SR 816.111

Ergänzung 2.1 zu Anhang 5 der Verordnung des EDI vom 22. März 2017 über das elektronische Patientendossier

Nationale Integrationsprofile nach Artikel 5 Absatz 1 Buchstabe c EPDV-EDI

Privacy Policy Query for Mobile (CH:PPQm)

Ausgabe 4: 15. März 2022

Inkrafttreten: 15. April 2022

1 Introduction

Das nationale Integrationsprofil «Privacy Policy Query» (CH:PPQ) ermöglicht die Erstellung, Änderung, Löschung und Abfrage von EPD-Zugriffsregeln (Policies) durch Patienten und berechtigte Gesundheitsfachpersonen. Es basiert auf den Standards SOAP und XACML 2.0 und eignet sich somit nur bedingt für mobile Geräte. Um diese Einschränkung zu überwinden, wird im vorliegenden Dokument das nationale Integrationsprofil «Provacy Policy Query for Mobile» (CH:PPQm) definiert, welches auf FHIR® und dem REST-Kommunikationsstil basiert.

The national integration profile «Privacy Policy Query» (CH:PPQ) allows patients and dedicated healthcare professionals to create, change, delete and query EPR access policies. It is based on the standards SOAP and XACML 2.0, and thus is only restrictedly suitable for mobile devices. To overcome this restriction, the given document defines a national integration profile «Privacy Policy Query for Mobile» (CH:PPQm), which is based on FHIR® and the RESTful communication style.

1.1 Terminology

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 [RFC2119][1].

[1] For full text of RFC2119 see https://www.ietf.org/rfc/rfc2119.txt

2 Volume 1 – Integration Profiles

2.1 Privacy Policy Query for Mobile (CH:PPQm)

According to Swiss EPR regulations, patients have the right to decide who is allowed to access and modify data in their EPR, and under which circumstance (cf. emergency access). The national integration profile “Privacy Policy Query” (CH:PPQ, see Amendment 2.1 of Annex 5 EPRO-FDHA) defines how to specify these decisions as access policies in the XACML 2.0 format and interchange them using the SOAP transport protocol. For mobile applications, this combination of standards is only restrictedly suitable, therefore a more lightweight solution like HL7 FHIR® is required instead.

The national integration profile CH:PPQm — “Privacy Policy Query for Mobile” — is a FHIR-based counterpart of CH:PPQ.

2.1.1 Actors and Transactions

CH:PPQm comprises the following actors and transactions:

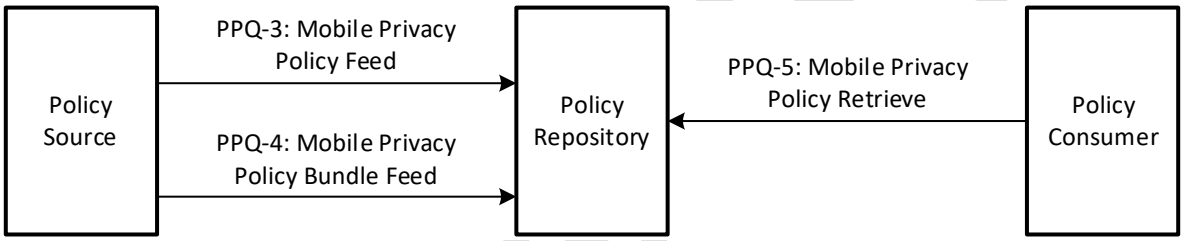

Figure 1: CH:PPQm actor diagram

Actor:	Policy Repository
Role:	Stores policies and policy sets and provides the possibility to add, query, update and delete them
Actor:	Policy Source
Role:	Initiates addition, update and deletion of policies and policy sets
Actor:	Policy Consumer
Role:	Retrieves policies and policy sets

Table 1: CH:PPQm actors and roles

Table 2 lists the transactions for each actor directly involved in the CH:PPQm Profile. To claim compliance with this profile, an actor shall support all required transactions (labeled “R”) and may support the optional transactions (labeled “O”).

Actors	Transactions	Optionality	Section
Policy Repository	Mobile Privacy Policy Feed (PPQ-3)	R	3.1
	Mobile Privacy Policy Bundle Feed (PPQ-4)	R	3.2
	Mobile Privacy Policy Retrieve (PPQ-5)	R	3.3
Policy Source	Mobile Privacy Policy Feed (PPQ-3)	O (Note 1)	3.1
	Mobile Privacy Policy Bundle Feed (PPQ-4)	O (Note 1)	3.2
Policy Consumer	Mobile Privacy Policy Retrieve (PPQ-5)	R	3.3

Table 2: CH:PPQm transactions

Note 1: The actor SHALL support at least one transaction.

The required actor groupings are shown in Table 3:

Actors	Actor to be grouped with	Reference
Policy Repository	IUA Resource Server	Amendment mHealth of Annex 5, section 2.3.1
	ATNA Secure Application	Amendment 1 of Annex 5, Section 1.5
Policy Source	IUA Authorization Client	Amendment mHealth of Annex 5, section 2.3.1
	ATNA Secure Application	Amendment mHealth of Annex 5, section 2.3.4
Policy Consumer	IUA Authorization Client	Amendment mHealth of Annex 5, section 2.3.1
	ATNA Secure Application	Amendment mHealth of Annex 5, section 2.3.4

Table 3: CH:PPQm required actors groupings

2.1.2 Referenced Standards

- HL7 FHIR standard Release 4: http://hl7.org/fhir/R4
- CH EPR Term Implementation Guide (R4) v2.0.7: http://fhir.ch/ig/ch-epr-term
- CH EPR mHealth Implementation Guide (R4) v1.0.0: http://fhir.ch/ig/ch-epr-mhealth
- CH EPR PPQm Implementation Guide (R4) v1.0.0: http://fhir.ch/ig/ch-epr-ppqm

2.1.3 Relation between CH:PPQm and CH:PPQ

This section is not normative.

Implementers may decide to implement CH:PPQm transactions on top of CH:PPQ ones, i.e. to create a FHIR layer over an existing XACML-based Policy Repository. The CH:PPQm specification supports this approach by defining transactions and data structures in a way which allows an efficient bridging between CH:PPQ and CH:PPQm, and by providing message transformation rules (see Chapter 4).

In terms of actor grouping, this would mean that the Policy Repository may be optionally grouped with CH:PPQ Policy Source and CH:PPQ Policy Consumer in order to communicate over PPQ-1 and PPQ-2 with itself.

Note that CH:PPQm is not intended to handle base policies and policy sets, i.e. the ones provided in the official Policy Stack and not related to any particular patients.

3 Volume 2 – Transactions

3.1 Mobile Privacy Policy Feed (PPQ-3)

3.1.1 Scope

This transaction is used by the Policy Source to add, update, or delete single privacy policies. Correspondingly, the following HTTP methods SHALL be supported: POST, PUT, and DELETE.

3.1.2 HTTP Method POST

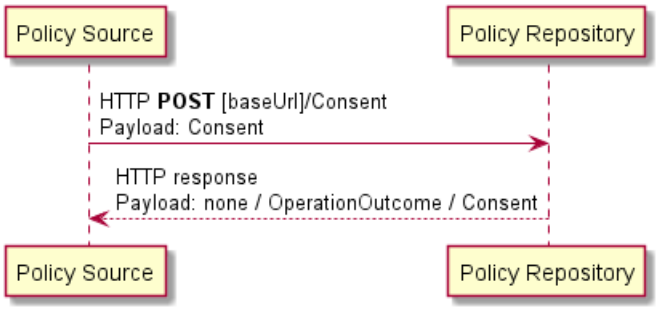

Figure 2: PPQ-3: HTTP Method POST

3.1.2.1 Trigger Event

The Policy Source uses HTTP method POST to submit a single new privacy policy to the Policy Repository.

3.1.2.2 Request Message

The request body SHALL represent a single Consent resource compliant to the PpqmConsent profile specified in the CH:PPQm Implementation Guide.

The request SHALL be sent to `[baseUrl]/Consent`.[2]

3.1.2.3 Expected Actions

Upon receiving the HTTP POST request, the Policy Repository SHALL:

- Validate the Consent resource contained in the request body.
- Persist the policy set represented by this Consent.
- Create a PPQ-3 response according to the transaction outcome.

3.1.2.4 Response Message

The PPQ-3 response SHALL be created according to the section 3.1.0.8 of the FHIR R4 specification.

[2] Here and in the whole document: Each FHIR URL may contain general parameters defined in the section 3.1.0.1.11 of the FHIR R4 specification; they are not shown for brevity.

3.1.3 HTTP Method PUT

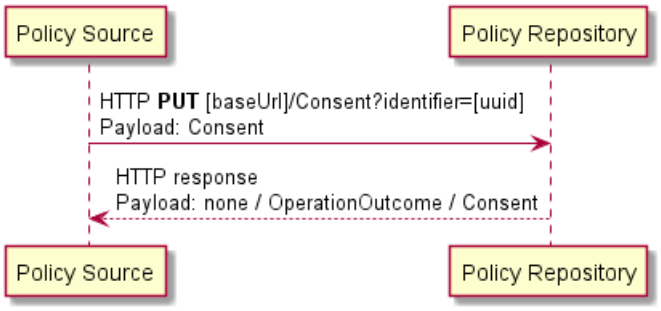

Figure 3: PPQ-3: HTTP Method PUT

3.1.3.1 Trigger Event

The Policy Source uses HTTP method PUT to submit a new or update an existing single privacy policy.

3.1.3.2 Request Message

The request body SHALL represent a single Consent resource compliant to the PpqmConsent profile specified in the CH:PPQm Implementation Guide.

The request SHALL be sent to `[baseUrl]/Consent?identifier=[uuid]`.

3.1.3.3 Expected Actions

The Policy Repository SHALL implement the Conditional Update pattern described in section 3.1.0.4.3 of the FHIR R4 specification.

Upon receiving the HTTP PUT request, the Policy Repository SHALL:

- Validate the Consent resource contained in the request body. In particular, it SHALL be validated that the policy set ID is the same as in the HTTP URL.
- Persist the policy set represented by this Consent.
- Create a PPQ-3 response according to the transaction outcome.

3.1.3.4 Response Message

The PPQ-3 response SHALL be created according to the section 3.1.0.4 of the FHIR R4 specification.

3.1.4 HTTP Method DELETE

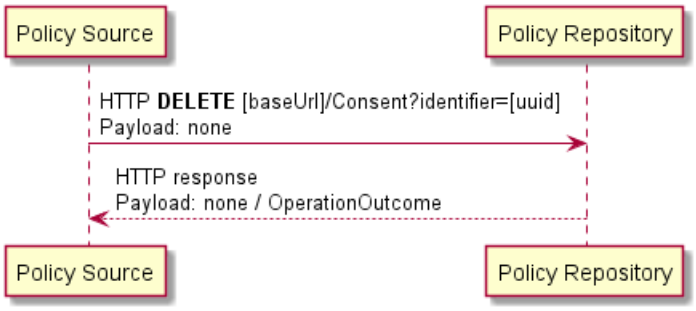

Figure 4: PPQ-3: HTTP Method DELETE

3.1.4.1 Trigger Event

The Policy Source uses HTTP method DELETE to delete a single existing privacy policy from the Policy Repository.

3.1.4.2 Request Message

The request body SHALL be empty.

The request SHALL be sent to `[baseUrl]/Consent?identifier=[uuid]`.

3.1.4.3 Expected Actions

The Policy Repository SHALL implement the Conditional Delete pattern described in section 3.1.0.7.1 of the FHIR R4 specification.

Upon receiving the HTTP PUT request, the Policy Repository SHALL:

- Delete the policy set referenced in the request.
- Create a PPQ-3 response according to the transaction outcome.

3.1.4.4 Response Message

The PPQ-3 response SHALL be created according to the section 3.1.0.7 of the FHIR R4 specification.

3.1.5 Security Considerations

TLS SHALL be used. For user authentication and authorization, the IUA profile with extended access token SHALL be used as described in the Amendment mHealth of Annex 5, Section 3.2. Consequently, the Mobile Privacy Policy Feed [PPQ-3] transaction SHALL be combined with the Incorporate Access Token [ITI-72] transaction of the IUA profile.

The involved actors SHALL record audit events. The Policy Source SHALL use the ATNA FHIR Feed option thereby, the Policy Repository SHALL use either the ATNA FHIR Feed option or the ATNA TLS Syslog option.

The audit records correspond to the ones of PPQ-1, with the following adaptations:

- EventTypeCode SHALL be set to EV(“PPQ-3”, “e-health-suisse”, “Mobile Privacy Policy Feed”).
- The Destination User ID SHALL be the FHIR endpoint URI of the Policy Repository.

3.2 Mobile Privacy Policy Bundle Feed (PPQ-4)

3.2.1 Scope

This transaction is used by the Policy Source to add, update, or delete a set of privacy policies. The only HTTP method which SHALL be supported is POST.

3.2.2 HTTP Method POST

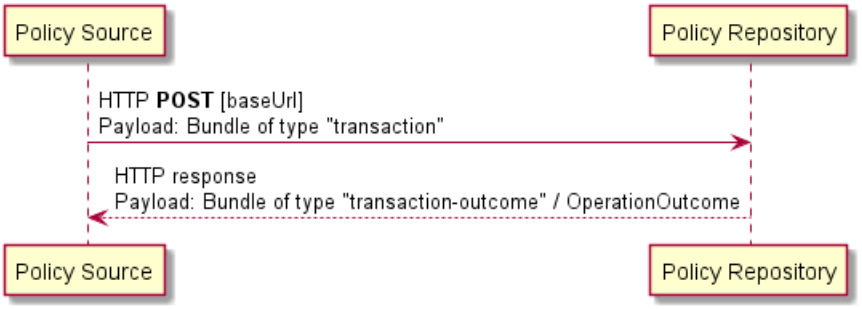

Figure 5: PPQ-4: HTTP Method POST

3.2.2.1 Trigger Event

The Policy Source uses HTTP method POST to perform an operation on a set of privacy policies in the Policy Repository, as an ACID transaction.

3.2.2.2 Request Message

The request body SHALL represent a single Bundle resource compliant to the PpqmRequestBundle profile specified in the CH:PPQm Implementation Guide.

The request SHALL be sent to `[baseUrl]`.

3.2.2.3 Expected Actions

Upon receiving the HTTP POST request, the Policy Repository SHALL:

- Validate the Bundle resource contained in the request body.
- On each request entry, perform the operation specified the attribute `entry.request.method` on the embedded or referenced PpqmConsent resource:
  - “POST” — add policy set.
  - “PUT” — update policy set if it is already present, otherwise add it.
  - “DELETE” — delete policy set.
- Create a PPQ-4 response according to the transaction outcome.

3.2.2.4 Response Message

The PPQ-4 response SHALL be created according to the section 3.1.0.11 of the FHIR R4 specification.

3.2.3 Security Considerations

TLS SHALL be used. For user authentication and authorization, the IUA profile with extended access token SHALL be used as described in the Amendment mHealth of Annex 5, Section 3.2. Consequently, the Mobile Privacy Policy Bundle Feed [PPQ-4] transaction SHALL be combined with Incorporate Access Token [ITI-72] transaction of the IUA profile.

The involved actors SHALL record audit events. The Policy Source SHALL use the ATNA FHIR Feed option thereby, the Policy Repository SHALL use either the ATNA FHIR Feed option or the ATNA TLS Syslog option.

The audit records correspond to the ones of PPQ-1, with the following adaptations:

- EventTypeCode SHALL be set to EV("PPQ-4", "e-health-suisse", "Mobile Privacy Policy Bundle Feed").
- The Destination User ID SHALL be the FHIR endpoint URI of the Policy Repository.

3.3 Mobile Privacy Policy Retrieve (PPQ-5)

3.3.1 Scope

This transaction is used by the Policy Consumer to retrieve policy sets. The only HTTP method which SHALL be supported is GET.

3.3.2 HTTP Method GET

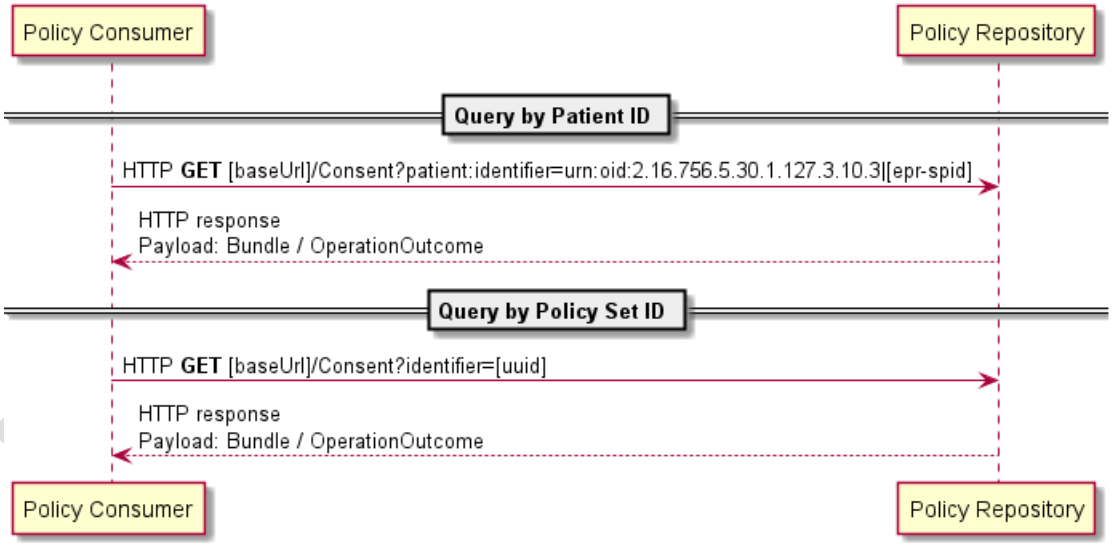

Figure 6: PPQ-5: HTTP Method GET

3.3.2.1 Trigger Events

The Policy Consumer sends this message to retrieve existing policy sets from the Policy Repository.

3.3.2.2 Request Message

The request body SHALL be empty.

The request SHALL be sent:

- For querying by patient ID — to `[baseUrl]/Consent?patient:identifier=urn:oid:2.16.756.5.30.1.127.3.10.3|[epr-spid]`.
- For querying by policy set ID — to `[baseUrl]/Consent?identifier=[uuid]`.

3.3.2.3 Expected Actions

Upon receiving the HTTP GET request, the Policy Repository SHALL create a PPQ-5 response according to the transaction outcome.

3.3.2.4 Response Message

The PPQ-5 response SHALL be created according to the section 3.1.0.9 of the FHIR R4 specification. If the response body is a Bundle, then it SHALL comply to the PpqmResponseBundle profile specified in the CH:PPQm Implementation Guide.

3.3.3 Security Considerations

TLS SHALL be used. For user authentication and authorization, the IUA profile with extended access token SHALL be used as described in the Amendment mHealth of Annex 5, Section 3.2. Consequently, the Mobile Privacy Policy Retrieve [PPQ-5] transaction SHALL be combined with the Incorporate Access Token [ITI-72] transaction of the IUA profile.

The involved actors SHALL record audit events. The Policy Consumer SHALL use the ATNA FHIR Feed option thereby, the Policy Repository SHALL use either the ATNA FHIR Feed option or the ATNA TLS Syslog option.

The audit records correspond to the ones of PPQ-2, with the following adaptations:

- EventTypeCode SHALL be set to EV(“PPQ-5”, “e-health-suisse”, “Mobile Privacy Policy Retrieve”).
- The Destination User ID SHALL be the FHIR endpoint URI of the Policy Repository.

4 Volume 3 – Content Profiles

Definitions of resource profiles, coding systems, and value sets for the FHIR resources used in the CH:PPQm profile are provided in the CH:PPQm implementation guide.

The rest of this chapter defines a mapping between CH:PPQm and CH:PPQ data structures — section 4.1 draws parallels between FHIR Consent resources and XACML 2.0 policy sets, section 4.2 focusses on the transformation of messages used in CH:PPQm and CH:PPQ transactions.

4.1 Mapping between PpqmConsent Resources and CH:PPQ Policy Sets

4.1.1 Transformation of PpqmConsent Resources into CH:PPQ Policy Sets

Each PpqmConsent resource contains an element `identifier` with `type.coding.code` equal to "templateId". In this element, the attribute `value` contains the ID of the official Policy Stack template. For example:

```
{
  "type" : {
    "coding" : [ {
      "system" :
              "http://fhir.ch/ig/ch-epr-ppqm/CodeSystem/PpqmConsentIdentifierType",
      "code" : "templateId"
    } ]
  },
  "value" : "201"
}
```

The PPQ-conformant XACML 2.0 Policy Set SHALL be created according to this template. Thereby, the value placeholders SHALL be filled according to the table below:

Placeholder	PpqmConsent attribute
Policy Set ID	`identifier.value` where `identifier.type.coding.code` equals to "policySetId"
EPR-SPID	`patient.identifier.value`
GLN	`provision.actor.reference.identifier.value`
Group OID	
Representative ID	
PolicySetIDReference	`policyRule.coding.code`
Start date	`provision.period.start`
End date	`provision.period.end`

Table 4: Mapping of PpqmConsent attributes onto CH:PPQ policy set template placeholders

4.1.2 Transformation of CH:PPQ Policy Sets into PpqmConsent Resources

Each patient-related CH:PPQ policy set is generated from a template provided in the official Policy Stack, but does not hold a direct reference to this template. Therefore, the first step is to determine the template ID. For that, the following heuristics SHALL be used:

- The policy set contains an element /PolicySet/Target/Subjects with *three* sub-elements Subject → template **203.**
- The policy set contains an element /PolicySet/Target/Subjects/Subject/SubjectMatch/AttributeValue/CodedValue with @codeSystem equal to “2.16.756.5.30.1.127.3.10.6” and:
  - @code equal to “PAT” → template **201**.
  - @code equal to “REP” → template **303**.
- The policy set contains an element /PolicySet/Target/Subjects/Subject/SubjectMatch/AttributeValue/CodedValue with @codeSystem equal to “2.16.756.5.30.1.127.3.10.5” and @code equal to “EMER” → template **202**.
- The policy set contains an element /PolicySet/Target/Subjects/Subject/SubjectMatch/SubjectAttributeDesignator with @AttributeId equal to “urn:oasis:names:tc:xspa:1.0:subject:organization-id” → template **302**.
- Otherwise → template **301.**

The PpqmConsent resource attributes SHALL be populated according to the rules defined for each template in the table below:

Ppqm-Consent attribute	**Template 201** (full access for the patient)	**Template 202** (confidentiality level for emergency access)	**Template 203** (minimal confidentiality level for upload)	**Template 301** (individual access permissions)	**Template 302** (group access permissions)	**Template 303** (full access for the patient’s representative)
identifier.value (1)	value of /PolicySet/@PolicySetId					
identifier.type (1)	fixed value: code “http://fhir.ch/ig/ch-epr-ppqm/CodeSystem/PpqmConsentIdentifierType|policySetId”					
identifier.value (2)	fixed value: “201”	fixed value: “202”	fixed value: “203”	fixed value: “301”	fixed value: “302”	fixed value: “303”
identifier.type (2)	fixed value: code “http://fhir.ch/ig/ch-epr-ppqm/CodeSystem/PpqmConsentIdentifierType|templateId”					
status	fixed value: code “active”					
scope	fixed value: code “http://terminology.hl7.org/CodeSystem/consentscope|patient-privacy”					
category	fixed value: code “http://terminology.hl7.org/CodeSystem/v3-ActCode|INFA”					
patient.identifier.system	fixed value: “urn:oid:2.16.756.5.30.1.127.3.10.3” (OID of EPR-SPID in URN format)					
patient.identifier.value	EPR-SPID of the patient					
policyRule.coding.code	value of /PolicySet/PolicySetIdReference					

Ppqm-Consent attribute	**Template 201** (full access for the patient)	**Template 202** (confidentiality level for emergency access)	**Template 203** (minimal confidentiality level for upload)	**Template 301** (individual access permissions)	**Template 302** (group access permissions)	**Template 303** (full access for the patient's representative)
provision.period.start	*not populated*	*not populated*	*not populated*	allowed only if the end date is provided	optional	*not populated*
provision.period.end	*not populated*	*not populated*	*not populated*	optional	required	*not populated*
provision.actor.role	fixed code: "EprRole|PAT"	fixed code: "EprRole|HCP"	fixed code: "EprRole|HCP"	fixed code: "EprRole|HCP"	fixed code: "EprRole|HCP"	fixed code: "EprRole|REP"
provision.actor.reference.identifier.type.coding.code	fixed value: "urn:e-health-suisse:2015:epr-spid"	*not populated*	*not populated*	fixed value: "urn:gs1:gln"	fixed value: "urn:oasis:names:tc:xspa:1.0:subject:organization-id"	fixed value: "urn:e-health-suisse:representative-id"
provision.actor.reference.identifier.value	EPR-SPID of the patient	*not populated*	*not populated*	GLN of the healthcare professional	OID of the HCP group, in URN format	ID of the patient's representative
provision.actor.reference.display	*not populated*	fixed value: "all"	fixed value: "all"	*not populated*	*not populated*	*not populated*
provision.purpose	*not populated*	fixed value: code "EprPurposeOfUse|EMER"	fixed value: set of codes "EprPurposeOfUse|NORM", "EprPurposeOfUse|AUTO" , "EprPurposeOfUse|DICOM_AUTO"	fixed value: code "EprPurposeOfUse|NORM"	fixed value: code "EprPurposeOfUse|NORM"	*not populated*

Table 5: Mapping of CH:PPQ policy set elements onto PpqmConsent attributes

4.2 Mapping between CH:PPQm and CH:PPQ Messages

4.2.1 Transformation of PPQ-3 Requests into PPQ-1 Requests

The PPQ-1 request type depends on the HTTP method used to submit the PPQ-3 request:

- POST — AddPolicyRequest.
- PUT — UpdatePolicyRequest if the policy already exists in the Policy Repository, otherwise AddPolicyRequest.
- DELETE — DeletePolicyRequest.

The policy set in AddPolicyRequest and UpdatePolicyRequest SHALL be constructed as described in section 4.1.1 from the PpqmConsent resource contained in the PPQ-3 request body.

The policy set ID in the DeletePolicyRequest SHALL be taken from the PPQ-3 request URL.

4.2.2 Transformation of PPQ-4 Requests into PPQ-1 Requests

PPQ-4 request body is a PpqmRequestBundle resource, where all attributes `entry.request.method` have the same value — “POST”, “PUT”, or “DELETE”.

The PPQ-1 request type depends on this value as follows:

- “POST” — AddPolicyRequest.
- “PUT” — UpdatePolicyRequest if the policy sets contained in the PPQ-4 request bundle already exist in the Policy Repository, otherwise AddPolicyRequest. If some policy sets do already exist in the Policy Repository while others do not, then the transformation of the PPQ-4 request into an PPQ-1 request is not possible, and the PPQ-4 request SHALL be rejected (creating two PPQ-1 requests would break the transactional semantics of PPQ-4).
- “DELETE” — DeletePolicyRequest.

Policy sets in AddPolicyRequest and UpdatePolicyRequest SHALL be constructed as described in section 4.1.1 from the PpqmConsent resources embedded in the PPQ-4 request bundle.

Policy set IDs in the DeletePolicyRequest SHALL be taken from the all attributes `entry.request.url` of the PPQ-4 request bundle.

4.2.3 Transformation of PPQ-1 Responses into PPQ-3/PPQ-4 Responses

Three variants are possible:

1. PPQ-1 response with the status “urn:e-health-suisse:2015:response-status:success”.
2. PPQ-1 response with the status “urn:e-health-suisse:2015:response-status:failure”.
3. PPQ-1 call ends with a SOAP Fault.

If the PPQ-1 response has the status “urn:e-health-suisse:2015:response-status:success”, then the PPQ-3/PPQ-4 response SHALL be created according to the sections 3.1.0.4 (for UpdatePolicy), 3.1.0.7 (for DeletePolicy), or 3.1.0.8 (for AddPolicy) of the FHIR R4 specification. If the client’s HTTP return preference is OperationOutcome, then this resource SHALL be created as defined in Table 6.

If the PPQ-1 response has the status “urn:e-health-suisse:2015:response-status:failure”, then the PPQ-3/PPQ-4 response SHALL be an OperationOutcome resource created as defined in Table 6, and the HTTP status code SHALL be set to 400.

The following mapping SHALL be used when transforming PPQ-1 responses into OperationOutcome resources:

OperationOutcome attribute	**Attribute “status” of the PPQ-1 response: urn:e-health-suisse:2015:response-status:< … >**	
	success	**failure**
severity	fixed value “`information`”	fixed value “`error`”
code	fixed value “`informational`”	fixed value “`processing`”

Table 6: Mapping of PPQ-1 response elements onto OperationOutcome attributes

If the PPQ-1 call ended with a SOAP Fault, then the PPQ-3/PPQ-4 response SHALL be an OperationOutcome resource created as defined in section 4.2.6, and the HTTP status code SHALL be set to the value defined in the same section.

4.2.4 Transformation of PPQ-5 Requests into PPQ-2 Requests

If the PPQ-5 request URL contains the parameter `patient:identifier`, then the PPQ-2 request SHALL address all policies related to the patient referenced there (retrieve policies by EPR-SPID).

If the PPQ-5 request URL contains the parameter `identifier`, then the PPQ-2 request SHALL address the policy set referenced there (retrieve policies by direct references).

4.2.5 Transformation of PPQ-2 Responses into PPQ-5 Responses

Three variants are possible:

1. PPQ-2 response with the status "`urn:oasis:names:tc:xacml:1.0:status:ok`".
2. PPQ-2 response with another status.
3. PPQ-2 call ends with a SOAP Fault.

If the PPQ-2 response has the status "`urn:oasis:names:tc:xacml:1.0:status:ok`", then the PPQ-5 response SHALL be a Bundle resource compliant to the PpqmRequestBundle profile specified in the CH:PPQm FHIR Implementation Guide. For each policy set contained in the PPQ-2 response, the following steps SHALL be performed:

- Transform the policy set into a Consent resource compliant to the PpqmConsent profile specified in the CH:PPQm FHIR Implementation Guide, as described in section 4.1.2.
- Add this Consent to the PPQ-5 response Bundle.

If the PPQ-2 response hat the status other than "urn:e-health-suisse: 2015:response-status:failure", then the PPQ-5 response SHALL be an OperationOutcome resource created as defined in Table 7, and the HTTP status code SHALL be according to Table 8.

The following mapping SHALL be used for transformation of negative PPQ-2 responses into OperationOutcome resources:

OperationOutcome attribute	Value
severity	fixed value "`error`"
code	Mapping of //samlp:StatusCode/@Value to a FHIR issue type code according to Table 8

Table 7: Mapping of PPQ-2 response elements onto OperationOutcome attributes

//samlp:StatusCode/@Value	FHIR Issue type code	HTTP status code
`urn:oasis:names:tc:SAML:2.0:status:Requester`	`invalid`	400
`urn:oasis:names:tc:SAML:2.0:status:Responder`	`invalid`	500
`urn:oasis:names:tc:SAML:2.0:status:VersionMismatch`	`structure`	500

Table 8: Mapping of SAML error codes onto FHIR issue type codes and HTTP status codes

If the PPQ-2 call ended with a SOAP Fault, then the PPQ-5 response SHALL be an OperationOutcome resource created as defined in section 4.2.6, and the HTTP status code SHALL be set to the value defined in the same section.

4.2.6 Transformation of SOAP Faults into OperationOutcome Resources

The following mapping SHALL be used when transforming SOAP Faults into OperationOutcome resources:

OperationOutcome attribute	Value
severity	fixed value “`error`”
code	Mapping of /soap:Fault/soap:Code/soap:Value to a FHIR issue type code according to Table 10
diagnostics	The whole soap:Fault element, Base64-encoded

Table 9: Mapping of SOAP Fault elements onto OperationOutcome attributes

SOAP error code	FHIR issue type code	HTTP status code
`VersionMismatch`	`structure`	500
`MustUnderstand`	`not-supported`	500
`DataEncodingUnknown`	`structure`	500
`Sender`	`invalid`	400
`Receiver`	`transient`	503

Table 10: Mapping of SOAP error codes onto FHIR issue type codes and HTTP status codes

5 Figures

6 Tables

7 Listings