MyData Authorisation Specification

Notice

Implementation or use of certain elements of this document may require licenses under third party intellectual property rights, including without limitation, patent rights. The Participants of and any other contributors to the Specification are not and shall not be held responsible in any manner for identifying or failing to identify any or all such third party intellectual property rights. This Specification is provided "AS IS," and no Participant makes any warranty of any kind, expressed or implied, including any implied warranties of merchantability, non-infringement of third party intellectual property rights, and fitness for a particular purpose.

MyData Architecture defines the operations and APIs between the Operational Roles (Operator, Source, Sink etc.). Any descriptions or figures of the role’s internal structure or operations are for illustrative purposes only.

1. Introduction

This document specifies MyData Authorisation and its management over its lifetime.

This document is part of the MyData architecture release 1.1. It is assumed that the reader is familiar with MyData Architecture - Consent Based Approach for Personal Data Management document available at https://hiit.github.io/mydata-stack/.

Known deficiencies in this release: missing JSON structures, API naming, message formats, and error messages. These will be part of the next release of this document.

1.1 Terms and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

1.2 Terminology

Key terminology used in this specification is defined in the Glossary of MyData Architecture - Consent Based Approach for Personal Data Management release 1.1 available at https://hiit.github.io/mydata-stack/.

1.3 Authorisation and Consent

Processing of personal data requires a legal basis and there are several possible bases. In MyData architecture, all data processing is based on consents from the Account Owner (data subject), and it is possible to change or withdraw the consent at will.

It should be noted that consent does not legitimise all sorts of processing activities, nor can it negate obligations stemming from general principles related to processing of personal data. Consent is *one possible* ground for processing personal data and does not always constitute the most appropriate ground. However, when used appropriately it provides a tool for the Account Owners to control their data.

In this document the term processing is used to describe all data collection, movement and processing.

2. MyData Consent Model

This section describes, how consents have been implemented in MyData Architecture.

2.1 MyData Consent Record

When Account Owner issues a consent, in MyData Architecture it is documented in a ***MyData Consent Record (CR)***. For authorising data processing within a service, the Account Owner creates a single Consent Record for the related service. For authorising data transfer from a specific Source to a specific Sink, the Account Owner creates a pair of Consent Records (one for the Source and one for the Sink). Then, the Source's Consent Record defines, what data can be provisioned to the specified Sink, and the Sink's Consent Record defines, how the data can be accessed. The Sink's Consent Record can also include the permissions for data processing.

A Consent Record is a manifestation of legally valid Consent and makes it technically feasible to change or withdraw the consent dynamically. Consent Records are stored in the MyData Account and at the related service.

2.2 Resource Sets

Consents refer to Account Owner's data using Resource Sets. A Resource Set defines a specific (subset) of the Account Owner's data that a particular Source provisions or processes. Source can provision subsets of the original data set in different ways by defining different Resource Sets. Service Registry contains a description of all data a service provisions or processes. Account Owner creates a Resource Set when creating a consent by selecting a subset of data for processing.

Resource Set is identified using Resource Set ID, rs_id. It MUST be unique within the data processor and can be used only in one consent at the time. This means that if two consents refer to same Resource Set they MUST use different Resource Set IDs. Resource set identifier is composed of a URI that identifies the Source and resource key unique within Source that identifies the resource inside the Source. URI and the resource key must not leak out any specific information about the user or data but only provide reference to the Source e.g. com.example.a3h413h4b13h41. By using rs_id's, both parties can refer to a specific resource set and Source using the rs_id as an globally unique identifier.

Table 1: Resource Set ID

Source URI	URI identifying Source
Resource Key	Unique key within a service identifying specific resource set.

2.3 Consent Lifecycle

Consent Records have four valid states:

- Active, Account Owner's data can be processed according to rules set in consent
- Paused, Data processing is not allowed
- No Service Link, Data processing is not allowed
- Withdrawn, Data processing is not allowed

When issued, consents are in Active state. Account Owner can set a consent to Pause state if they want to temporarily disable data processing. Similarly Account Owner can set a Pause-state consent to Active-state when they want to re-enable data processing. Account Owner can also withdraw consent (from any other state) at will by setting its state to Withdrawn. A Withdrawn Consent cannot be reactivated, a new consent must be created in it's place. Finally, Operator sets a consent to 'No Service Link' state when the Service Link between the related service and Account Owner is modified and the Service Link modifications have impact on Consent. Account Owner has to review the changes, after which the consent can be switched to Active, Paused or Withdrawn state. All states and allowed transitions are shown in Figure 2.1.

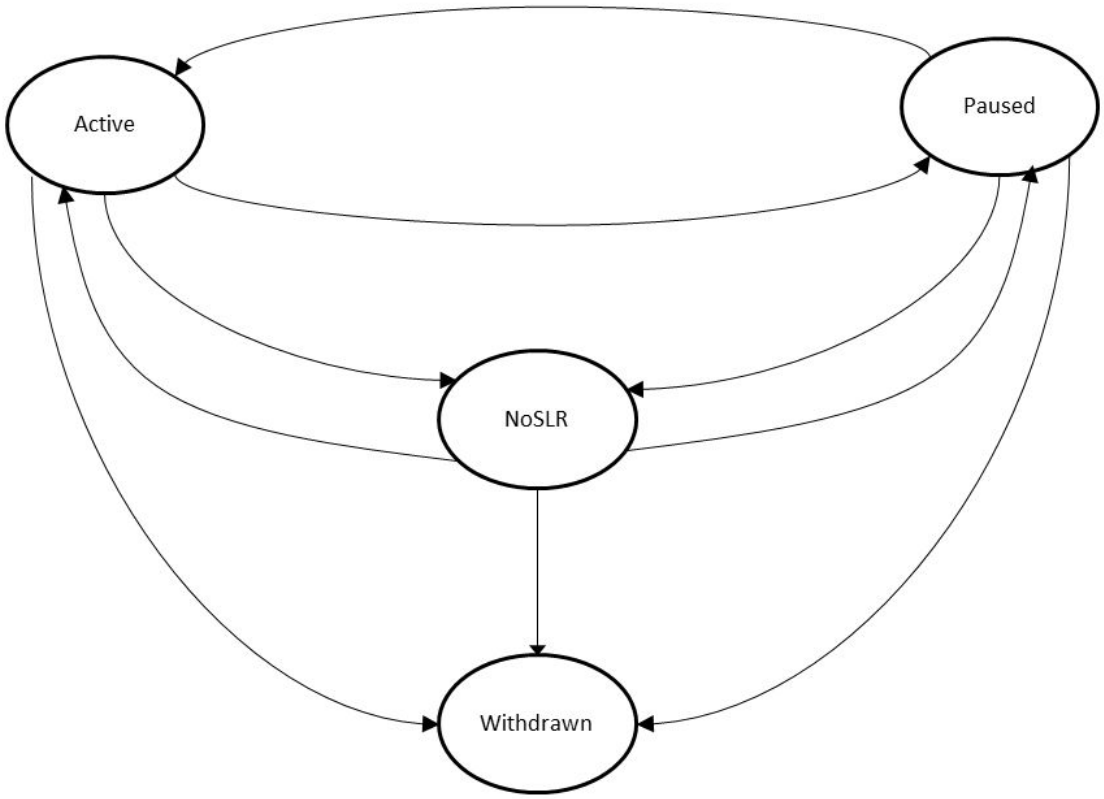

Figure 2.1: Consent lifecycle

When Account Owner changes the status of the consent given to a Sink, also the status of the related consent given to related Source MUST be updated, to ensure it doesn't provide data due to accidentally sent 'late' data requests from the Sink. If Account Owner changes the status of the consent given to Source, it is sufficient to notify the Source only.

When Sink related Service Link is changed, both consent given to Sink and related consent(s) given to Source(s) are set to No Service Link state. If Source-related Service Link is changed, it is sufficient to put Source consent to No Service Link State.

2.4 Consent Status Record

Account Owner or Operator can modify a consent's state by issuing *Consent Status Records*, which is stored at the at the Operator and sent to the related service.

Source and Sink MUST verify the signature of the Consent Status Record and reject status changes that are not signed with one of the keys contained in Service Link Record.
Source and Sink MUST also verify that the Previous Record ID field equals to previous Status Change Record ID in the Consent Status Record Log. Services MUST maintain local copies of Status Change Records.

If the service can not verify the Status Change Record it MUST stop data processing based on the consent and update the status of this consent via sending Consent Status Record Request to the Operator.

2.5 Consent Record Verification and Validation

Consent MUST be verified when service receives it from the Operator.
Verification steps consist of:

1. verify consent is issued by authorised party
2. verify consent has not been corrupted
3. verify that is well-formed and contains all mandatory information
4. verify associated Consent Status Record

Consent is verified if, and only if all the verification steps pass.

Consent MUST be validated every time when the data is processed based on the consent.
Consent is valid if:

1. Time of use is between not before and not after timestamps; AND
2. Consent status in latest Status Change Record is Active

It is assumed that the Operator will deliver Consent Status Records immediately to relevant services. If the service is uncertain whether it has the latest Status Record it can request the latest record from the Operator.

3 MyData Authorisation Transactions

This section describes MyData Authorisation transactions. There are 4 main transactions:

- Issuing consent
- Withdrawing a consent
- Modifying consent status
- removing a linked service

Modifications to an existing consent are implemented by withdrawing the existing consent and issuing a new consent based on modifications made by Account Owner.

3.1 Account Owner Issues Consent

Prerequisites: There is a Service Link between Account Owner and service
Process: Account Owner issues consent (for data processing within a service) or pair of consents (for data connection between Source and Sink) at the Operator.
Outcome: Consent Record(s) that contain all required information [for both technical & legal perspective]. Consent Record(s) are distributed to relevant parties. Authorisation token is delivered to Sink. For detailed management and lifecycle of Authorisation token see MyData Data Connection specification.

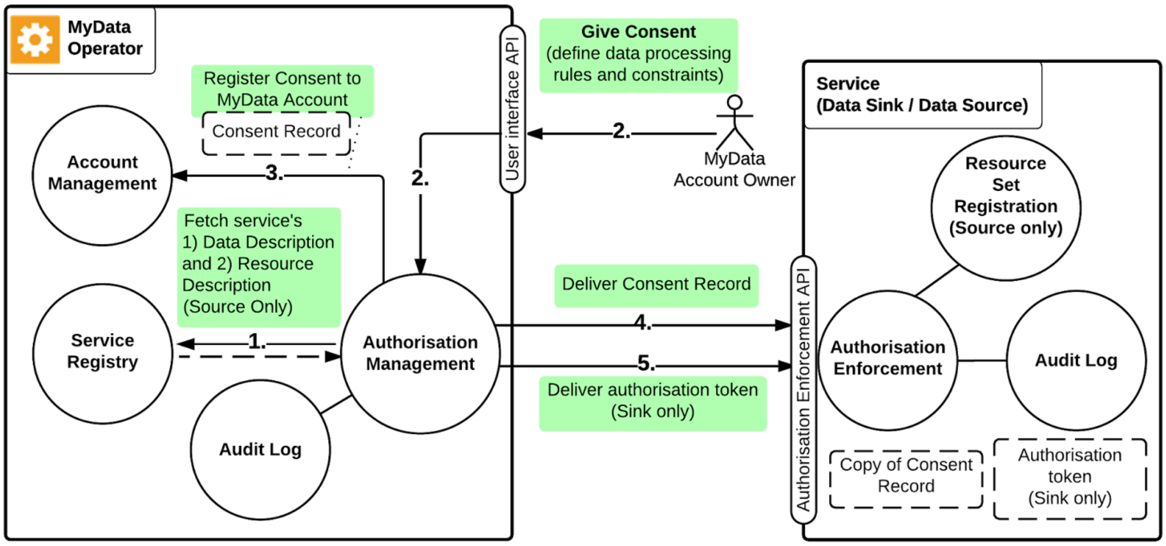
MyData Operator
Register Consent to MyData Account
Consent Record
Account Management
3.
2.
User interface API
Give Consent (define data processing rules and constraints)
2.
MyData Account Owner
Service (Data Sink / Data Source)
Resource Set Registration (Source only)
Fetch service's 1) Data Description and 2) Resource Description (Source Only)
Service Registry
1.
Authorisation Management
Audit Log
Deliver Consent Record
4.
5.
Deliver authorisation token (Sink only)
Authorisation Enforcement API
Authorisation Enforcement
Audit Log
Copy of Consent Record
Authorisation token (Sink only)

Figure 3.1: Authorisation flow. For details of step 5, creation and delivery of Authorisation token see MyData Data Connection document.

After a Source and a Sink have been linked at MyData Account, the Sink can be authorised to access data on the Source by conducting MyData Authorisation step. The Authorisation results in a pair of Consent Records. In the first, Account Owner gives an explicit consent for particular Sink to access a specific Resource Set on the Source (depicted in the Source's Consent Records). In the second, Account Owner also authorises Sink to process the specified data according to terms defined in the Sink's Consent Records. The contents of a Consent Records are depicted in *section 4.1*. So both Source and Sink have their own Consent Records, which contain role specific information necessary in establishing a Data Connection between Source and Sink.

Upon creation of Consent Record, Operator delivers both Records to the corresponding services as shown in *Figure 2* (Sink's Record is delivered to Sink and Source's Record is delivered to Source).

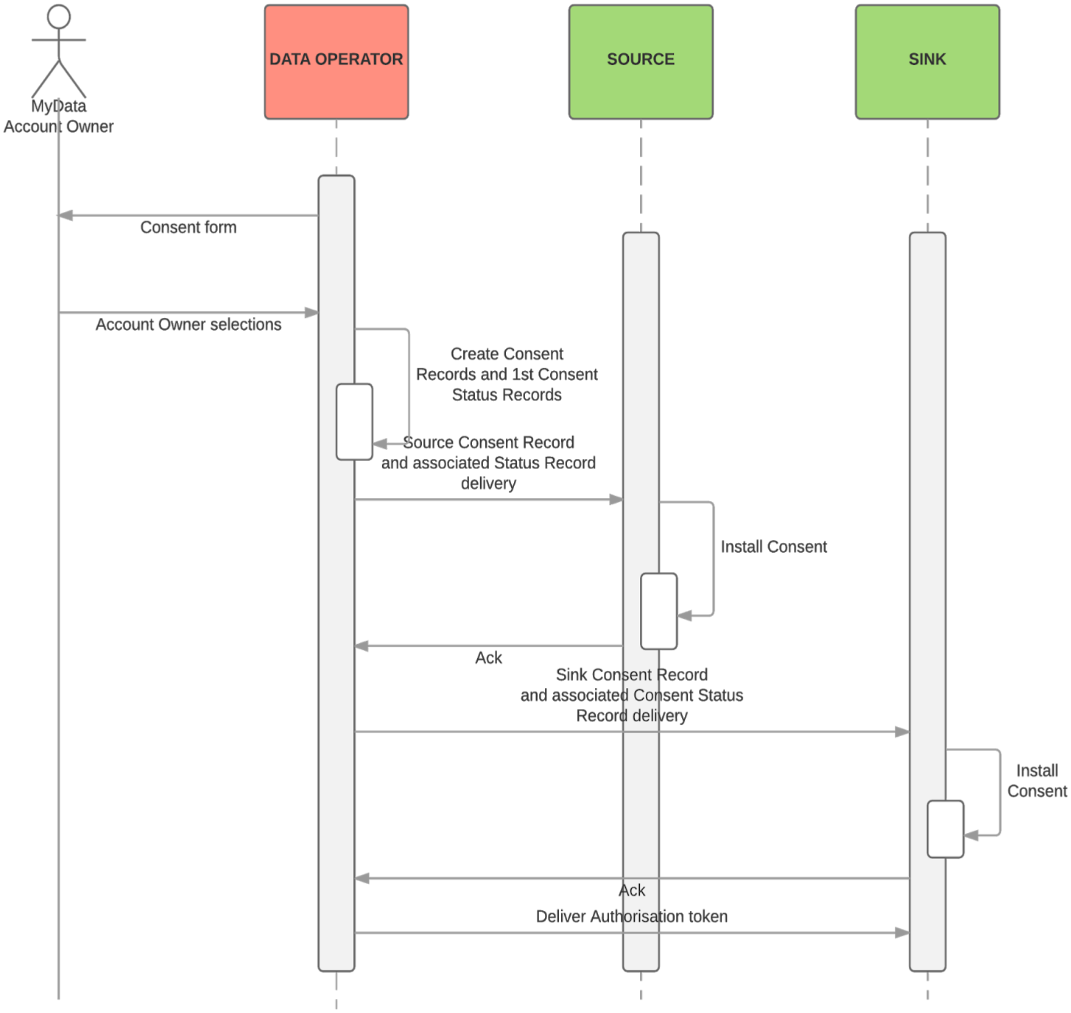

Figure 3.2: Consent creation and delivery

3.2 Account Owner Withdraws Consent

Prerequisites: Consent or pair of consents exists
Process: Account Owner withdraws previously issued consent or consent pair thus disabling further data processing.
Outcome: Consents are in Withdrawn state and further data processing based on the consent cannot legally happen.

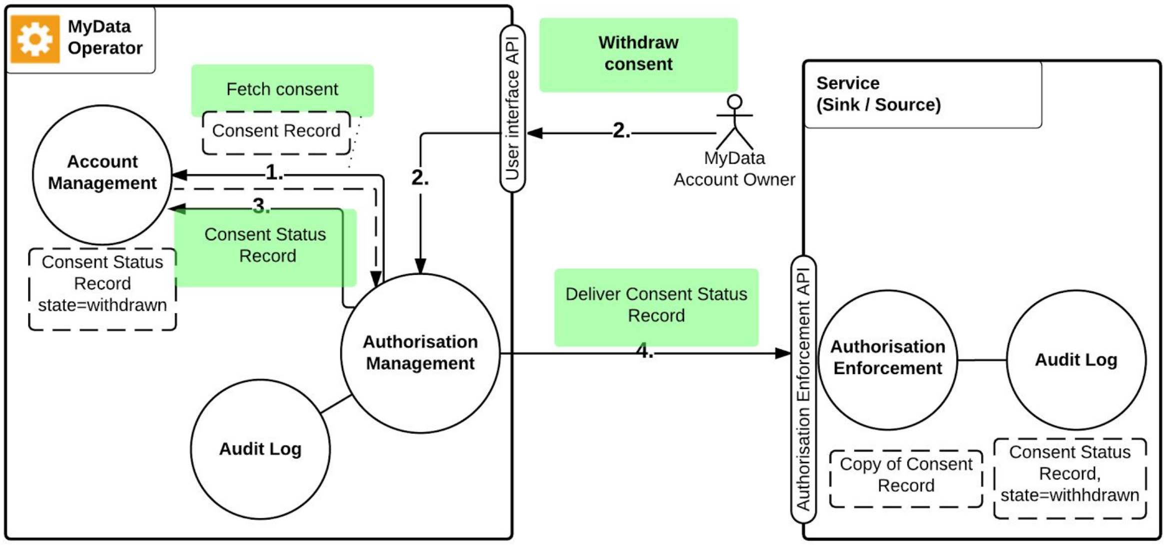

Figure 3.3: Consent withdrawal

Account Owner may withdraw the consent issued to a Sink or Source at will, at which point the affected Source and Sink MUST be informed about withdrawal as soon as possible. Upon receiving such notification from the Operator, Source MUST immediately stop processing request made by the affected Sink. For example, if Account Owner withdraws a consent, Operator notifies Source about the change and the Source will reject further requests that concern data defined in the withdrawn consent.

If a Source is processing request at the time it receives a consent withdrawal notification, it checks if the information presented in the notification concerns the request-under-processing, and stops processing the request if necessary.

Consent Withdrawal process is shown in Figure 3.4.

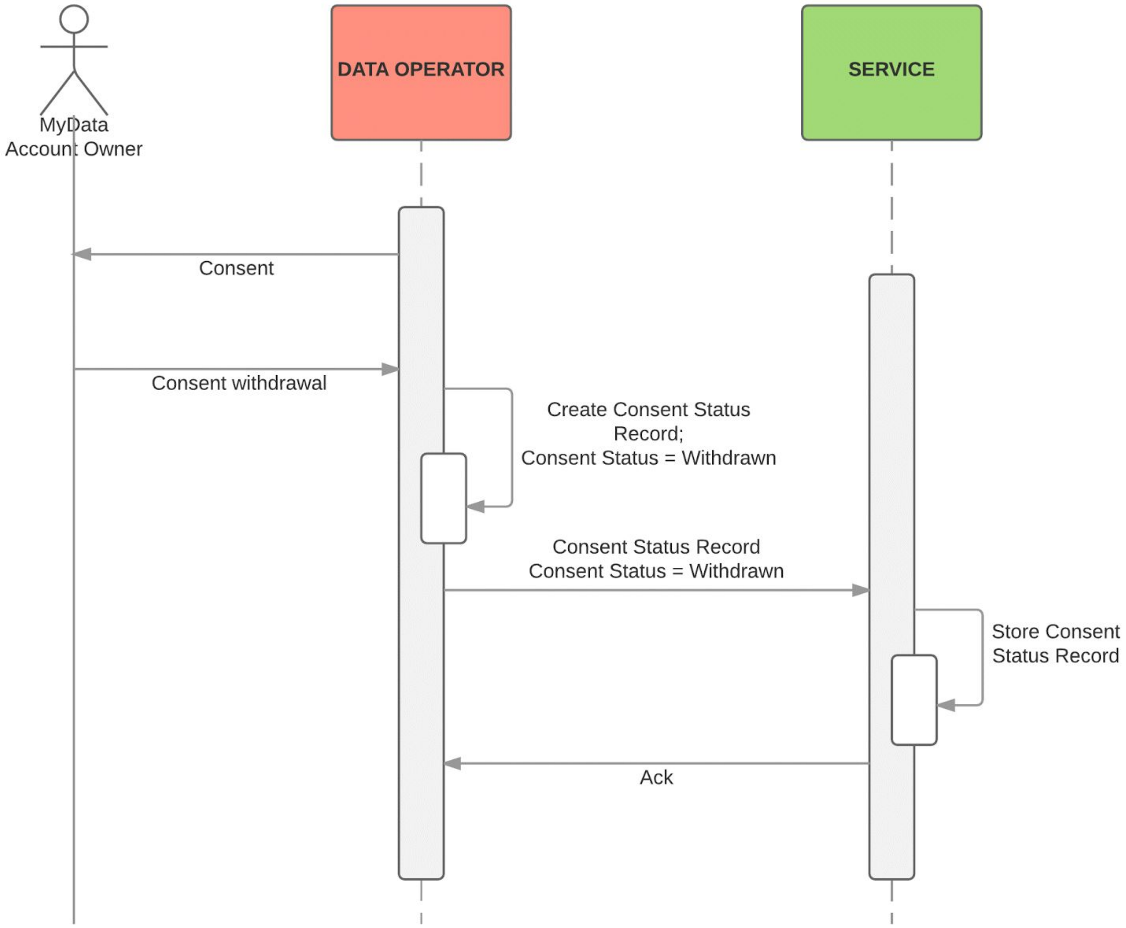

Figure 3.4: Consent withdrawal related consent status change propagation flow

3.3 Consent Status Change

Prerequisites: Consent or pair of consents exists
Process: Account Owner changes the consent status at Operator.
Outcome: Consent status is changed and data processing is either enabled or disabled based on the new status.

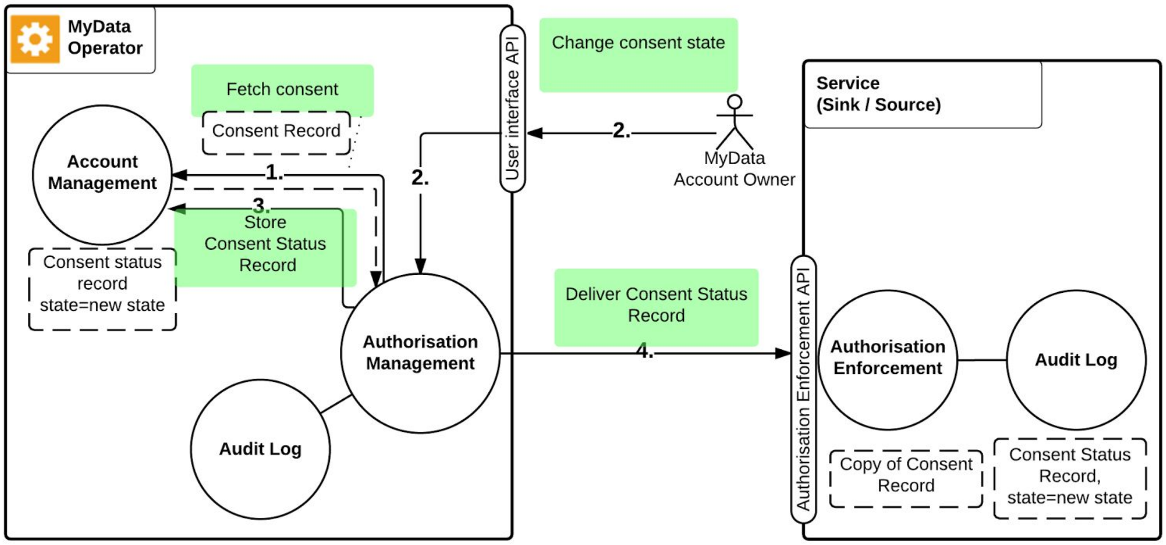

Figure 3.5: Consent status change flow

Account Owner may pause or re-activate the consent given to a Sink or Source at any time, at which point the affected Source and Sink MUST be informed about these changes as soon as possible. Source MUST immediately act and change its processing of request made by Sink, according to information presented in the notification. For example, if Account Owner pauses a consent, Operator notifies Source about the change and the Source will reject further requests that concern data defined in the withdrawn consent.

If a Source is processing a request at the time it receives a consent change notification, it checks if the information presented in the notification concerns the request-under-processing, and updates its request processing if necessary.

The consent status change flow is shown in Figure 3.6.

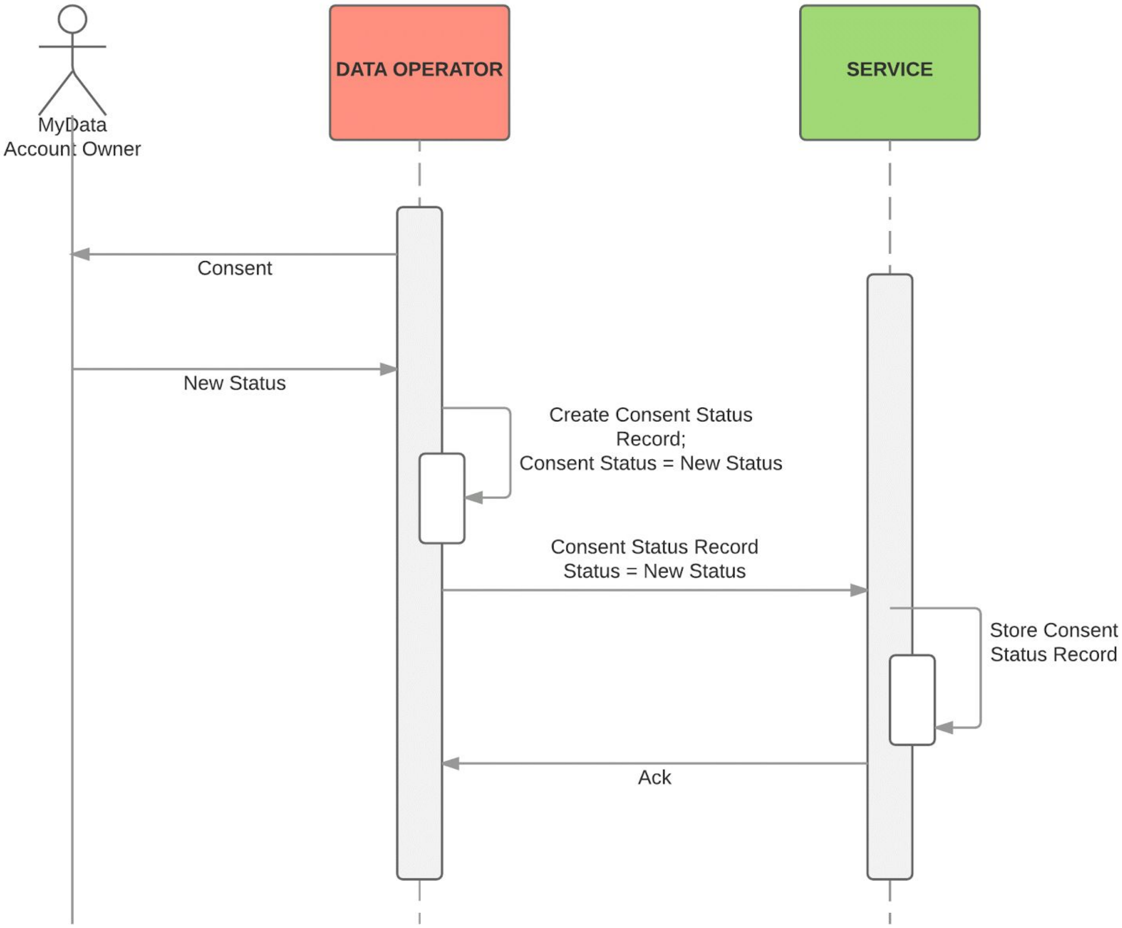

Figure 3.6: Consent status change related propagation flow

3.4 Service Link Modified or Removed, or Service Unregistered

Prerequisites: Consent or pair of consents exist
Process: When Service Link is removed due to service unlinking or Service removal, Operator sets all affected consents to No Service Link state. Consents remain in No Service Link state until service is re-linked and the Account Owner has either re-activated, paused or withdrawn the consent.
Outcome: All affected consents initially in No Service Link state, and after acceptance in state set by the Account Owner.

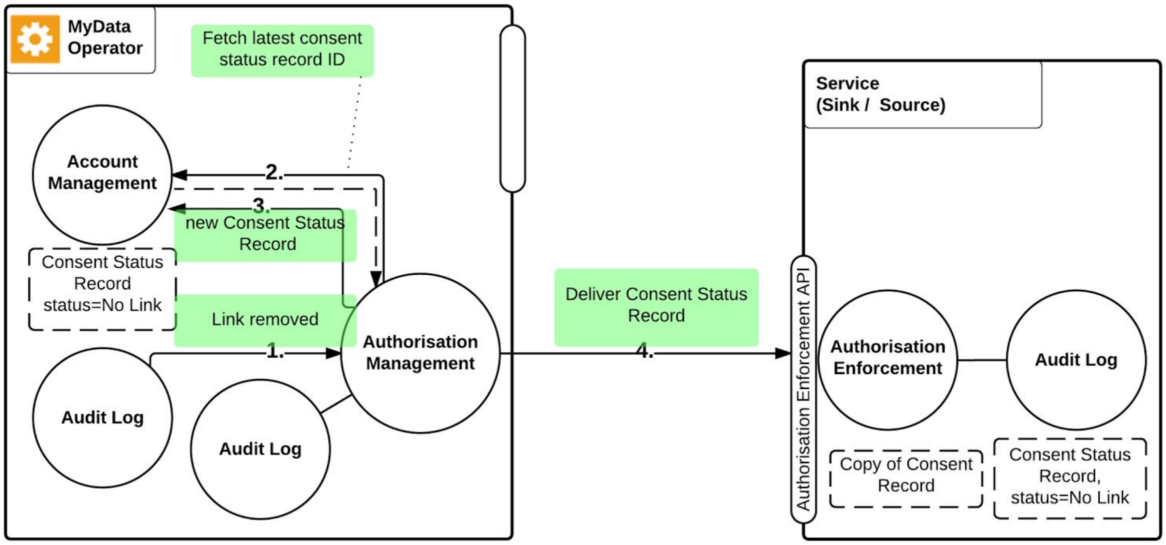
MyData Operator
Fetch latest consent status record ID
Account Management
2.
3.
new Consent Status Record
Consent Status Record status=No Link
Link removed
1.
Authorisation Management
Audit Log
Audit Log
Audit Log
Deliver Consent Status Record
4.
Service (Sink / Source)
Authorisation Enforcement API
Authorisation Enforcement
Audit Log
Copy of Consent Record
Consent Status Record, status=No Link

Figure 3.7: Link removal related flow

Service can be unlinked or removed from the Service Registry at any time, at which point the affected Source and Sink MUST be informed as soon as possible.

Upon receiving such notification from the Operator, Source MUST immediately stop processing request made by Sink. If a Source is processing request at the time it receives a service unlinking notification, it checks if the information presented in the notification concerns the request-under-processing and stops its request processing if necessary.

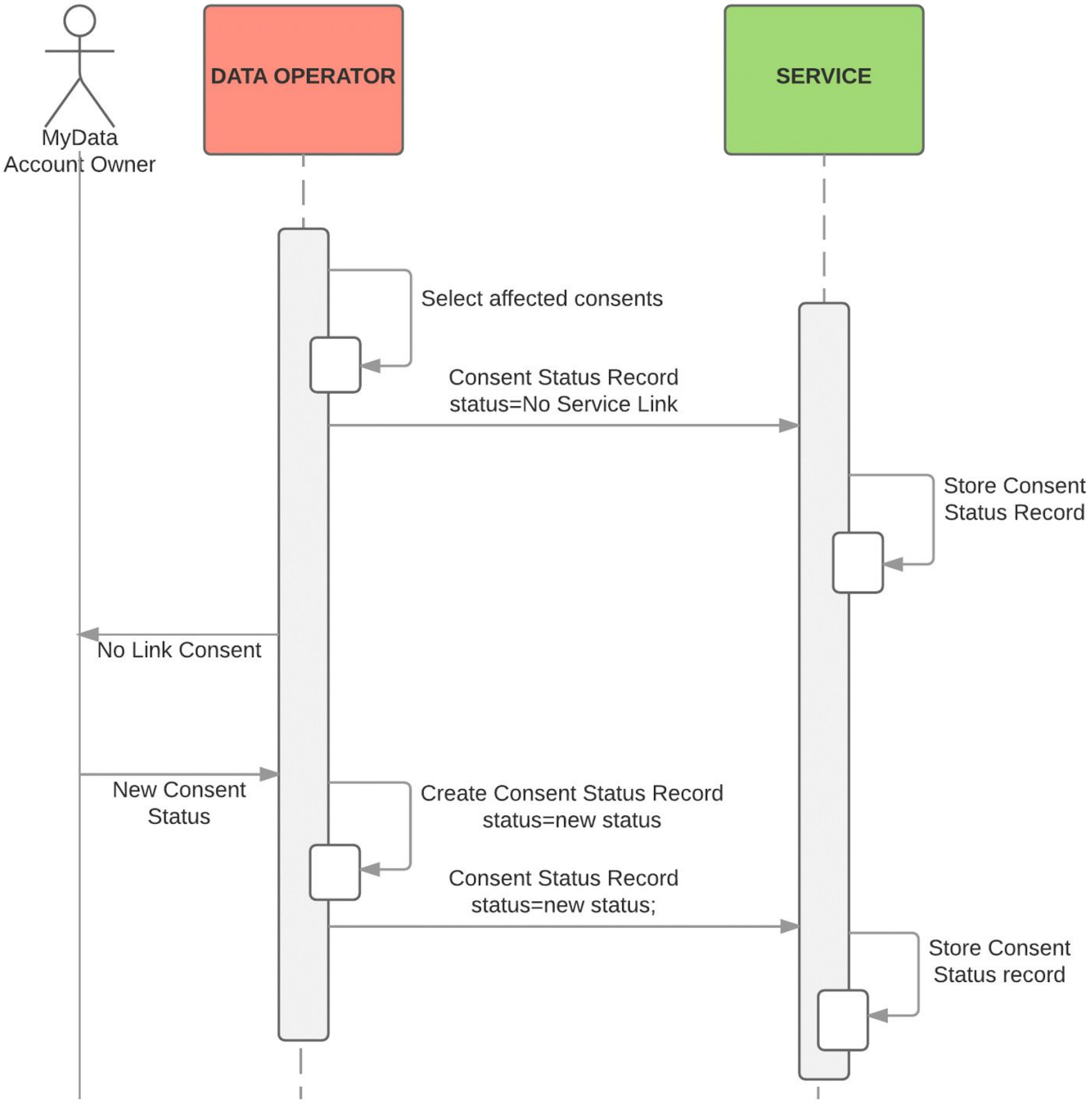

Figure 3.8: Service removal related consent status change propagation flow. This is executed for each affected consent.

4. MyData Consent Record

This section presents the structure of MyData Consent Record and Consent Status Record.

4.1 MyData Consent Record Details

MyData Consent Record consists of five parts: common part, role specific part, Minimum Viable Consent Receipt [MVCR, Kantara Initiative CISWG's work in progress] part, potential extension part(s) and signature as shown in Table 4.1.

Table 4.1: Consent Record structure

Common part
Role specific part
MVCR
MVCR extensions
Signature

Table 4.2 presents a detailed structure of MyData Consent record.

Table 4.2: Consent Record

COMMON PART	
Version number	Specification version number, for this release MUST be 1.1
Record ID	unique ID for the Consent Record
Account ID	the Account Owner's surrogate ID (see Service Linking document)
Resource Set ID	Identifies resource set on Source
Link ID	Service Link ID
Issued	Time (UTC) when the consent was issued
Not before	Time (UTC), consent not valid before, OPTIONAL
Not after	Time: (UTC), consent not valid after, OPTIONAL
Issued at	ID of the Operator where the consent was created

Subject ID	ID of the service to which the consent was issued
Role specific part	
Role	Sink **OR** Source **OR** Internal Processing
Authorisation token issuer key	[SOURCE ONLY]: What key will be used to digitally sign the Authorisation token that is given to a Sink.
Resource set description	[SOURCE and INTERNAL PROCESSING] Resource set description
Usage rules[1..n]	[SINK and INTERNAL PROCESSING] How the data can be processed, in machine readable form String [1..*]
MVCR PART	
MVCR	MVCR, see Appendix A
EXTENSION PART	
	NONE FOR THIS RELEASE
Digital signature	Account Owner's signature

4.2. Resource Set Description

Resource sets are described using Resource Description Framework (RDF), see Data Description in the MyData Service Registry Specification.

4.3 MyData Consent Status Record

Table 4.3 presents the structure of the MyData Consent Status Record.

Table 4.3 Consent Status Record

Record ID	Unique ID of the record
Account ID	the Account owner's Surrogate ID
Consent Record ID	Unique ID of the consent
Consent status	Active/Paused/Withdrawn/No Service Link

Timestamp	Time when the Status Record was issued
Previous Record ID	Link to previous Status Record ID, NULL if first Status Record
Signature	MUST be signed with one of the keys in the related Service Link Record

4.4 Status Record Log

Status Record Log stores all Status Records and is described in Table 4.4.

Table 4.4 Consent status change log

Consent Status Records[]	a latest-first ordered list of Status Records associated with specific consent

5. Consent APIs

Exact APIs to be used for authorisation will be released in upcoming MyData SDK, which is expected together with MyData architecture (minor) revision 1.2.

5.1 Interfaces of Different Actors

There are two main interfaces:

- Authorisation Management API
- Authorisation Enforcement API

Both Source and Sink MUST provide an Authorisation Enforcement API for Operator, through which it can deliver Consent Records and Consent Status Records.

Operator's Authorisation Management API defines methods for Sink and Source to request Consent Records (referenced using surrogate ID) and Consent Status Records (referenced using unique Consent ID and last known Status Record ID) and to verify that the service has the latest Consent Status Record.

References

[RFC2119] Bradner, S, "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
[MVCR]: https://github.com/KI-CISWG/MVCR/blob/master/MVCR%20v0.7.1.md

APPENDIX A: MVCR 0.7 Version

Knowing the work-in-progress state of MVCR, below is listed the current draft structure the CISWG is documenting for the consent record. As the JSON structure of the consent record is frozen, this appendix will be refreshed along with a minor revision of the MyData architecture.

Field Name	**Data Type**	**Description**	**Example Input**
Section 1: Header	------	This is the first section of the receipt	-------------
jurisdiction	string. ISO two-letter country code if applicable, otherwise free text	This is the legal jurisdiction under which the processing of personal data occurs	US
iat	number. Integer number of seconds since 1970-01-01 00:00:00 GMT	Timestamp of when the consent was issued	1435367226
iss	string. HTTPS URL	This is the URI or Internet location of processing, i.e. one party-two party or three	http://www.consentreceipt.org/
jti	string.	Unique identifier for this consent receipt	9ef6b81a414b2432ec6e3d384c5a36cea8aa0c30d3dd2b67364126ed80856f9c20654f032eef87ad981187da8c23c1186eefe1503714835c2e952bbb3f22729c
sub	string.	Subject provided identifier, email address - or Claim, defined/namespaced	example@example.com
Section 2: Data Controller	----------	This section has the data controller, contact and privacy service information	----------------------
data_controller	object	The identity and company	{"on_behalf": true, "contact":

		of the data controller and any party nominated to be data controller on behalf of org	"Dave Controller", "company": "Data Controller Inc.", "address": "123 St., Place", "email": "dave@datacontroller.com", "phone": "00-123-341-2351"}
		The object contains information of the data controller in the following fields:	
		Field Name Data Type Description Example Input Required on_behalf boolean. acting on behalf of an organization? true contact string. person to contact Jon Doe company string. company name Data Controller Inc. address string. physical address 123 Main St., Anywhere email string. Email address contact email address jon@datacontroller.com phone string. Phone number contact phone number 00-000-000-0000	
policy_uri	string. HTTP URL	the internet and immediately accessible privacy policy of the service referred to by the receipt	http://example.com/privacy
Section 3: Purpose Specification			
purpose	array of strings.	Explicit, Specific and Legitimate: interpreted here as: 'Naming the Service' and 'Stating the Active Purpose '	[Bob’s store, delivery,]or [[" CISWG Membership", "Join"]

Section 4: Personal Information			
pi_collected	object. Keys are the name of the field, value is the information collected.	Personal information collected in relation to, or adjacent of purposes specified	{"name" : "Example Example", "email" : "example@example.com"}
sensitive_pi	array of strings.	In many jurisdictions their are additional notice and administrative requirements for the collection, storage and processing of what are called Sensitive Personal Information Categories. These are Sensitive in the business, legal, and technical sense, but not specifically in the personal context. This list of categories are required in some jurisdiction, but, the actual notice and purpose requirements are out the scope of the MVCR.	{"health"}
Section 5: Information Sharing		Sharing information with 3rd parties, what categories, with whom, and how information is shared	
sharing	array of strings.	This refers to the sharing of personal information collected about the individual, with another external party by the data controller (service provider). Should list categories of PII shared, from above list and under what purpose. Sharing is also a container for listing	[?]

		trust marks and trust protocols.	
In Review			
aud	string. HTTP URL	Audience URI that identifies the target service of this consent	http://engageidentity.com/protected
consent_payload	object. Keys are the name of the consent, values are whether or not the user has agreed.	Examples include: Device Identifier, UID, IP Address, Browser Fingerprint, DNT signal client header, .Mobile device id	{"privacy policy" : "agree","ToS" : "agree"}
context	array of strings.	Operational Context refers to the conditions that ensure the consent is fair, reasonable and proportional. , e.g. if it is on a website, then there are requirements like; are mandatory fields indicated, is there a separate consent for privacy policy and terms of service? set of registry values?	["active privacy policy consent", "passive terms of service consent"]
notice	string. HTTP URL	Link to the short notice enables usability and layered policy. to provide enhanced transparency about data collection and information sharing practices	http://example.com/notice
scopes	string. space separated string values	What you’re allowed to do on the service (these can be tied to legal / business / technical layers)	read update