Tailor reports for a gamified design based on user types explored in task 3.1.5	
Synergies and Interdependencies between tasks/WP	- Merges into 3.2.1 and 3.2.2 in regard to methodological perspective.	
Deliverables	Report on PoC - Planning - Results	

Technical Architectural Proof of Concepts – Clustered Task 3.2.3, 3.2.4

The main goal of the PoCs is to close the gap between the high-level description of the TSS platform components (WP3) and the hands-on technical implementation of the pilots (WP2). More precisely, for each dimension of analysis (architecture, technology, and metadata), the PoCs will ensure the alignment between WP2 pilots and WP3 models. In Trusted Smart Surveys, security is a key issue that should be included 'by design'. Implementation of Privacy Enhancing Technologies (PET) in the context of Smart Surveys could enable collection of new sources of data.

We planned three PoCs with the following goals:

- **Metadata and Architecture:** the metadata task has proposed a metadata Repository composed of several subsets and has investigated metadata related to sensor data. This PoC, starting from the pilot experience, should help identifying the metadata related to the sensor data, track data

transformation throughout the executed steps, and classify the identified metadata according to the subsets of the proposed metadata Repository. Further the PoC should allow to test the modelling of TSS business layer performed in Task 3.1.3.

- **Technical Infrastructure:** the main goal of this PoC is to verify the scalability, maintainability and security of the technical infrastructure, based on the conditions written in Task 3.1.2. Further it should allow to detail the list of technical requirements provided in Task 3.1.2.
- **Privacy enhancing technologies** are an essential part of the trust aspect of TSS. Currently, survey participants are asked to trust NSO's with their data. However, given the wealth of information present in smart device data, risks involved in sharing these data is high. Blindly trusting NSO's with these data might be too much to ask of survey participants. Privacy enhancing technologies could bridge this gap when trust in these technologies is established. The main goal is of the PoC is to investigate the possibility of (1) local pre-processing on the participants device to minimize data and prepare for privacy enhanced computation and (2) subsequent private computation of simple statistical output(e.g. average, total, etc.) using pre-processed data, without individual records being exposed.

Task 3.2 – Sub-task 3.2.4 – Full transparency and auditability of processing algorithm "POC on architecture and metadata"

Driver (Rationale behind the PoC, High level Goal)	The main goal of the PoC is to test the assumptions and the results of the "metadata framework" (T3.1.6) task and to provide a feedback on the business layer modelled by the architectural task. In addition, the PoC will provide a set of technical requirements for the TSS platform, based on WP2 field experience and on Task 3.1.2 outcome	
Expected outcomes and impact	Benchmark of metadata repository subsets modelled according to ontology concepts. Model the data collection steps in detail, particularly ML steps. Identify the metadata related to the sensor data collection and processing. Document data structures and methods applied in each step. Track data transformation throughout the executed steps. Classify the identified metadata according to the subsets of the proposed metadata Repository. Provide an initial description of metadata component requirements.	
PoC description	**Overview**	The PoC is intended to validate the metadata repository, designed for process tracking and auditability. The analysis will allow to model i) data collection process steps, ii) application components involved in data processing and metadata management. Therefore, the PoC will facilitate the specification of the platform requirements, providing an enhancement of the architectural framework.
	Scope	Metadata, Process design, IT platform requirements
	Assumptions	The metadata task has proposed a metadata Repository composed of several subsets and has investigated metadata related to sensor data. This PoC, starting from the pilot experience, should provide evidence that the results achieved are consistent with the implemented solutions.
	Methods	Process steps description, analysis of the IT solutions implemented in the analysed pilot surveys
	Success criteria	▪ First version of TSS platform metadata repository ▪ Overview of TSS platform business layer ▪ Overview of TSS platform application components ▪ Specification of a set of technical requirements for the TSS platform

	Risks	The analysis of the process steps should investigate the integration between sensor and traditional data to highlight potential challenges during process execution due to the combination of different data sources.
Roadmap	Business process modelling: describe each process step of a pilot survey in terms of input/output data, methods, and metadata, to test the consistency between the theoretical model and survey implementation. Modelling of platform application components: detailed analysis of one or more software components implemented by WP2 for a pilot survey. Metadata modelling: ▪ Input/output data structures ▪ Methods ▪ Sensor data transformations throughout the process steps ▪ Quality assessment ▪ Privacy issues IT platform requirements: ▪ Frontend: the platform should provide a data collection tool that allows to design questionnaires for mobile devices, web browsers, etc ▪ Backend: the backend should be designed and implemented according to modern cloud-native architectures. The infrastructure hosting the backend could be on-premises, in the cloud (Amazon, Google, Azure) or both ▪ Data storage: data should be stored in Relational databases (MySQL, PostgreSQL, etc) in NoSQL databases (MongoDB, Redis, Couch DB, etc) or both? Where should data be stored (on-premises, at national level, in the cloud)? ▪ Security issues: privacy preserving techniques to be implemented in ad-hoc components? Privacy by design should be a guiding principle in the design and implementation of the platform	
Synergies and Interdependencies between tasks/WP	Due to the central role of metadata, this task will interact with the methodological, the ML and the technological tasks within WP3. The activities will be carried out also in cooperation with WP2 colleagues involved in the selected pilot survey.	
Deliverables	Report on PoC - Planning - Results	

Task 3.2 – "Poc on Technical Infrastructure"

Driver *(Rationale behind the PoC, High level Goal)*	The main aim of the PoC is to verify the scalability, maintability and security of the technical infrastructure, based on the conditions written in 3.1.2 task (demonstrate that the concept of technology is feasible and has practical potential).	
Expected outcomes and impact	• Checklist of the requirements to become a smart survey (i.e., gather most of the data without user interaction). • Create a model of technical infrastructure for smart surveys and user acceptance concept. • Classification of data repositories used in smart surveys and its possible integration with other datasets. •	
PoC description	**Overview**	This PoC will be used as a reference technical infrastructure in the most general model for smart surveys. Most of the requirements from 3.1.2 task will be adapted to specify the general technical infrastructure.
	Scope	IT infrastructure requirements, Application modelling
	Assumptions	The general model will be based on the documentation for the smart surveys currently in the use.
	Methods	System adoption – like UTAUT (user acceptance) if possible (based on the surveys/feedback). Best technology used, e.g. Android, iOS, NoSQL Testing experience
	Data sources	Data structure from smart surveys based on documentation Feedback from users Pilot smart survey
	Success criteria	General technical model of smart survey application Specification of requirements for the technical infrastructure (e.g., data repository technology, mobile application platform)
	Risks	Probably there can be difficulties in accessing the data directly from smart surveys and some information in the model may be based on assumptions.
Roadmap	1. Identification of recommended technical infrastructure (i.e., platforms etc.). 2. Preparation of the general model. 3. Looking for the results from user feedbacks to prepare UTAUT user acceptance analysis. 4. Create of the set of technical infrastructure requirements.	
Synergies and Interdependencies between tasks/WP	The results will be based on documentation and results delivered by WP2.	
Deliverables	Report on PoC - Planning - Results	

Task 3.2 – Sub-task 3.2.3 "Poc on Application of Privacy Enhancing Technologies to TSS "

Driver (Rationale behind the PoC, High level Goal)	Implementation of Privacy Enhancing Technologies(PET) in the context of Smart Surveys could enable collection of new sources of data. Possibilities should therefore be investigated.	
Expected outcomes and impact	• We expect to produce a simple example of a pipeline enabling private computation on data collected by smart surveys. • Expected architecture will involve a two-step process defined by initial pre-processing on survey participant devices for data minimization purposes and to reduce the computational cost down the line, followed by private computation using privacy enhancing technologies (e.g. SMPC, Homomorphic Encryption) .	
PoC description	**Overview**	Privacy Enhancing Technologies are an essential part of the trust aspect of TSS. Currently, survey participants are asked to trust NSO's with their data. However, given the wealth of information present in smart device data, risks involved in sharing these data is high. Blindly trusting NSO's with these data might be too much to ask of survey participants. Privacy enhancing technologies could bridge this gap when trust in these technologies is established. To investigate the applicability of PET to smart surveys, a proof of concept showcasing a simple, but representative architecture should be done.
	Scope	Investigate the possibility of (1) local pre-processing on the participants device to minimize data and prepare for privacy enhanced computation and (2) subsequent private computation of simple statistical output(e.g. average, total, etc.) using pre-processed data, without individual records being exposed.
	Assumptions	Methodological considerations permit use of technologies necessary to achieve the goals set within the PoC.
	Methods	- Local pre-processing on survey participant devices using standard methods, possibly some inference using pre-trained machine learning models. - Privacy enhanced computation will probably involve Secure Multi-Party Computation or Homomorphic Encryption.
	Data sources	Ideally, data collected by WP2 pilots. If this turns out to be problematic for some reason, a suitable substitute should be found. Perhaps in the form of synthetic data.
	Success criteria	PoC is considered succesful when both: - Insights are obtained with regards to local data minimization and preparation on the survey participants device. - Insights are obtained with regards to the feasibility of application of PET to data collected with smart surveys, given that sufficient data preparation is done.
	Risks	High technological difficulty
Roadmap	TBD	
Synergies and Interdependencies between tasks/WP	Data from WP2 smart survey pilots should ideally be used. Methodology developed in WP3 should provide input with respect to the type of pre-processing and desired statistical output resulting from privacy enhanced computation.	
Deliverables	Report on PoC - Planning - Results	