Technical Agreement

Exchanging FHIR Data using a generic notified pull mechanism

Versie: 0.1

Disclaimer: Concept

Table of contents

1 Introduction

This Technical Agreement (TA) describes and specifies technical and quality responsibilities to which parties agree when connecting to exchange transactions to facilitate the Sender intent-based receiver pull, also known as “Notified Pull”.

1.1 Background

In an information exchange use-case (medical) patient information can be considered to have a “medical home”. This medical home is often represented by a clinical information system. This document does not define which systems within the medical home are responsible for the creation, storage, or maintenance of the BgZ. Neither will this document address the use-cases that drive the need to exchange the BgZ as there are many programs in the Netherlands that already do so. Instead, this document will focus on the roles and responsibilities a system or systems may have to get the BgZ from A to B using FHIR.

1.2 Definition of terms

Term	Definition
BgZ	Basisgegevensset Zorg; a Dutch set of patient information comparable to the International Patient Summary.
FHIR	Fast Healthcare Interoperability Resources; a next generation standards framework created by HL7.
Sender intent-based receiver pull	A more formal designation of the notified pull. Sometimes referred to as SIRP.

1.3 Goal, scope, and assumptions

The goal of this document is to introduce a neutral, objective FHIR-design for the Sender intent-based receiver pull. In creation existing specifications have been part of the balance to come to a generic solution. The following principles are followed:

1. The design must use international standards and is in line with the BgZ information standard and the future NEN7540 (NEN standard BgZ)
2. The design should be as generic and sustainable as possible (with a lifecycle of at least 3-5 years)
3. The design should be reusable for multiple uses.
4. The design should reuse already developed designs (e.g., MedMij, NUTS, Twiin, LSP)
5. The design must comply or explain, if anything deviates based on earlier principles, this can only be done if the reason is explained. This could be a deviation of use of standards, principles or if parts of the design are not reusable.
6. The design should be fitting for at least 80%, the rest will be produced during trial of the specification.

7. The design should not contain more specification than is strictly necessary within the goal and scope of this design.

1.4 Benefits of Notified Pull / Sender intent-based receiver pull

1. The receiving EHR system only receives on its own terms, by controlling how and when the data is pulled. This allows for data minimalization by only asking what you want to receive, when you want to receive it.
2. The notified pull mechanism allows for a deeper layer of security. When a receiver wants to retrieve the medical data, the receiver needs to identify herself which is in line with NENE7513 and the AVG, which state that a log should be kept for views. In comparison, using a PUSH the data will already be in the receiving system, without a check or log of who is the user of the information.
3. In relation to Pull, the notified pull mechanism allows for better timing and security. With Pull the receiving system will have to continuously pull to discover new information. Using a notification to initiate a pull reduces network communications and better timing by communicating when the message is ready to be received.
4. Security wise, notified pull benefits from a more linear authorization matrix than would be necessary for Pull. For pull mechanisms the requesting party needs to be identified within a pre-existing authorization matrix, whereas with notified pull the requesting party can prove its authentication for the request by the received consent token, which was received from the initiating party.

Compared to a regular Push, the Notified-Pull has the following additional security objectives:

- Availability of data for a Requester may (basically) not be lower than with a regular Push. The source EHR system obtains the necessary (assumed) consent from the patient on the spot. This permission can be used during the request.
- Integrity: The requester can potentially have access to more up-to-date data, because the request can take place at the moment the data is actually needed (used).
- Confidentiality: Data is only requested when it will be used and is also requested by a healthcare provider/employee who is entitled to do so due to his/her position and work context. In addition, it is sometimes possible (as Source) to undo an incorrectly initiated notified pull before actual medical data is exchanged.

1.5 Decisions

- Generic notified pull description
- Generic description without states, specifics (like BgZ attachment) may contain the use of states.
- Attachment specific for BgZ, because it was part of what triggered the Technical Agreement.
- External attachment for the differences with current eOverdracht implementation, did we make a choice which is deemed unacceptable and can we change this.
- A few notification resources have been discussed, since we chose to use FHIR STU3, the choice was made to focus on the Task resource.
- When Notified Pull is being used as part of a workflow, the notification must contain a reference to the original workflow request.
- Patient BSN is needed for processing of the notification and will be communicated through the push (in the assertion and notification task) OR pull (in the referenced workflow task in the notification task.

1.6 Relation to other documents

This document is written with the following documents as reference:

- Informatiestandaard BgZ MSP
- Technical Agreement Exchanging BgZ

1.7 Format of technical agreement

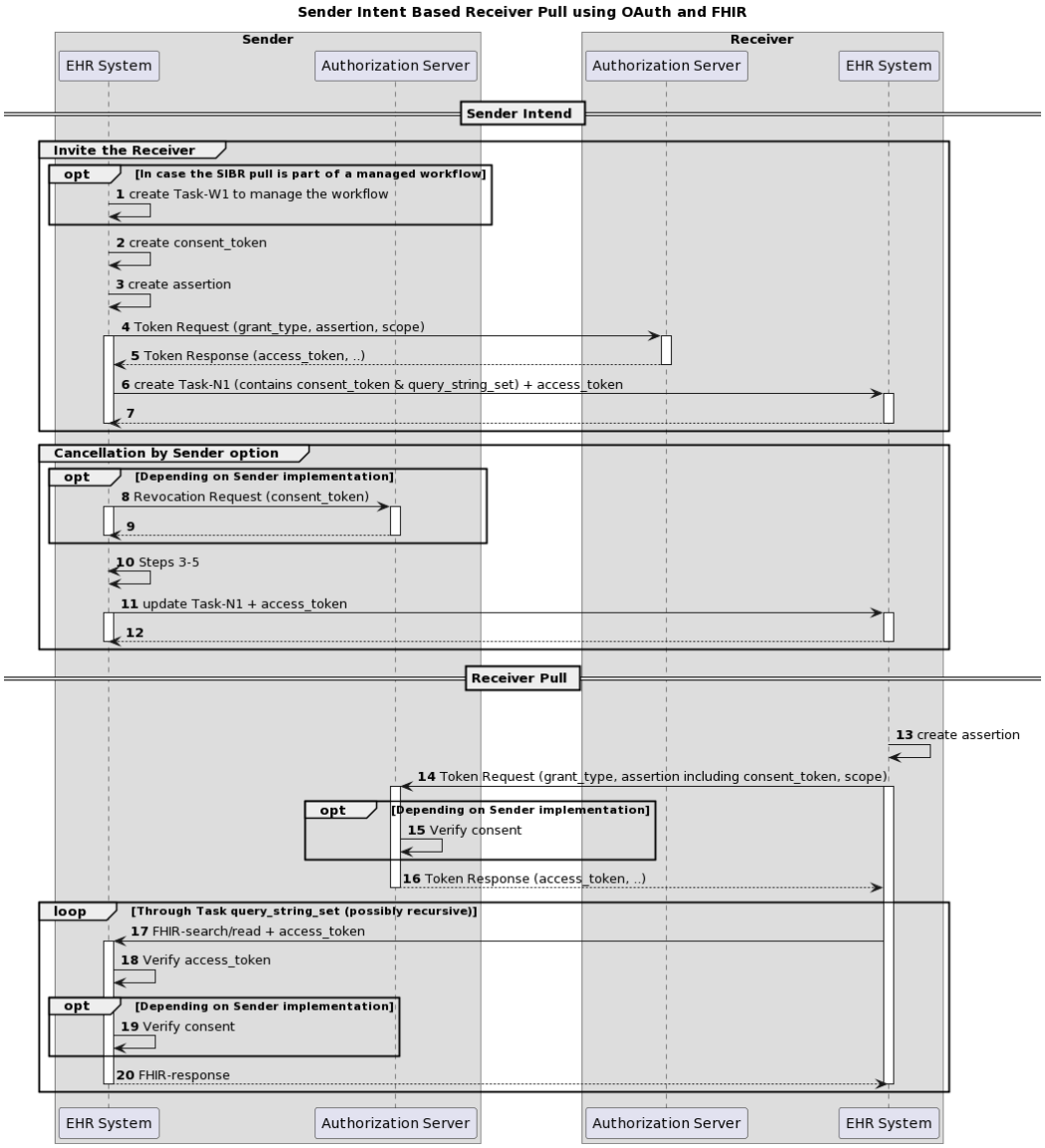
The sequence diagram describes the flow of the interaction between sending and receiving EHR system. In access control both authentication and authorization are described. Addressing describes where to connect to. Notification and Pull describe the two parts of the process of notifying about the availability of data and pulling the resources

2 Technical agreements

2.1 Sequence diagrams

2.1.1 Sender Intent Based Receiver Pull (notified pull)

The sequence diagram below depicts the flow for the Sender Intent Based Receiver Pull (SIBR Pull) using OAuth 2.0 and HL7-FHIR, also known as "notified pull".



The flow contains the following sections:

- Invite the Receiver;
- Cancellation by Sender (option), this block is only to be used when the Sender needs to withdraw the pull invitation, e.g., when the Sender invited a wrong Receiver;
- Receiver performs pull interaction(s).

Each section consists of several steps. The steps correspond to the numbers in the sequence diagram.

Section	Step	Description
Invite the Receiver	1	Optional reference to a request-Type resource that produced this event. If a workflow has been initiated, this should be referenced.
	1	If the SIBR pull is part of a managed workflow involving both the Sender and the Receiver, and this workflow specifies the creation of a Task (W1) at the Sender, then the flow starts with a creation of this Task on the Sender EHR System.
	2	The Sender creates a consent_token, which is used later to communicate a presumed consent for the exchange of patient information. The Receiver must treat the consent_token as opaque. The Receiver should not depend on any information contained in the consent_token.
	3	The Sender creates an assertion, which can be used as an authorization grant when requesting an access_token in the next step.
	4, 5	The Sender requests an access_token which can be used in step 6. The Receiver processes the token request and returns a token response containing (among others) an access_token. The Sender must treat the access_token as opaque. The Sender should not depend on any information contained in the access_token.
	6, 7	By creating a Task (N1) on the Receiver EHR System, the Sender invites the Receiver to perform one or more pull interactions. The Receiver processes the invitation and sends a technical response to complete the create interaction.
Cancellation by Sender	8, 9	Depending on the implementation at the Sender side, the Sender EHR System might have to revoke the consent_token created in step 2, by sending a revocation request to the Sender Authorization Server. The Authorization Server processes the request and returns a response.
	10	The Sender repeats step 3-5.
	11, 12	The Sender informs the Receiver by updating the Task (N1) on the Receiver EHR System. The Receiver returns a technical response message.

Receiver performs pull interaction(s)	13	The Receiver creates an assertion, which can be used as an authorization grant when requesting an access_token in the next step.
	14, 15, 16	The Receiver requests an access_token which can be used to perform the intended pull interactions. The Sender Authorization Server processes the token request and returns a token response containing (among others) an access_token. Depending on the Sender implementation, the Sender can choose to verify the consent before issuing an access_token (preferred option). The Receiver must treat the access_token as opaque. The Receiver should not depend on any information contained in the access_token.
	17, 18, 19, 20	The Receiver initiates the intended interactions and processes the responses. The Sender verifies the access_token and, next to that, can choose to verify the consent at this point in the flow. In case a query_string involves a FHIR-read on Task, then any query_strings included in this second Task will be processed as well.

2.2 Access control

Both the sending and receiving EHR system expose endpoints that must be protected from unauthorized and malicious interactions. As the sequence diagram in section 2.1 illustrates, security measures must be applied to the following resource endpoints:

- Notification endpoint of receiving EHR system
- Resource endpoint of sending EHR system

2.2.1 Network level security: mTLS 1.3

On network level mutual TLS (mTLS) must be applied. As advised by the NCSC version 1.3 of the TLS standard must be applied. The implementation of mTLS serves the following purposes:

- Authentication of client and server on network level
- Encryption of communication between client and server

Both the client and server certificates must be PKIo-certificates. Accepted PKIo-certificates are:

- UZI server certificate issued by UZI-registry (CIBG)
- PKIoverheid Private Services CA – G1 certificate

Note that the requirements as specified in this paragraph apply to **notification, resource, and token** endpoints. On JWK Set URIs server-side TLS must be applied.

2.2.2 Resource server authorization: OAuth 2.0

On application level both the notification endpoint of the receiving EHR system and the FHIR endpoint of sending EHR system must be secured by OAuth 2.0. This implies that a client that wants to interact with a resource server (FHIR or notification endpoint) must obtain an access token at an authorization server before it can interact with that resource server. The client must present this access token as bearer token in each request to the resource server.

2.2.2.1 Client authentication

The resource server must be able to authenticate the client as a trusted client. The client is specified as the **system** that submits the access token request (not to be confused with the **organization** for which that system is acting). The client must authenticate itself by providing a client assertion by means of a signed JWT as specified in RFC 7523 section 2.2.

The assertion is a JWS Compact Serialized JWT that consists of a header, a payload, and a signature. The signature is created using a key pair belonging to the initiating organization or to a third party trusted by the initiating organization.

The header carries the claims listed below:

Claim	Description	Required
typ	Token type, must be “JWT”	Yes
alg	Cryptographic algorithm used to sign the assertion. See RFC 7515 section 4.1.1. must be one of PS256, PS384, PS512, ES256, ES384 or ES512.	Yes
kid	Identifier of the key pair used to sign this JWT. See RFC 7515 section 4.1.4.	Yes

The payload contains a set of claims listed below:

Claim	Description	Required
jti	Unique identifier of this assertion. See RFC 7519 section 4.1.7.	Yes
iss	Identifier of the system that issued the assertion. See RFC 7519 section 4.1.1 and RFC 7523 section 3.	Yes
iat	The time at which the JWT assertion was issued. See RFC 7519 section 4.1.6.	Conditional[1]
exp	The expiration time on or after which the assertion shall not be accepted for processing. See RFC 7519 section 4.1.4 and RFC 7523 section 3.	Yes
nbf	The time before which the token shall not be accepted for processing. See RFC 7519 section 4.1.5 and RFC 7523 section 3.	No
aud	Identifier of the authorization server token endpoint where this assertion is to be used. See RFC 7519 section 4.1.3 and RFC 7523 section 3.	Yes
sub	Identifier of the OAuth client that requests access. This claim must match the value of the client_id parameter in the access token request. Note that the client is specified as the system that submits the access token request.	Yes

[1] If there is an agreed age of an assertion.

The issuer of the client assertion may include additional claims in the assertion, but the issuer shall not require the authorization server to process these claims.

The issuer of the client assertion must publish the public key that was used to sign the assertion in a JWK Set in accordance with RFC 7517 at an URI that can be accessed by the authorization server. This enables the authorization server to obtain the public key to validate the client assertion. Exchange of the JWK Set URI and the corresponding identifier used in the iss claim is out of scope at this moment and must be agreed upon by the involved vendors.

Note that authorization server can authenticate the client on network level by the client certificate that the client must present during the mTLS handshake (see section Network level security). In theory, this could be used by the authorization server to authenticate the client on application level. However, this may cause problems since it introduces additional and potentially unwanted requirements on TLS termination and related matters. Therefore, a client must always provide a client assertion in the access token request.

2.2.2.2 Authorization grant

OAuth 2.0 requires the use of an authorization grant to request an access token. As specified in RFC 6749 section 1.3 “an authorization grant is a credential representing the resource owner's authorization (to access its protected resources) used by the client to obtain an access token.” OAuth 2.0 and specifies several different authorization grants. Additionally, there are several RFC’s that specify extension grants. Because this TA applies to situations where a resource client is acting on behalf of a user (healthcare professional) that works for an organization (healthcare supplier) the use of the JWT Bearer Assertion authorization grant as specified in RFC 7523 section 2.1 is the most suitable authorization grant. This means that the resource client must provide an assertion in each access token request to identify the acting user, organization, and consent to proof that it is authorized to access the requested data.

The assertion is a JWS Compact Serialized JWT that consists of a header, a payload, and a signature. The signature is created using a key pair belonging to the initiating organization or to a third party trusted by the initiating organization.

The header carries the claims listed below:

Claim	Description	Required
typ	Token type, must be “JWT”	Yes
alg	Cryptographic algorithm used to sign the assertion. See RFC 7515 section 4.1.1. must be one of PS256, PS384, PS512, ES256, ES384 or ES512.	Yes
kid	Identifier of the key pair used to sign this JWT. See RFC 7515 section 4.1.4.	Yes

The payload contains a set of claims that carry information required by NEN 7512 and NEN 7513.

Claim	Description	Required
jti	Unique identifier of this assertion. See RFC 7519 section 4.1.7.	Yes
iss	Identifier of the system that issued the assertion. See RFC 7519 section 4.1.1 and RFC 7523 section 3.	Yes
iat	The time at which the JWT assertion was issued. See RFC 7519 section 4.1.6.	Conditional[2]
exp	The expiration time on or after which the assertion shall not be accepted for processing. See RFC 7519 section 4.1.4 and RFC 7523 section 3.	Yes
nbf	The time before which the token shall not be accepted for processing. See RFC 7519 section 4.1.5 and RFC 7523 section 3.	No
aud	Identifier of the authorization server token endpoint where this assertion is to be used. See RFC 7519 section 4.1.3 and RFC 7523 section 3.	Yes
sub	Identifier of the **organization** (healthcare supplier) that requests access.	Yes
user_id	Identifier of the **responsible user** (healthcare professional) who requests access.	Conditional[3]
user_role	Code of the role of the responsible user (healthcare professional) who requests access.	Conditional[4]
authorizer	Identifier of the **organization** (healthcare supplier) that grants access.	Yes
consent_token	See Consent	No

[2] The issued at claim is only required if there is an agreed age of an assertion.
[3] User identification (user_id and user_role claims) is only required in the assertion when access to patient data is requested. This implies that these claims are not required in assertions used in access token requests for notification endpoints.
[4] See previous.

patient	Identifier of the patient for whom data is exchanged. must be an OID encoded BSN (I.e., BSN with the “urn:oid:2.16.840.1.113883.2.4.6.3.” prefix and without a leading zero)	Conditional[5]

The issuer of the assertion may include additional claims in the assertion, but the issuer shall not require the authorization server to process these claims.

The issuer of the assertion must publish the public key that was used to sign the assertion in a JWK Set in accordance with RFC 7517 at an URI that can be accessed by the authorization server. This enables the authorization server to obtain the public key to validate the assertion. Exchange of the JWK Set URI and the corresponding identifier used in the iss claim is out of scope at this moment and must be agreed upon by the involved vendors.

2.2.2.3 Authorization scope

The scope defines the requested access to the FHIR Server as specified in RFC 6749 section 3.3. If a scope is provided in the access token request or access token response, it must be expressed in a string of space delimited scopes as defined in SMART on FHIR v2. The following additional requirements apply to the scope values:

- When requesting an access token for a notification endpoint at the receiving EHR system, the scope value must be either “system/Task.c?code=http://xxx.nl/fhir/CodeSystem/xxx|pull_notification” (create) or “system/Task.u?code=http://xxx.nl/fhir/CodeSystem/xxx|pull_notification” (update)
- When requesting an access token for a FHIR endpoint at the sending EHR system, the query parameters in the scopes must match (a subset of) the queries in the FHIR search requests listed in Task.input of the notification Task (see section Task resource).

The client must provide the requested scope in the access token request, except for cases where a content token is provided in the access token request as part of the assertion that serves as an authorization grant.

The authorization server must provide the granted access scope in the access token response in accordance with RFC 6749 section 5.1 and the requirements mentioned above. The issued access token must grant access to the granted scope that the authorization server specifies in the access token response. The granted scope must be equal to or less than the scope that can be deduced from the consent token.

[5] Patient identification is only required when the sending EHR system requests access to the notification endpoint of the receiving EHR system and the sending EHR system does not provide a workflow Task that refers to a Patient resource containing the BSN of the patient. This way, the receiving EHR system is always able to identify a patient by BSN based on a notification. The receiving EHR system must support receiving the BSN through the patient claim.

2.2.2.4 Access token request

Based on the paragraphs above each access token request contains the parameters listed below:

Parameter	Value	Required
grant_type	"urn:ietf:params:oauth:grant-type:jwt-bearer"	Yes
assertion	JWT bearer assertion as specified in paragraph 2.2.2.2.	
client_assertion_type	"urn:ietf:params:oauth:client-assertion-type:jwt-bearer"	Yes
client_assertion	JWT client assertion as specified in paragraph 2.2.2.1.	Yes
client_id	ID of the resource client. This ID is issued by the authorization server. If present, the value of the "client_id" parameter must identify the same client as is identified by the client assertion.	No
scope	Space separated list of requested scopes, see paragraph 2.2.2.3.	Conditional

Note that the access token request effectively contains two JWT assertions:

1. A client assertion that is used to authenticate the client. This assertion identifies and authenticates the system that is requesting access.
2. A JWT bearer assertion that is used as an authorization grant. This assertion identifies both the organization and user that are requesting access.

Separating client authentication from client authorization in two separate assertions enables the client to select different assertion issuers for the two assertions. The targeted authorization server must register both issuers as trusted assertion issuers for a specific client.

2.2.2.5 Access token requirements

The access token will be processed only by the party that issued the access token. Therefore, the form and contents of the token are determined by the authorization server (audience), so the access token is opaque to the resource client. The resource client should not take any dependency on the format or contents of an access token.

2.2.3 Consent

Verification of (implicit) consent is the responsibility of the sending EHR system. For that purpose, the sending EHR system may submit a consent token to the receiving EHR system as part of the notification (see section Task resource). If the receiving EHR system received a consent token in the notification, it must include that consent token in the access token request at the sending EHR system (see section Authorization grant). This enables the authorization server of the sending EHR system to determine if the requested access can be granted based on the provided consent token.

Since a consent token is to be processed by the sending EHR system only, the form and contents of a consent token are determined by the sending EHR system. The receiving EHR system should not take any dependency on the format or contents of a consent token.

2.2.4 User authentication

Healthcare professionals are identified in their EHR system by logging in with their personal account. When a user of the receiving EHR system wants to request resources at the sending EHR system, the sending EHR system must be able to identify the user at the receiving EHR system as a legitimate healthcare professional who is working for the receiving organization before it can serve the requested data. Therefore, the receiving EHR system must implement the appropriate means to ensure the authenticity of the user. Multifactor authentication is preferred to ensure the identity. Whether this is done using an UZI-card or another safe login method is not specified at this time. NEN 7512 requirement is eIDAS high for exchange of patient/medical information.

2.2.5 Accountability / Audit logging

All logging must comply with NEN7513. No specific extra information based on Notified Pull is described.

2.2.6 Delegation

Delegation may be supported by including an *act* claim in the assertion, however it is out of scope of the first iteration of this technical agreement. Requirements and consequences for delegation support are not yet clear in the use-cases currently in view.

2.3 Addressing

Every connected health organization has at least three endpoints that should be known by another organization:

- Notification endpoint; the endpoint to which the notification can be pushed
- Authorization server endpoint; the endpoint where the access_token can be requested.
- Resource server endpoint; the endpoint which is used to request the actual resources.

Endpoints can be used for multiple organizations, identification of the sending organization will be managed in the notification. Identifiers that can be used for organizations are code system OID, DID or (Dutch) URA.

More specific delivery to an internal receiver/person(s) in an organization can be managed by FHIR ActivityDefinitions (healthcare products a receiver defines). Agreements about this topic will be specified in the specific use case for now.

Communication/publication of the endpoints and identifiers of each organization will be managed outside this Technical Agreement (government) between implementing partners, or so-called trusted gateways/nodes/trusted networks. So, the exact method of distribution of endpoint URLs is not specified in this version of the TA.

Options (informative):

- Using a trusted third party that acts as an issuer of endpoint information (e.g., "ZORG-AB")
- Using a distributed registry that is managed by the connected healthcare organizations and/or their service providers
- Implementing partners have made an agreement about their own communication method for endpoints and organizations

There are several methods to share endpoint URLs, via another endpoint URL of a connect healthcare organization (informative):

- Share Authorization server endpoint via the Resource Server's SMART configuration:
    - Via /.well-known/smart-configuration
    - https://build.fhir.org/ig/HL7/smart-app-launch/conformance.html
- Share Resource Server endpoint via the Authorization Server's well-known registry
    - https://www.rfc-editor.org/rfc/rfc8414.html#section-7.3

2.4 Notification

The invitation of the Receiver to get the information that is made available results in a notification.

2.4.1 Scope

The notification transaction passes a Task from a Sender to a Receiver.

2.4.2 Actors & Roles

Actor	Role
Sender	Sends a notification of the availability of data for the receiver
Receiver	Handles a notification and processes whatever needs to be processed in the receiving system.

2.4.3 Referenced Standards

FHIR STU3 - FHIR Release 3 (STU)

2.4.4 Messages

2.4.4.1 New Notification Task request message

This message uses the HTTP POST method on the target notification endpoint to convey the notification information as a FHIR resource.

2.4.4.1.1 Trigger Events

This method is invoked when the Sender needs to send a notification of available information to a Receiver.

2.4.4.1.2 Message Semantics

The Sender must initiate a FHIR notification using a “create” action by sending an HTTP POST request method composed of a FHIR Task resource.

The media type of the HTTP body must be either application/fhir+json or application/fhir+xml.

The Notification Task is sent using the information described in the Addressing section.

2.4.4.1.2.1 Task resource

For complete information on constructing a FHIR Task Resource, see https://hl7.org/fhir/stu3/task.html.

Two types of notification tasks are identified:

- Full notification
- Hybrid notification (getting a workflow request resource that is referenced is necessary to get complete overview of the shared resources)

The full notification contains everything for the receiver to know which information was made available. For instance, useable if the search url does not contain patient-specific parameters.

The hybrid notification is available to be able to keep patient-specific identifiers out of the notification. For instance, if a patient identifier like the BSN is contained in the identifier, this would be beneficial to not communicate in the notification, as the receiver is not known as a single legal person to the sender.

Attribute	Card.	Description
basedOn	0..*	Optional reference to a request-Type resource that produced this event. If a workflow has been initiated, this should be referenced.
identifier	**1**..*	Business identifier of the task
status	1..1	The state communicated by this event. Preferred value: • requested See also: https://hl7.org/fhir/stu3/valueset-request-status.html
intent	1..1	Indicates the "level" of actionability associated with the Task. Preferred value: • proposal See also: https://hl7.org/fhir/STU3/valueset-request-intent.html
definitionReference	0..1	Optional reference to an ActivityDefinition defining the activity that would be performed when retrieving the data.
code	0..1	A code briefly describing what the task involves: • code.coding.system="http://xxx.nl/fhir/CodeSystem/xxx" • code.coding.code="pull_notification"
for.identifier	0..1	The patient identifier in the form of BSN.
restriction.period	0..1	The period during which the data will be available for retrieval.
requester.agent.identifier	1..1	Identifier of the Device at which the data has been made available.
requester.onBehalfOf.identifier	**1**..1	Identifier of the Organization at which the data has been made available.
owner	**1**..1	Identifier of the receiving organization. When identifier is filled, it can be assumed it is a reference to a resource of the type Organization.
input • **type**	0..**1**	The consent_token to be used when retrieving the data. • type.coding.system="http://xxx.nl/fhir/CodeSystem/TaskParameterType".

• **valueString**		• type.coding.code="consent_token". • valueString

Input
 	
• **type**
• **valueString**
	0..1	The FHIR read and/or FHIR search interactions that can be performed to retrieve the task that contains more inputs than could be made available through this notification.
		
		• type.coding.system="http://xxx.nl/fhir/CodeSystem/TaskParameterType"
		• type.coding.code="query_task"
		• valueString format:
		  ○ Task/[id], or
		  ○ Task{?[parameters]}
		
		Where:
		
		• *type* denotes a FHIR resourcetype;
		• *id* represents a logical id of a FHIR resource instance;
		• *parameters* can be added to refine a FHIR-search.
		
		A specific implementation can override the type with LOINC and/or SNOMED CT codes, if deemed necessary.

input
 	
• **type**
• **valueString**
	1..*	The FHIR-read and/or FHIR-search interactions that can be performed to retrieve the data that was made available.
		
		• type.coding.system="http://xxx.nl/fhir/CodeSystem/TaskParameterType"
		• type.coding.code="query_resource"
		• valueString format:
		  ○ [type]/[id], or
		  ○ [type]{?[parameters]}
		
		Where:
		
		• *type* denotes a FHIR resourcetype;
		• *id* represents a logical id of a FHIR resource instance;
		• *parameters* can be added to refine a FHIR-search.
		
		A specific implementation can override the type with LOINC and/or SNOMED CT codes, if deemed necessary.

2.4.4.1.3 Expected Actions

The Receiver must accept both media types *application/fhir+json* and *application/fhir+xml*.

On receipt of the submission, the Receiver must validate the resource and respond with one of the HTTP codes defined in the response Message Semantics.

The notification should trigger an event in the Receiver system to process the expected pull.

2.4.4.2 New Notification Task response message

The Receiver returns a HTTP Status code appropriate to the processing outcome, conforming to the specification requirements as specified in http://hl7.org/fhir/stu3/http.html.

2.4.4.2.1 Trigger Events

This message must be sent when a success or error condition needs to be communicated. Success is only indicated once the notification is received and completely processed. Persistence of the resource is not necessary.

2.4.4.2.2 Message Semantics

To enable the Sender to know the outcome of processing the notification, the Receiver must return either an empty body or an OperationOutcome resource. This body must be accompanied with the correct HTTP status code, e.g.:

- 200 OK – Notification received and not persisted.
- 201 Created – Notification received and persisted. In this case http-headers Location must be filled and Etag should be filled.
- 400 Bad Request – Notification could not be parsed or failed basic FHIR validation rules.
- 404 Not Found – Resource type not supported, or wrong endpoint.
- 412 Precondition Failed – The processing of the Notification could not be finished, since the criteria were not selective enough.
- 422 Unprocessable Entity – The Notification resource violated applicable server business rules. This should be accompanied by an OperationOutcome resource providing additional detail.

2.4.4.2.3 Expected Actions

The Sender processes the result according to application defined rules.

2.4.4.3 Cancel Notification Task request message

This message uses the HTTP PUT method on the target notification endpoint to convey the notification cancellation as a FHIR resource.

2.4.4.3.1 Trigger Events

This method is invoked when the Sender needs to send a notification of revocation of the notification to the Receiver.

2.4.4.3.2 Message Semantics

The Sender must cancel the FHIR notification using a “conditional update” action by sending an HTTP PUT request method with the identifier parameter.

The media type of the HTTP body must be either application/fhir+json or application/fhir+xml.

The Notification Task is sent using the information described in the Addressing section.

2.4.4.3.2.1 Task resource

For complete information on constructing a FHIR Task Resource, see https://hl7.org/fhir/stu3/task.html.

Attribute	Card.	Description
basedOn	0..*	Optional reference to a request-Type resource that produced this event. If a workflow has been initiated, this should be referenced.
identifier	**1**..*	Business identifier of the task
status	1..1	The state communicated by this event. Preferred value: • requested See also: https://hl7.org/fhir/stu3/valueset-request-status.html
intent	1..1	Indicates the "level" of actionability associated with the Task. Preferred value: • proposal See also: https://hl7.org/fhir/STU3/valueset-request-intent.html
definitionReference	0..1	Optional reference to an ActivityDefinition defining the activity that would be performed when retrieving the data.
code	0..1	A code briefly describing what the task involves: • code.coding.system="http://xxx.nl/fhir/CodeSystem/xxx"

		• code.coding.code="pull_notification"
restriction.period	0..1	The period during which the data will be available for retrieval.
requester.agent.identifier	1..1	Identifier of the Device at which the data has been made available.
requester.onBehalfOf.identifier	**1**..1	Identifier of the Organization at which the data has been made available.
owner	**1**..1	Identifier of the receiving organization. When identifier is filled, it can be assumed it is a reference to a resource of the type Organization.

2.4.4.3.3 Expected Actions

The Receiver must accept both media types *application/fhir+json* and *application/fhir+xml*.

On receipt of the submission, the Receiver must validate the resource and respond with one of the HTTP codes defined in the response Message Semantics.

The notification should trigger an event in the Receiver system to process the expected pull.

2.4.4.4 Cancel Notification Task response message

The Receiver returns a HTTP Status code appropriate to the processing outcome, conforming to the specification requirements as specified in http://hl7.org/fhir/stu3/http.html.

2.4.4.4.1 Trigger Events

This message must be sent when a success or error condition needs to be communicated. Success is only indicated once the notification is received and completely processed. Persistence of the resource is not necessary.

2.4.4.4.2 Message Semantics

To enable the Sender to know the outcome of processing the notification, the Receiver must return either an empty body or an OperationOutcome resource. This body must be accompanied with the correct HTTP status code:

- 200 OK – Notification received and not persisted.
- 201 Created – Notification received and persisted. In this case http-headers Location must be filled and Etag should be filled.
- 400 Bad Request – Notification could not be parsed or failed basic FHIR validation rules.
- 404 Not Found – Resource type not supported, or wrong endpoint.
- 412 Precondition Failed – The processing of the Notification could not be finished, since the criteria were not selective enough.
- 422 Unprocessable Entity – The Notification resource violated applicable server business rules. This should be accompanied by an OperationOutcome resource providing additional detail.

2.4.4.4.3 Expected Actions

The Sender processes the result according to application defined rules.

2.4.4.5 Availability of BSN

For correct handling BSN should be available as soon as possible. The sending EHR system has two possibilities:

- The BSN is sent in both the access token as well as the notification Task resource (as described in this chapter).
- The BSN is made available through the workflow Task resource which is referenced in the basedOn of the notification Task resource. The workflow Task resource must have a for reference with the identifier filled with the BSN.

The receiving EHR system must support both.

2.5 Pull

Getting the resources, based on default FHIR requests, described as RESTful API[6]. The Task will provide the search URLs needed for the pull in Task input.

Patient identifiers shall not be included in the search URLs listen in the Task input. It is up to the EHR system to relate the FHIR requests to the patient.

[6] https://www.hl7.org/fhir/stu3/http.html (STU3) or https://www.hl7.org/fhir/http.html (currently R4)

3 For future reference

What should be done in the future, to improve the CIA triad of the interoperability using FHIR.

4.2.1 should be updated as soon as a universally available method is available.

4 Document management

4.1 Involved parties

This document is a co-creation of the companies listed below. The following people have been involved in creating this document.

Company	Contact person	Mail
Nexus	Dennis Willemsen	
Tenzinger	Jorrit Spee	
Twiin	Marc Sandberg	
VZVZ	Ron van Holland	
Zorgdomein	Stephan Opdenberg	
ZorgDomein	Ruben Pape	

4.2 Version control

Rev	Release Date	Author	Description of change
0.9	23-01-2023	All	Version for consultation

Appendix: BgZ implementation

The implementation for BgZ with Notified Pull is fully based on the Nictiz informatiestandaard “BgZ medisch specialistische zorg”, which itself is based on the MedMij BgZ. This appendix will provide a guideline of how to use the Notified Pull exchange pattern to transfer the BgZ.

As the sequence diagram Notified Pull points out, the sending system may choose to provide a Task resource that can be used to exchange status updates and other workflow related details related to the healthcare process that demands the data exchange. In the context of a BgZ-referral, the sending system may choose to provide a Task resource that is used to exchange details about status updates or other workflow updates related to the referral. This Task resource will be referred to as the “Workflow task”. Note that this is not the same as the status of the data exchange.

To enable the implementation of a hybrid notification, a minimal profile of the Task that represents the referral workflow is provided. After all, when a minimal notification is sent by the system of the referring party, the receiving system must be able to request the references to the provided data set at the sending system. The implementation of this workflow Task is only required when the sending system uses a hybrid notification. When the sending system uses a full notification, the implementation of the workflow Task is optional. If provided, it must be referenced by the Task resource in the notification in Task.basedOn.

The BgZ workflow Task profile is based on the NL Core Workflow task.

Name	Card.	Type	Comments
definition	0..1	Reference(ActivityDefinition)	Reference to ActivityDefinition resources that defines the requested activity or service
status	1..1	code	requested | received | accepted | rejected | cancelled | completed
intent	1..1	code	“order”
priority	0..1	code	normal | urgent | asap | stat
code	1..1	CodeableConcept	
-- coding	1..1	Coding	
-- -- SNOMED	1..1	Slice	
-- -- -- system	1..1	string	“http://snomed.info/sct”
-- -- -- code	1..1	code	“3457005”
-- -- -- display	0..1	string	“verwijzen van patiënt”
-- text	1..1	string	“Verwijzing”
description	0..1	string	
focus	0..1	Reference(ReferralRequest | CarePlan)	
for	0..1	Reference(nl-core-patient)	Reference to referred patient
authoredOn	0..1	dateTime	Date of referral submission
requester	0..1	BackboneElement	
-- agent	1..1	Reference(nl-core-practitioner)	Reference to the practitioner who sent the referral
-- -- extension		Extension	
-- -- -- practitionerRole		Extension(Reference(nl-core-practitionerrole))	Extension to relate the Practitioner to an Organisation, Location, HealthcareService, role, specialism, etc.
-- onBehalfOf	0..1	Reference(nl-core-organization)	Reference to the sending organization
owner	0..1	Reference(nl-core-organization)	Reference to the receiving organization
restriction	0..1	BackboneElement	
-- period	0..1	Period	
-- -- start	0..1	dateTime	Earliest date to start requested treatment or service
-- -- end	0..1	dateTime	Latest date to start requested treatment or service
input	0..*	BackboneElement	
-- demografieIdentificatie	0..1	Slice	
-- -- type	1..1	CodeableConcept	
-- -- coding	1..*	Coding	
-- -- -- LOINC	1..1	Slice	
-- -- -- -- system	1..1	string	“http://loinc.org”
-- -- -- -- code	1..1	code	“79191-3”
-- -- -- -- display	0..1	string	“Patient demographics panel”
-- -- text	1..1	string	“Demografie en identificatie”

-- -- valueString	1..1	string	“/Patient?_include=Patient:general-practitioner”

As described in the Notification-section every reference can be coded specific to the part. The codes of all HCIMs are in the table below.

HCIM	Code	System
Patient **MaritalStatus** **ContactPerson** **HealthProfessional**	79191-3	http://loinc.org
Payer	48768-6	http://loinc.org
TreatmentDirective	11291000146105	http://snomed.info/sct
AdvanceDirective	11341000146107	http://snomed.info/sct
FunctionalOrMentalStatus	47420-5	http://loinc.org
Problem	11450-4	http://loinc.org
LivingSituation	365508006	http://snomed.info/sct
DrugUse	228366006	http://snomed.info/sct
AlcoholUse	228273003	http://snomed.info/sct
TobaccoUse	365980008	http://snomed.info/sct
NutritionAdvice	11816003	http://snomed.info/sct
Alert	75310-3	http://loinc.org
AllergyIntolerance	48765-2	http://loinc.org
MedicationAgreement	16076005	http://snomed.info/sct
AdministrationAgreement	422037009	http://snomed.info/sct
MedicationUse2	422979000	http://snomed.info/sct
MedicalDevice	46264-8	http://loinc.org
Vaccination	11369-6	http://loinc.org
BloodPressure	85354-9	http://loinc.org
BodyWeight	29463-7	http://loinc.org
BodyHeight	8302-2	http://loinc.org
LaboratoryTestResult	15220000	http://snomed.info/sct
Procedure	47519-4	http://loinc.org
Encounter	46240-8	http://loinc.org
PlannedCareActivityForTransfer	18776-5	http://loinc.org

Appendix: Notification considerations

In the process of deciding the content of the notification several options have been up for review. This appendix has been added to inform about the options that were reviewed, and to a certain extent why they were ultimately not used.

Resource	Pro's / con's	Deciding factor
Bundle **type Collection**	- Communication of a (collection of) resource(s) is usually done using Bundle, because of its flexibility. - Light weight; this type forces minimalization of data. This way only clinical data can be transmitted. - Suits the narrative when changing to R5 alternatives. - Extensible with entry.link, to add more detail about the send resources.	The suggestion was made to not include the resources itself in this resource. But the collection explicitly needs the entry to contain the resource itself.
List	- Easy solution, conceptually ready for notification. - No support for search queries - No support for linked request resources - No real support for details on sender and/or receiver	Too much con's, which should really be supported for notification purposes.
AuditEvent	- A lot of space to go into detail which data is made available for whom - Limited support for search queries - No support for linked request resources - No support for recipient details	Purpose-build for auditing specific actions, not as a notification.
Consent	- Support for an end-date. - Links a notification to the authorization, while authorization should be concluded from the consent or access token. - No support for search queries - No support for linked request resources. - No real support for details on sender and/or receiver	Purpose-build to contain consent, not a notification. Would insinuate availability based on resource, while consent and access token are still needed to determine authorization.