

Übung: Transformation in HL7 FHIR mit Mirth-connect

Vor Übungsbeginn: 15 min

Docker Installation, z.B mit Docker Desktop

Mirth-connect Installation (benötigt Java)

Installationsanweisungen siehe GitHub: <https://github.com/IMISE/MI-Lab-CSV-FHIR-transformation>

Aufgabe 1: Mirth-connect

1. Anmeldung in Mirth-connect

Folgen Sie der Readme im GitHub, anschließend können Sie sich in Mirth-connect anmelden.

user: admin, *passwort:* admin

Nach der Erstanmeldung werden Sie aufgefordert, den User und das Passwort neu zu setzen. Folgen Sie den Anweisungen.

2. Erstellen von Channels

Gehen Sie zu *Channels* und unter Channel Tasks klicken Sie auf *New Channel*.

Fügen Sie jeweils einen Channel für Patient und Encounter hinzu.

3. Configuration der Channels

Editieren Sie die zuvor hinzugefügten Channels und führen Sie jeweils die folgenden Schritte durch:

3.1. Source Connector

Öffnen Sie den jeweiligen Channel und klicken Sie auf *Source*.

1. Setzen Sie den Connector Type auf "File Reader"
2. Ändern Sie unter "Source Settings" den Process Batch auf Yes.
Dadurch werden alle Reihen der CSV ausgelesen und nicht nur der Header.

Source Settings

Source Queue: OFF (Respond after processing) ▼

Queue Buffer Size: 1000

Response: None ▼

Process Batch: ☒ Yes ☐ No

Batch Response: ☐ First ☒ Last

Max Processing Threads: 1

3. Ändern Sie unter *File Reader Settings* das *Directory*:

- a. Patient Channel: „/opt/connect/appdata/MII-Patient/“
- b. Encounter Channel: „/opt/connect/appdata/MII-Fall/“

Das Verzeichnis „/opt/connect/appdata/“ in der Docker Umgebung ist nach außen gemappt auf das Verzeichnis „Setup/mirth-connect“.

4. Setzen Sie das *Filename Filter Pattern* auf „*.csv“, damit nur CSV-Dateien eingelesen werden.

3.2. Destination Connector

Damit Mirth Connect die transformierten FHIR-Daten auf einen Hapi FHIR Server laden kann, müssen Sie eine die Destination für den FHIR Server anpassen.

1. Klicken Sie auf *Destinations* und setzen Sie den *Connector Type* auf "HTTP Sender"
2. In den *HTTP Sender Settings*:
 - a. Setzen Sie die *URL* auf "<http://hapi:8080/fhir/>"
Denn wir befinden uns im Docker-Compose-Netzwerk.
 - b. Ändern Sie den *Content Type* zu "application/json", damit die Daten im JSON-Format gesendet werden.
 - c. Setzen Sie den *Content* auf
"`${JsonUtil.prettyPrint(${message.rawData})}`", damit die Nachrichten formatiert als JSON gesendet werden.

3.3. Data Types

Setzen Sie nun unter "Summary" die Data Types wie folgt: damit die Message im JSON Format gesendet wird:

Diese Schritte müssen für beide Channels (Patient, Encounter) durchgeführt werden.

4. Import von Transformers

Um die CSV Dateien nach FHIR zu transformieren, muss für die Channels ein Transformer hinzugefügt werden.

Klicken Sie jeweils unter "Source" auf "Edit Transformer" und dort auf "Import Transformer". Die zu importierten Transformer sind im Ordner "Material/Transformer/":

MII-Patient Transformer.xml und *MII-Encounter Transformer.xml*

5. Configuration der Transformer

Im Folgenden werden Spalten aus dem CSV Datensatz auf die entsprechenden FHIR-Felder in einem JSON-Message Template gemappt. Das JSON-Template ist eine Vorlage für die Nachrichten an einen FHIR Server.

Fügen Sie unter "Transformer Tasks" jeweils neue Schritte hinzu:

1. Für *Patient* mappen Sie folgendes nach FHIR:
 - a. Name_family
 - b. Gender
2. Für *Encounter* mappen Sie folgendes nach FHIR:

- a. Aufnahmenummer
- b. Referenz

Da jeder Fall einen Patienten referenziert soll jede Referenz mit "Patient?identifizier=" beginnen.

6. Deployen der Channels und laden der Daten

Speichern und deployen Sie die Channels. Kopieren Sie danach die entsprechenden CSV-Dateien der zu mappenden Daten (*MIMIC IV Patients.csv* und *MIMIC IV Fall.csv*) in die in Aufgabe 1.3a) erstellten Ordner.

Wichtig: Es ist wichtig, dass die Patienten zuerst geladen werden, da die Fälle eine Referenz auf vorhandene Patienten benötigen.

Hinweis: Durch das Deployen der Channels führen Inputs in die von Mirth-Connect 'Source Connector' überwachten Ordnern direkt zum Versenden von POST-Requests an den unter 'Destination Connector' verknüpften FHIR Server. Versendete Requests und ggf. Fehlermeldungen können unter 'View Messages' angesehen werden. Das erfolgreiche Übermitteln der Daten wird in Aufgabe 2 durch ClinFHIR überprüft.

Aufgabe 2: ClinFHIR

1. Öffnen Sie ClinFHIR lokal im Browser: <http://localhost:8000/launcher.html>
Fügen Sie den Hapi FHIR (Version R4) Server hinzu:
URL: <http://localhost:8090/fhir/>

Add new server

Name	<input type="text" value="Mi-Lab Hapi"/>	The name of the server (will be displayed in server selection dialogs)
Url	<input type="text" value="http://localhost:8090/fhir/"/>	The full url to the server (including port). Include a trailing /
FHIR Version	<input type="text" value="R4"/>	The version of FHIR that the server supports
Support \$everything	<input checked="" type="checkbox"/>	Set true if this server supports the \$everything operation against Patient (Needed for Patient Data viewer)
Is Terminology Server	<input type="checkbox"/>	Set true if this server supports Terminology services (especially \$expand)

4.0.1 Cancel Add

2. Sehen Sie sich mit 'Patient Viewer' die hinzugefügten Patienten und Encounter an und beantworten Sie die folgenden Fragen:
 - 2.1. Wie alt ist der Patient mit ID 10015860 (ID 58)?
 - 2.2. Wie viele Fälle gibt es zu dem Patienten mit dem Identifier 10003400 (ID 2)?
 - 2.3. Ist der Patient mit ID 10018845 verstorben? Lösung: Ja
3. Sehen Sie sich mit 'clinFHIR Server Query' die hinzugefügten Patienten und Encounter an und beantworten Sie die folgenden Fragen:
 - 3.1. Wie viele Patienten sind weiblich?
 - 3.2. Wie viele Patienten sind bereits verstorben?
 - 3.3. Wie viele Fälle gehören zu den bereits verstorbenen Patienten?