



EHE FHIR Repository

Commercial Description

Version 5.1

1 Introduction

Electronic Healthcare Exchange (EHE) is a line of products fulfilling a variety of eHealth system needs, ranging from fundamental ones like infrastructure, security, and integration, over exchange and management of clinical documents and discrete medical information, to advanced functionalities like clinical decision support. Solutions made of different EHE products, alone or through integration with the existing infrastructure, support a wide range of processes in a healthcare system.

The EHE FHIR Repository is a specialized database that enables the management and storage of various types of data used in healthcare processes (clinical, administrative, public health, research, etc.), in accordance with HL7 FHIR R4 (4.0.1) standard [1].

Products such as EHE Medical Records Database [3] and EHE Document Registry and Repository [4] use this server to store their data in a database.

2 Supported Interfaces and Operations

The EHE FHIR Repository supports the following types of interfaces for data management and processing:

- **RESTFull API (Application Programming Interface)** – enables saving and retrieving healthcare data in the form of individual FHIR resources
- **FHIR Messages** – enables saving, retrieval and processing of health data using FHIR messages. Each FHIR message refers to a specific action that the FHIR repository must perform. Actions that must be performed are implemented in specialized modules and the EHE FHIR Repository is used to store data from messages and validate messages.
- **FHIR Documents** – enables saving and retrieving healthcare data in the form of clinical and administrative documents. Each document consists of one or more FHIR resources that together form a specific document. The FHIR document interface does not save individual FHIR resources separately, but saves entire FHIR documents, and using this interface, after saving the FHIR document, it is possible to search and retrieve only entire documents.
The product EHE Document Registry and Repository [4] is used to extract individual FHIR resources that make up a FHIR document in the form of discrete information (individual FHIR resources) and store them in the medical data repository.

2.1 RESTFull API Interface

RESTFull API interface enables the following types of operations:

- **operations related to instances of a particular FHIR resource** - An instance of a FHIR resource is a concrete entity or event in healthcare that is represented by a certain type of FHIR resource. For example, administrative data on patient John Doe or the drug Aspirin of 500 mg given to patient John Doe on October 5, 2023. These operations enable the management and processing of data about specific entities and events such as storing, searching, and retrieving.
- **operations related to the type of FHIR resource** – Different entities and events in healthcare are represented by different types of FHIR resources. For example, FHIR resource `CodableConcept`, which refers to code lists in healthcare, or `PlanDefinition`, which refers to certain patient treatment plans. Operations related to the type of FHIR resource enable functions that are defined and refer to a specific type of FHIR resource, i.e. entities or events in healthcare that are represented by that specific type of FHIR resource, e.g., validation of a specific code in the code list (a function that is defined specifically for FHIR `CodeSystem` resource) or executing a specific treatment plan for a specific patient (a function that is defined specifically for the FHIR resource `PlanDefinition`).
- **operations that are not related to the type of FHIR resource** – These operations do not refer to a specific type of FHIR resource, i.e., data that is stored according to a specific type of FHIR resource but refer to several types of FHIR resources. An example of this type of operation is data validation according to the definition of a certain FHIR resource.

2.1.1 Operations Related to Instances of Individual FHIR Resources

The EHE FHIR Repository supports the following RESTful API operations related to instances of individual FHIR resources:

- **reading/retrieving a FHIR resource instance (read)** – this operation allows retrieving a specific FHIR resource instance using the unique identifier of the FHIR resource instance.
- **deleting the FHIR resource instance (delete)** – this operation physically deletes the FHIR resource instance from the FHIR server. The FHIR resource instance to be deleted is defined using the unique identifier of the FHIR resource instance.
- **FHIR resource instance history (history)** – this operation retrieves the modification history of either a specific resource, all resources of a specific type, or all resources supported by the system. When saving and changing data in the EHE FHIR Repository, the history of all changes to a specific instance of the FHIR resource is saved, and with this operation it is possible to see that history or a specific version within that history. The FHIR resource instance for which the version history is to be retrieved is defined using a unique identifier of the FHIR resource instance.

- **modification of individual data within the FHIR resource instance (patch)** – this operation is used to modify certain data in the FHIR resource instance. As an alternative to updating the entire resource, clients can perform a data modification operation within the FHIR resource. With this operation, the entire FHIR resource is not sent, only the information on which attribute within the FHIR resource should be created, deleted or modified. This operation is useful because the client does not have to send a copy of all the data inside the FHIR resource, but only the ones that need to change. The instance of the FHIR resource whose data is to be modified is defined using the unique identifier of the instance of the FHIR resource.
- **reading/retrieving a specific version of the FHIR resource instance (read)** – this operation allows retrieving a specific version of the FHIR resource instance. The FHIR resource instance for which a specific version is to be retrieved is defined using a unique identifier of the FHIR resource instance.
- **update** - creates a new current version for an existing resource or creates an initial version if no resource exists for the given ID. In this operation, it is necessary to send the entire FHIR resource, that is, all the data of that resource, those that change and those that do not change. The FHIR resource instance to be modified is defined using a unique identifier of the FHIR resource instance.

2.1.2 Operations Related to the Type of FHIR Resource

The EHE FHIR Repository supports the following RESTful API operations related to the FHIR resource type:

- **creating a new instance of the FHIR resource type (create)** – this operation creates a new instance of the defined FHIR resource type. When creating a new instance, the server assigns a new unique identifier of that FHIR resource instance, which can later be used for operations related to the FHIR resource instance, such as retrieval, modification or deletion. If the user wants to have control over the identifier of the stored FHIR resource instance, the update operation of the FHIR resource instance (update) can be used instead of the operation of creating a new instance of the FHIR resource (create).
- **Search for instances of FHIR resources (search)** - this operation enables searching a set of instances of a certain type of FHIR resource based on search parameters. Permitted search parameters are defined by the standard for each type of FHIR resource, but it is also possible to expand them according to the expansion rules (Profiling). This operation can be combined with operations related to the FHIR resource instance and thus enable the so-called conditional operations. All operations related to FHIR resource instances can be conditional, the difference is that with conditional operations, the FHIR resource instance on which the operation is performed is not defined using the unique identifier of the FHIR resource instance but using search parameters.

When searching for instances of FHIR resources, it is possible to include in the request additional instructions that enable the inclusion of additional instances of FHIR resources of the same or different type that are related to the instance of FHIR resource being searched. For example, when searching for drugs given to a specific patient (MedicationAdministration FHIR resource), it is possible to request inclusion of the FHIR resource Patient, which contains information about the patient who was given the drug, but also the resource Practitioner, which contains information about the healthcare professional who administered the drug, or the resource Encounter that contains information about the visit during which that drug was administered.

The EHE FHIR Repository also enables the so-called indirect search, i.e. indirect retrieval of data. In this search mode, it is possible to search/retrieve data (instances of FHIR resources) according to the search parameters of the FHIR resource to which that instance of FHIR resource is linked. For example, the data on the drugs administered contains a link to the patient to whom the drug was administered. This type of search, for example, makes it possible to retrieve all patients who have received a certain drug.

- **terminology operations** – these operations enable the search of code lists and sets of concepts, validation of concepts (by code list or set of concepts) and translation of concepts. The following terminology operations are supported:
 - for code lists (CodeSystem)
 - access to all concept data (\$lookup)
 - concept validation (\$validate-code) – checks whether the concept is in the code list
 - access to all concepts with a specific property (\$find-matches)
 - for sets of concepts (ValueSet)
 - concept validation (\$validate-code) – checks whether the concept is in a certain set of concepts
 - population of a set of concepts (\$expand) – enables the inclusion of all concepts in a set of concepts in accordance with the rules for including concepts defined in the set of concepts
 - for concept maps
 - concept translation (\$translate)

- **FHIR document generation (Composition/\$document)** – enables the generation of a FHIR document (Bundle) based on the specification of the structure of a certain document (Composition)
- **retrieving all patient data (Patient/\$everything)** – enables retrieving all data (all types of FHIR resources) related to a specific patient
- **retrieving all visit data (Encounter/\$everything)** – allows retrieving all data related to a specific visit.

2.1.3 Operations not Related to the Type of FHIR Resource

EHE FHIR Repository supports the following RESTful API operations related to the type and/or instance of the FHIR resource:

- **capabilities retrieval** – this operation retrieves information about the capabilities of the FHIR server (which parts of the FHIR specification it supports, which operations, which extensions, which profiles, etc.).
- **group of data and transactions (batch, transaction)** – this operation allows sending a set of data and/or actions that are executed on the server in one request/response. Actions can be performed independently as a "batch" or as a single atomic "transaction" where the entire set of changes succeeds or fails as a single operation. Multiple actions can be sent to multiple resources of the same or different types, and they can be a mix of other interactions (e.g., read, search, create, update, delete, etc.).
- **FHIR Server search** – this operation enables searching for instances of several different types of FHIR resources that have the same search parameters. For example, searching for visits (Encounter) and medications given (MedicationAdministration) for a specific patient. Both resources have the same search parameter "Patient", so it is possible to search all visits and medications for a specific patient.
- **Search using GraphQL** – this operation allows defining search parameters and which data should be returned in the response using GraphQL notation.

2.2 Notifications

In addition to the mentioned interfaces and data management operations, the EHE FHIR Repository also supports sending notifications about changes to instances of FHIR resources.

This functionality allows users to subscribe to receive changes using the Subscription FHIR resource that defines the filtering rules (FHIR resource type and search parameters) and how notifications are sent. The EHE FHIR Repository supports sending notifications via the REST interface (rest-hook) and via message queues (Kafka topic message).

3 Interdependencies

The EHE FHIR Repository depends on EHE Infrastructure [2].

To implement the EHE FHIR Repository, it is necessary to provide the relational database PostgreSQL or Oracle and the operating system Ubuntu Linux.

EHE FHIR Repository product components can be installed on physical servers, virtual machines or containers.

4 Free and Open Source Software

This product uses Free and Open Source Software (FOSS) components with the following licenses:

- Apache Software License 2.0 [5]
- MIT License [6]
- Eclipse Distribution License [7]
- Eclipse Public License [8]
- Creative Commons CC0 [9]
- BSD License (2 clause and 3 clause) [10]
- Bouncy Castle License [11]
- Common Development and Distribution License [12]
- GNU Library General Public License [13]
- Mozilla Public License (MPL) [14]
- Elastic license v2 [15]

5 Version

The current product version is 5.1.

6 References

- [1] HL7 FHIR (Fast Healthcare Interoperability Resources) - a standard describing data formats and elements and an application programming interface for the exchange of electronic health records. It was created by Health Level Seven, an international health standards organization. The specification is available at <https://www.hl7.org/fhir/>.
- [2] EHE Infrastructure – standard Ericsson Nikola Tesla’s product which implements the functions necessary for the operation, internal communication and monitoring of the solution components

- [3] EHE Medical Records Database – standard Ericsson Nikola Tesla’s product which enables the management and storage of health and clinical data in accordance with the HL7 FHIR standard and the IHE QEDm integration profile
- [4] EHE Documents Registry and Repository - standard Ericsson Nikola Tesla’s product which enables the management and storage of health and clinical documents in accordance with the HL7 FHIR standard and the IHE MHD integration profile
- [5] Apache Software License 2.0
<https://www.apache.org/licenses/LICENSE-2.0.txt>
- [6] MIT License <https://opensource.org/licenses/MIT>
- [7] Eclipse Distribution License
<https://www.eclipse.org/org/documents/edl-v10.php>
- [8] Eclipse Public License <https://www.eclipse.org/legal/epl-v10.html>,
also <https://www.eclipse.org/legal/epl-2.0/>
- [9] Creative Commons CC0
<https://creativecommons.org/publicdomain/zero/1.0/>
- [10] BSD License <https://opensource.org/licenses/BSD-2-Clause>, also
<https://opensource.org/licenses/BSD-3-Clause>
- [11] Bouncy Castle License <https://www.bouncycastle.org/licence.html>
- [12] Common Development and Distribution License
<https://opensource.org/licenses/CDDL-1.0>
- [13] GNU Library General Public License
<https://www.gnu.org/licenses/old-licenses/lgpl-2.0.html>
- [14] Mozilla Public License (MPL)
<https://www.mozilla.org/media/MPL/2.0/index.48a3fe23ed13.txt>
- [15] Elastic license
<https://www.elastic.co/licensing/elastic-license>