**///** /Thematic Workshop 2 – OSLO TRAPEZE/

**////////////////////////////////////////////////////////////////////////////////////////////////////////**

**Date**: 21/10/2021

**Time:** 13:00 – 15:30

**Location**: Online – Microsoft Teams Meeting

# Attendees

* Digitaal Vlaanderen
  + Frédéric Hennequin
  + Lauro Vanderborght
  + Yves Meersmans
  + Michael Geamanu
* Datanutsbedrijf: Filip Borloo
* Itsme: Thijs Degheldere
* Datavillage: Philippe Ducchesne
* Inrupt: Nick Mondada
* TenForce: Alexande Vasylchenko
* Doccle: Bram Lerouge
* Meeco: Jo Vercammen
* KBC: Nils Meulemans
* Fluvius: Koen Putteman
* Cipal Schaubroek: Wim Van Acker

# Agenda of the working group

| **Part 1** | Welcome: Who is who |
| --- | --- |
| **Part 2** | The reason for this trajectory |
| **Part 3** | Summary of the previous workshop |
| **Part 4** | Elaborate on use case 1 |
| **Part 5** | Elaborate on use case 2 |
| **Part 6** | Q&A and next steps |

## Part 1: Welcome

All attendees presented themselves. The input was used to put together the list of attendees.

## Part 2: Reason for this trajectory

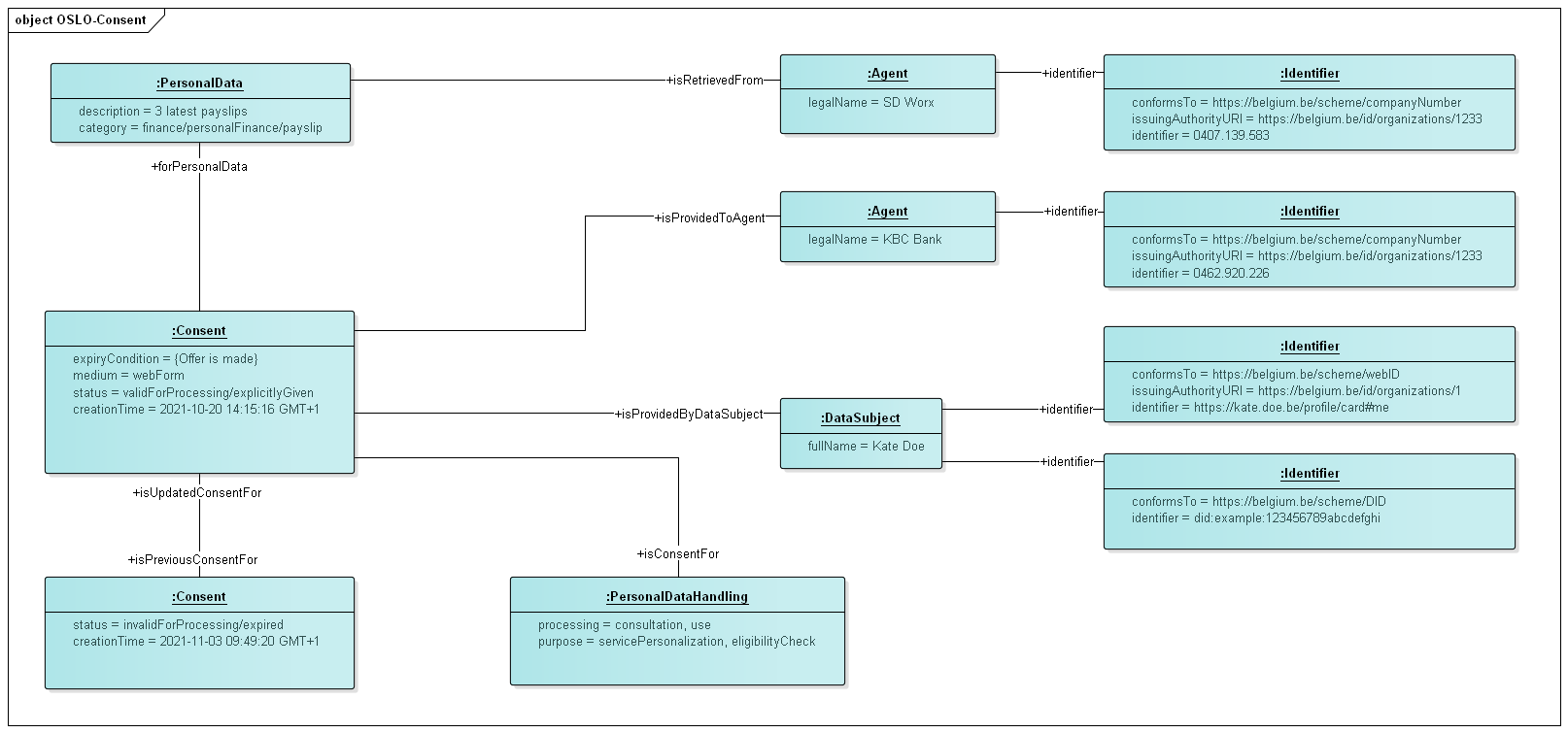
An introduction was given on the reasoning behind TRAPEZE and the focus on core consent, as shown in the presentation.

## Part 3: Summary of the previous workshop

A short summary of the previous workshop was given in the presentation. The previous workshop was the first thematic workshop of the consent trajectory, and the meeting minutes of that workshop have been shared via the invitation mail for the 2nd thematic workshop sent by Digitaal Vlaanderen.

Out of the first thematic workshop a list of issues was identified, and these were addressed during this workshop. These issues are available on [GitHub](https://github.com/Informatievlaanderen/OSLOthema-consent/issues) and everyone is open to give input with regards to the issues.

## Part 4: Elaborate on use case 1



The above context model was shown to the attendees and was demonstrated on the basis of a use case. The use case revolved around giving a bank consent to salary data for a mortgage loan. A fictitious character Kate is a customer of KBC bank. She has both a current and a savings account. Recently, she has set her eyes on a house, and thus applies for a mortgage loan at KBC. KBC requests her three latest payslips to verify her eligibility and to calculate an appropriate interest rate. Kate gives consent for this via a web form and specifies that the requested data can be retrieved from SD Worx. She also specifies that her consent will be withdrawn once the offer is made. KBC gives Kate an offer for her mortgage loan.

### The following topics were discussed

**Identifiers:** A voting session concluded that the attendees prefer to leave the options for different type of identifiers open. They expect no advantage in limiting the list in advance. It was mentioned that it is important to keep in mind that transactions between different countries should be possible. Hence, the strategy should be to define the identifier in a more global/general way. Possible types of identifiers are:

* For a person:
  + WebID
  + DID (i.e., decentralised ID)
  + National Registration Number
  + eID (i.e., electronic identification)
  + URN (i.e., Uniform Resource Name): rather a way to formulate the identifier than an identifier on itself
* For a company:
  + VAT Number
  + GLEIF (i.e., Global Legal Entity Identifier Foundation)

**Legal validity:** We will gather more input on this part of the topic in the future workshop.

**Granularity:** It is important to protect people by preventing the requester from asking for more data than needed.

* A good starting point could be to differentiate necessary and optional data.
* The granularity of the consent should be linked to the purpose and the data should only be used for this specific purpose.
* It is important to define the standard in which the data is shared when asking for consent.
* It will be complex to explain to an end-user what will be shared with the requester if we go for a high granularity. This should be looked at in the future.

**Purpose:** The purpose is the actual service you want to get. A relationship could be broader than a purpose and help to fulfil a specific transaction. There are 2 objectives linked to the purpose:

1. define which information the requester needs; and
2. define what can be done with the information that is shared.

**Expiry condition:** Expiry condition needs to be added to the draft model. GConsent has some useful parts on the expiry of a consent.

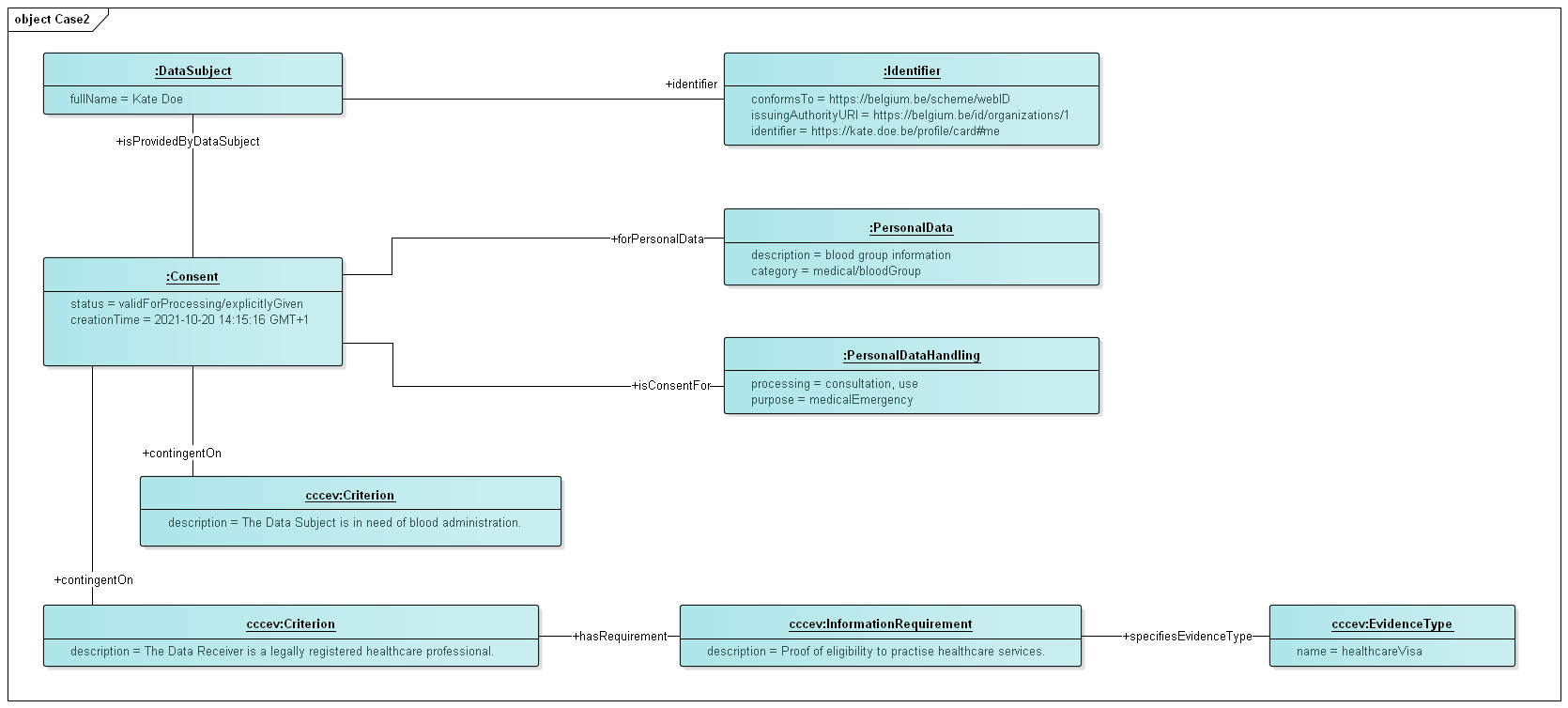
**Complexity of the request:** How do we bring the request to the data-agent in a human readable way?

* A concise and detailed version of the consent needs to be provided. The user should be able to choose which version he reads.
* The concise version should not be more than 2 paragraphs and has everything in it (purpose, time, frequency, data, type of request…). It would be ideal to work with some clear guidelines on what should be in it and how it should be formulated. This should help for the end-user to get used to the form of the request and understand the consent more easily. It is important here to standardise the obliged field (e.g., always put a date as *DD/MM/YYYY*).
* In the detailed version, a breakdown should be given of the origin and the content of the data.

**Consent:**

* The possibility to add ExpiryTime from GConsent to the consent-class is needed.
* Datavillage also keeps the chain of consent. Meaning that they store every version of consent to keep track of all the changes. They have the most recent one as the first one in their chain.

## Part 5: Elaborate on use case 2



The above context model was shown to the attendees and was demonstrated on the basis of a use case. The use case revolved around giving consent to blood group information in case of a medical emergency. In this use case, Kate got involved in a car accident while on her way back from the bank. She was rushed to the hospital and is in urgent need of a blood administration. Kate is unconscious, but luckily, she has already given consent to access her blood group in the past in case of medical emergencies with regards to healthcare professionals.

### The following topics were discussed

**How do we identify an accident?** This is about the granularity of the purpose. It would be good to set criteria on this. Therefore, Digitaal Vlaanderen proposes to use the CCCEV data model which is a European standard for organisations to set the requirements of a certain service. It helps to standardise the wide possibilities of contexts for which a consent could be needed.  
It is not entirely clear who/how CCCEV checks or makes sure that the criteria are met. This will be checked.

## Part 7: Q&A and next steps

**Q: Will it be allowed to change a consent, e.g., because of a human error?**

* Case 1: If the content is changed by the requester, the consent should circle back to the agent who then should sign it again. Another possibility is to make a new consent in this case.
* Case 2: The agent giving the consent should be able to change the status to *expire* or *withdrawn* any moment in time.
* The status of the consent is the result of an event and the agent must sign that the event has happened. Status should not be part of the consent but something that happens to it.

**Q: Is there anything that describes the encryption of the data?**

* The possibility to add this to the model should be checked. In this case, the goal should be to give a data subject the possibility to have the data transfer encrypted.
* Existing models should be checked to see if something comparable is in there.
* At this moment, most data are being encrypted. The level of encryption that is expected could be specified. The challenge here is again how this will be made understandable for the end-user. It may be too technical for a user to understand, so maybe it is not relevant.
* Usually, one can only oblige people to include or exclude certain attributes. The inclusion or exclusion of attributes (and how this is done) is considered at application/integration level. Hence, it is probably out of scope for the data model.

**Q: How will the agent be able to check what (s)he has given consent to?**

* This is something that will not be put in the data model itself.
* One possibility is to get the instance of an application to show an example of the data that someone will share.
* SOLID has the idea to create a [PodBrowser](https://inrupt.com/products/podbrowser/) which will tell you all the data sources and the different consents you have given to these sources. This would be a more technical thing. A central place with all the consents in it.
* In most situations you will see the consent receipt, but you will not see the data itself via this receipt.

To close the workshop, an overview was given of the next steps for the coming months:

* Process the input from the workshop
* Circulate the main findings of this workshop
* Further research and prepare the third thematic workshop
* Capture further input through GitHub

If you would like to participate in one of the following thematic workshops, you can find an overview of the workshops and register via the link below. The next thematic workshop will take place on 18/11/’21 at 13h via Microsoft Teams. The link will be sent to the participants after registration.

<https://overheid.vlaanderen.be/opleiding/oslo-trapeze>

In the meantime, if you have any questions or notice a problem, you can always open an issue on GitHub or send an e-mail to the e-mail addresses below:

* laurens.vercauteren@vlaanderen.be
* michael.geamanu@pwc.com