HL7Query Module

Overview

The aim of this project is to allow users to export OpenMRS data (encounters) as hl7 objects. It is built using Groovy templates, and can be extended / configured by users as per their wishes.

This document will cover the following information -

- What types of messages are currently supported by the hl7query module
- How may users extend on the hl7query module
- An introduction to the hl7query segments currently used in the module
- A technical overview of how the module is built
- Installation and configuration of the hl7query module
- Making requests

What types of messages are currently supported by the hl7query module?

The default version of the hl7query module supports only the export of ORUR01 messages.

This means that the first release of the module contains only hl7 templates that support only the generation of ORUR01 message segments. However, developers are free to write up their own custom hl7 message templates, which can be added to the module. Custom templates can be used to export OpenMRS data in any form of hl7 message.

How may users extend on the hl7query module?

Users may extend upon the default version of the hl7query module by adding custom hl7 templates of their own. This may be done via the 'Manage Templates' page.

However, this requires a strong knowledge of groovyscript.

Adding a new template can be done as follows -

1) Creating a new template from scratch
2) Editing an existing template

If a user wishes to edit an existing template, simply click on an existing template, edit it, and save as new.

To create a new template navigate to the 'Manage Templates' page, and add the new template

Understanding the 'Manage templates' Page
Following is an explanation of what each segment of the 'Manage templates' Page represents.

```
<table>
<thead>
<tr>
<th>Template Name</th>
<th>Entity</th>
<th>Language</th>
<th>Template</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

**Template name**: The name by which the template will be identified. Examples: Generic PV1, FullPV1 or AMPATHPV1

**Entity**: The hl7 segment which the template represents. Examples: PV1, MSH, OBR

**Language**: The language the template is written in. Currently, this may be only Groovyscript

**Template**: The actual Groovyscript code.

A Groovyscript template may also refer to another existing template by name.

**Note**: If the new template represents an combined hl7 message type (ORUR01, ADT etc.), and the use wishes to export data in that particular message type, then he / she must also edit the 'Template' global property to point to the given template.

### Making Requests

The hl7query module supports only GET requests. All GET requests are made on the HL7QueryController.java class.

A GET request can consist of the following optional parameters:

- `patientId` (The patient identifier number we are using)
- `idTypeUuid` (The UUID of the patient identifier we are using)
- `encounterUuid` (a UUID of the encounter which we want to retrieve)
- `startDate` (Start date to filter retrieved encounters)
- `endDate` (End date to filter retrieved encounters)

**A sample GET request would look as follows** -

```
/openmrs/module/hl7query/ORUR01.htm?patientId=100&idTypeUuid=8d79403a-c2cc-11de-8d13-0010c6dffd0f
```

The above request contains the patient id and the UUID of the identifier type. Other additional url parameters (specified above) can also be added to the URL as per the users wishes.

**Note**: please set an Implementation ID on OpenMRS for this requests to work.
Potential Improvements

Users may document potential improvements / fixes to the module in the following document -

Improvements and fixes

Downloads

https://addons.openmrs.org/#/show/org.openmrs.module.hl7query

Source code: https://github.com/OpenMRS/openmrs-module-hl7query

Screenshots

Release Notes

- 1.0 (initial release)
- TBD

About

This module was developed by Suranga K for Jembi (and during this sprint)