## Defining eCQMs using CQL and QUICK

### HL7 For-Comment Ballot

### Contents

1. **Introduction**
   1.1 Purpose
   1.2 Ballot Material
   1.3 Focus Areas for the Community

2. **Representing eCQMs using HQMF and CQL**
   2.1 Proportion Measures
   2.2 Continuous Variable Measures
   2.3 Stratification and Supplemental Data
   2.4 Conformance Requirements

3. **Representing eCQMs using FHIR and CQL**
   3.1 Measures with Multiple Populations
   3.2 Continuous Variable Measures
   3.3 Conformance Requirements

4. **Recommendations**

---

© 2014 Health Level Seven International. All rights reserved. Ballot cycle Dec 2014
1 Introduction

1.1 Purpose

The Clinical Quality Language (CQL) standard aims to unify the expression of logic for Electronic Clinical Quality Measures (eCQM) and Clinical Decision Support (CDS). This document defines an approach to using CQL with Health Quality Measures Format Release 2.1 (HQMF R2.1) for defining eCQMs.

The original intended scope for this document was to define the mechanism for combining CQL expressions with HQMF R2.1 documents. However, while developing this Implementation Guide (IG), we realized that the role of HQMF in this IG was reduced to defining measure metadata and high level population structure and that it would be straightforward to represent this information in a simpler form by profiling resources from the Fast Healthcare Interoperability Resources (FHIR) specification that reference the same CQL expressions.

Both of these approaches are described in this document, and we are soliciting comments from the community (see the Focus Areas for the Community section) on the appropriateness of each. Our recommended approach is to create a FHIR eCQM profile that uses CQL for expression of logic.

1.2 Ballot Material

This for-comment ballot review material includes the following items:

- For-Comment Ballot Material Overview (this guide)
- Other additional material (TBD)

1.3 Focus Areas for the Community

As part of this For-Comment ballot, we would like to solicit comments from the community on the topics listed below and any additional topics you would like to bring to our attention upon your review of the material. In addition, we would greatly appreciate concrete examples and clinical use cases that we can apply to validate the proposals.

1. On Format:
   1. On referencing CQL expressions from within HQMF R2 documents
   2. On additional FHIR Resources for Measures and Reports

2. On Expressions:
   1. On the required use of CQL
   2. On the should-also use of ELM

3. On the usability of the proposals:
   1. Are the approaches well-suited for Clinical Quality Measures?
   2. Would the FHIR approach be suitable for Clinical Decision Support?
   3. Are the approaches well-suited for encoding (not authoring) clinical expressions?

The following topics are not within scope of this proposal: FHIR Extensions and FHIR Profiles.
2 Representing eCQMs using HQMF and CQL

In HQMF R2.1, a CQM is formatted in XML as a QualityMeasureDocument with various metadata and components, including a dataCriteriaSection and a populationCriteriaSection. The population criteria section typically contains initialPopulationCriteria, denominatorCriteria, and numeratorCriteria sub-components amongst others.

The HQMF structure (abridged for clarity) is shown below:

```
<QualityMeasureDocument ...>
  <!-- lots of metadata -->
  <component>
    <dataCriteriaSection>
      <!-- data entry elements referenced in this quality-measure -->
      <!-- are listed here -->
      <entry ... />
    </dataCriteriaSection>
  </component>
  <component>
    <populationCriteriaSection>
      <component...>
        <initialPopulationCriteria ...>
          <precondition ...>
            <allTrue>
              <precondition ...>
                <!-- criteriaReference refers to 'entry' (line 7) -->
                <criteriaReference ... />
              </allTrue>
              <allTrue>
                <precondition>
                  <allTrue>
                    <precondition>
                      <initialPopulationCriteria>
                    </initialPopulationCriteria>
                  </precondition>
                </allTrue>
              </precondition>
            </allTrue>
          </precondition>
        </initialPopulationCriteria>
      </component>
      <denominatorCriteria ... />
    </populationCriteriaSection>
  </component>
  <component>
    <numeratorCriteria ... />
  </component>
</QualityMeasureDocument>
```

The dataCriteriaSection (lines 4-8) defines the patient data of interest in the measure as a set of entries. Each entry identifies specific types of data along with constraints that the data must meet. For example, an individual data criteria entry might identify encounters whose code is a member of a particular value set and whose effective time occurred during a specific interval.

The population criteria sub-components - initialPopulationCriteria (line 13), denominatorCriteria (line 25), and numeratorCriteria (line 28) -- reference the entry elements (line 7) from the dataCriteriaSection (line 4) and use a few logical operations such as allTrue (line 15) to define logical combinations of data that must exist for a patient to meet a particular criteria. For example, an initialPopulationCriteria might want to identify patients 18 years or older that had been seen by their doctor in the last year, it would do this using an allTrue element that
contained references to two data criteria: one to identify patients of 18 years or older, and another to identify patients that had an encounter during the last year.

With the introduction of CQL we are able to remove the data criteria elements (since their function is completely superseded by CQL) and simplify the population criteria (since CQL can also supersede the HQMF logical operators like allTrue). An abridged example of HQMF R2.1 referencing logical expressions defined in CQL is shown below in Figure 2, and the general concept is illustrated in Figure 4.

**Figure 2 - HQMF with Linked Expression Document**

```xml
<QualityMeasureDocument>
  <!-- ... abridged for clarity ... -->
  <relatedDocument typeCode="COMP">
    <expressionDocument>
      <id root="0DF0523B-463A-4011-ACD1-2DF89465B631"/>
      <text mediaType="application/cql">
        <reference value="http://hl7.org/expressions/CMS146v2_CQM.cql"/>
      </text>
    </expressionDocument>
  </relatedDocument>
  <component>
    <populationCriteriaSection>
      <!-- ... abridged for clarity ... -->
      <initialPopulationCriteria classCode="OBS" moodCode="EVN" ... >
        <id root="c75181d0-73eb-11de-8a39-0800200c9a66" extension=... />
        <code codeSystem="2.16.840.1.113883.5.1063" code="IPP">
          <displayName value="Initial Patient Population"/>
        </code>
        <precondition typeCode="PRCN">
          <criteriaReference moodCode="EVN" classCode="OBS">
            <id root="0DF0523B-463A-4011-ACD1-2DF89465B631" extension="CMS146.InInitialPopulation" />
          </criteriaReference>
        </precondition>
      </initialPopulationCriteria>
    </component>
  </populationCriteriaSection>
</QualityMeasureDocument>
```

Lines 3-10 in Figure 2 identify a CQL expression document (CMS146v2_CQM.cql) and assign an internal root identifier to it (0DF0523B-463A-4011-ACD1-2DF89465B631). Later, in the population criteria section (lines 12-28), we define our initial patient population (lines 15-26). Rather than referring to data criteria entries as in Figure 1, we instead refer to a namespaced CQL expression CMS146.InInitialPopulation (lines 22-23). Note that the root identifier of the criteria reference (line 22, 0DF0523B-463A-4011-ACD1-2DF89465B631) matches the internal identifier assigned to the CQL expression document CMS146v2_CQM.cql from line 5. In this example the criteria reference identifies the InInitialPopulation expression defined in the CMS146v2_CQM.cql expression document using the CMS146 namespace (which is declared inside the document itself). If we were to peek inside CMS146v2_CQM.cql we would find the following snippet where InInitialPopulation is explicitly defined:
What exactly is going on here? Let’s take a moment to step back and look at the general case, illustrated in Figure 4.

In Figure 4, the HQMF QualityMeasureDocument references a CQL expression script (#1). Later in the QualityMeasureDocument, any of the population criteria sections can reference a particular expression from the referenced CQL file. In Figure 4, the initialPopulationCriteria element (#2) references the Expression as part of its definition. The referenced expression in turn (#3) may include or call another expression in the same (or a different) CQL expression script.

For example, if we go back and take another look at Figure 3, we’ll notice that our InInitialPopulation expression is in turn referencing other expressions. In this case, InDemographic and HasTargetEncounter. If we were to go back and peek inside CMS146v2_CQM.cql again, we would find the following snippet where InDemographic is explicitly defined:

The HQMF R2.1 specification says in Section 4.4 that expression documents provide "an optional extensibility point for HQMF that allows use of alternate expression languages to supplement built-in expression capabilities. An implementation guide can further specify use of this extensibility point." Also, it states that "it is not to be used to embed the expression document."
Using this approach, we reference external CQL scripts using the expressionDocument element, and reference particular expressions using the criteriaReference inside population definitions. The entry data typically defined in the dataCriteriaSection of the HQMF document is no longer used.

CQL includes both a human-readable text representation and a machine-oriented XML representation. Both representations should be referenced from the HQMF to follow the HL7 tradition of supporting human-readability at a minimum (in this case, CQL) and a canonical representation for machine processing (in this case, ELM).

Using ELM, our InInitialPopulation expression would appear as such

**Figure 6 - ELM defining the initial population criteria (CMS146v2_CQM.elm)**

```xml
<def localId="114" name="InInitialPopulation" context="PATIENT">
  <annotation xsi:type="cqla:Annotation">
    <ns0:s r="114">define InInitialPopulation =
      <ns0:s r="113">
        <ns0:s r="111">InDemographic</ns0:s> and
        <ns0:s r="112">HasTargetEncounter</ns0:s>
      </ns0:s>
    </ns0:s>
  </annotation>
  <expression localId="113" xsi:type="And">
    <operand localId="111" name="InDemographic" xsi:type="ExpressionRef"/>
    <operand localId="112" name="HasTargetEncounter" xsi:type="ExpressionRef"/>
  </expression>
</def>
```

and our QualityMeasureDocument would reference both the CQL and the ELM as follows:

**Figure 7 - Measure referencing both CQL and ELM**

```xml
<QualityMeasureDocument>
  <!-- ... abridged for clarity ... -->
  <relatedDocument typeCode="COMP">
    <expressionDocument>
      <id root="ID-REFERENCING-CQL-DOCUMENT"/>
      <text mediaType="application/cql">
        <reference value="http://hl7.org/expressions/CMS146v2_CQM.cql"/>
        <translation mediaType="application/elm+xml">
          <reference value="http://hl7.org/expressions/CMS146v2_CQM.elm"/>
        </translation>
      </text>
    </expressionDocument>
  </relatedDocument>
</QualityMeasureDocument>
  <!-- ... abridged for clarity ... -->
```

© 2014 Health Level Seven International. All rights reserved. Ballot cycle Dec 2014
2.1 Proportion Measures

An HQMF R2.1 document representing a proportion measure will include one or more populationCriteriaSection elements each of which will include components selected from the following: initialPopulationCriteria, numeratorCriteria, denominatorCriteria, denominatorExceptionCriteria, denominatorExclusionCriteria and numeratorExclusionCriteria. The semantics of these components are unchanged from the HQMF R2.1 specification, the only difference is that each component references a single criteria encoded as a CQL expression.

The referenced CQL expressions return the count that a particular patient contributes to the population component. CQL expressions may return counts in a variety of formats as shown in Table 1.

**Table 1 - CQL expression return type and effective value for proportion measures**

<table>
<thead>
<tr>
<th>Return Type</th>
<th>Effective Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Integer</td>
<td>Integer value</td>
</tr>
<tr>
<td>Boolean</td>
<td>0 (false) or 1 (true)</td>
</tr>
<tr>
<td>Array</td>
<td>Length of array</td>
</tr>
</tbody>
</table>

Patient based measures will return 0 or 1 (false or true) for each population component whereas episode-of-care based measures may return zero or more for a given population. For example, consider two measures:

1. **Denominator**: All patients with condition A that had one or more encounters during the measurement period.
   
   **Numerator**: all patients with condition A that underwent procedure B during the measurement period.

2. **Denominator**: All encounters for patients with condition A during the measurement period.
   
   **Numerator**: all encounters for patients with condition A during the measurement period where procedure B was performed during the encounter.

The first measure is an example of a patient-based measure. Each patient may contribute at most one count to the denominator and numerator regardless of how many encounters they had. The second measure is an episode-of-care measure where each patient may contribute zero or more counts to the denominator and numerator depending on the number of encounters they had.

The HQMF ITMCNT measure attribute is not used to identify the items to count for measures conforming to this profile. Instead the CQL expressions should be written to return an appropriate count for each population depending on the measure type (patient or episode-of-care).

2.2 Continuous Variable Measures

An HQMF R2.1 document representing a continuous variable measure will include one or more populationCriteriaSection elements each of which will include components selected from the following: initialPopulationCriteria and measurePopulationExclusionCriteria. In addition it will also include one measureObservationSection with one or more measureObservationDefinition elements. The semantics of these components are unchanged from the HQMF R2.1 specification, the only difference is that each measure population component and each measure observation definition references a single criteria encoded as a CQL expression.

The CQL expression referenced from the measureObservationDefinition component returns an aggregate result computed from one or more values for each patient, an example is shown in Figure 8.
In the above, the measure reports the result of the MeasureScore CQL expression (line 15 in the HQMF, line 72 in the CQL). Note that the value element (line 11) of the measureObservationDefinition does not include an expression element as normally needed when using only native HQMF R2.1 constructs. Instead the MeasureScore CQL expression can directly perform the desired computation and simply return the aggregated result. In addition, all joins and aggregation are also computed directly in the CQL expression so no methodCode or join element is required in the measureObservationDefinition. Finally, note that use of CQL negates the need for a measurePopulationCriteria component since the data required by the CQL expression is defined by that expression, either directly, or by reference to other CQL expressions.
2.3 Stratification and Supplemental Data

Stratification and supplemental data elements are represented using a `stratifierCriteria` component. The semantics of this component is unchanged from the HQMF R2.1 specification, the only difference is that each child `criteriaReference` references a CQL expression that determines whether a given patient meets the criteria for that stratification. Figure 9 shows an example stratifier that stratifies results for four sub populations.

**Figure 9 - Sample stratifier**

```xml
<stratifierCriteria>
  <id root="aebb3a81-74da-21de-7a23-0800200c9a65"/>
  <precondition>
    <!-- Stratify: Women < 50 years of age -->
    <criteriaReference classCode="OBS" moodCode="EVN">
      <id root="ID-REFERENCING-CQL-DOCUMENT" extension="IsFemaleLessThanFifty"/>
    </criteriaReference>
  </precondition>
  <precondition>
    <!-- Stratify: Men < 50 years of age -->
    <criteriaReference classCode="OBS" moodCode="EVN">
      <id root="ID-REFERENCING-CQL-DOCUMENT" extension="IsMaleLessThanFifty"/>
    </criteriaReference>
  </precondition>
  <precondition>
    <!-- Stratify: Women >= 50 years of age -->
    <criteriaReference classCode="OBS" moodCode="EVN">
      <id root="ID-REFERENCING-CQL-DOCUMENT" extension="IsFemaleFiftyOrMore"/>
    </criteriaReference>
  </precondition>
  <precondition>
    <!-- Stratify: Men >= 50 years of age -->
    <criteriaReference classCode="OBS" moodCode="EVN">
      <id root="ID-REFERENCING-CQL-DOCUMENT" extension="IsMaleFiftyOrMore"/>
    </criteriaReference>
  </precondition>
</stratifierCriteria>
```

A shared CQL library could supply definitions of common stratifiers to avoid each measure redefining them.
2.4 Conformance Requirements

An HQMF R2.1 document conforming to this implementation guide:

- **SHALL NOT** include a dataCriteriaSection
- **SHALL** include one or more expressionDocument elements that:
  - **SHALL** contain a child text element that:
    - **SHALL** have a mediaType attribute value of application/cql
    - **SHALL** include a child reference element whose value contains a URI (relative or absolute) that identifies the CQL expression document
    - **SHALL** include a translation element with a mediaType attribute value of application/elm+xml that:
      - **MUST** include a child reference element whose value contains a URI (relative or absolute) that identifies an ELM expression document whose content **MUST** be semantically equivalent to the corresponding CQL expression document referenced by the parent text element.
  - **SHALL NOT** include HQMF logical operators within population criteria, instead population criteria **SHALL** each reference a single CQL expression.
    - CQL expressions referenced from populationCriteriaSection components: initialPopulationCriteria, numeratorCriteria, denominatorCriteria, denominatorExceptionCriteria, denominatorExclusionCriteria, numeratorExclusionCriteria, and measurePopulationExclusionCriteria **SHALL** use PATIENT context and **MUST** be executed within the context of a single patient record at a time.
    - CQL expressions referenced from measureObservationDefinition elements: **SHALL** use POPULATION context and **MUST** be executed within the context of a population of patients.
  - **SHALL NOT** include HQMF logical operators within stratifier criteria, instead each child precondition **SHALL** reference a single CQL expression.

3 Representing eCQMs using FHIR and CQL

In this alternate approach, we reuse the FHIR Query resource to represent eCQMs, adopt the standard FHIR approach of using bundles (in XML, as an ATOM feed) for the eCQM reports, and specify the use of CQL for defining clinical expressions.

In Figure 10, the FHIR Query resource defines an eCQM in a very similar manner to the approach used in Section 2 except using FHIR conventions. The FHIR Query contains a list of parameters. Here, the parameters represent either an expression document import, defining a population criteria, measurement period, or stratification and supplemental data.
In Figure 10, we have a standard FHIR Query with a language, text narrative, identifier, and appropriate parameter elements. For representing eCQMs, the text narrative section contains the same HTML narrative typically generated via XSLT from an HQMF document, except that it is marked up
with microdata. Microdata is a specification to add machine-processable metadata and semantic meaning to a document. In this way, we can tag metadata about the measure in the narrative, such as title, status, effective time, version number, author, responsible organization, custodian, and so on and so forth.

Using parameter elements, we specify the expression documents used (1..*) in this eCQM in lines 17-22. In this case, the first is the CQL representation, and an optional extension supplies the ELM representation. Because parameter elements of the same "name" (represented by the parameter.url attribute) are allowed in FHIR, we can include as many expression documents as needed.

Additional parameter elements are used to define the population criteria (lines 23, 26, and 29) while the values themselves reference a CQL namespace (aka library) and expression (lines 24, 27, and 30).

We can also override the default measurement period (specified in the CQL document) by specifying a FHIR valuePeriod of type Period (lines 32-37).

Finally, we can specify stratification criteria (lines 38-40) and supplemental data elements (lines 41-46). Stratification criteria are supplied as a comma-separated list of named CQL expressions (notice the explicit library or namespace, such as CMS146.AgesUpToNine) and FHIR resource attributes (in our example, Patient.gender). These values are comma-separated in the same convention as other FHIR composite search parameters. When the stratification criteria is an expression, that expression defines a single stratification group. When the stratification criteria is a FHIR resource attribute, there will be as many stratification groups as values. For example, specifying Patient.gender will yield four stratification groups, since FHIR has four gender codes (in the development branch): male, female, other, and unknown.

Supplemental data elements are also specified using a parameter (0..*), where we can specify additional FHIR data attributes or elements to be included in the results. In this example, patient gender and whether or not they are still alive.

The results of the query are returned synchronously if the Query was a POST operation, or asynchronously if FHIR Messaging was used. In the former case, the result is the posted Query with an additional response element containing the results of the eCQM calculations. In the latter case, the result is a bundle (in XML, as an ATOM feed) containing the submitted Query object and the results of the eCQM calculations.

The results of the eCQM calculations, being designed by the HL7 CQI Working Group and not explicitly addressed here, should contain references to the relevant Patient and other FHIR resources used in the eCQM (possibly paged) as well as a summary resource providing population counts and so forth.

### 3.1 Measures with Multiple Populations

In practice there are real-world examples of eCQMs with multiple patient populations (for example, CMS172v4, which specifies eight different initial populations, eight different denominators, numerators, and so forth).

To represent that in FHIR, we create a single Query with the appropriate microdata in the text narrative section of the resource. Then, in the contained portion of the Query (inherited from base Resource) we include all the sub-queries required to define that measure (for example, we would have eight Query resources contained within a single master Query for the CMS172v4 measure).
In Figure 11, for a measure with multiple population groups, we see that all of the measure metadata is encoded as microdata inside the `text` narrative (lines 3-15). The top-level `Query` imports all expression documents (lines 17-22) and overrides any expressions that will have scope across all the subqueries (in this example, `measurementPeriod` on lines 23-28). Finally, the `contained` element contains all of the sub-queries. Each sub-query could import its own expression documents (which would only be scoped to that query) or they can reference expressions from the documents included in the parent `Query`. For
example, CMS172.InInitialPopulation1 (line 33) was defined in the CMS172v4_CQM.cql document (line 18) included in the parent Query.

### 3.2 Continuous Variable Measures

Continuous variable measures also include a "Measure Observations" section. This section defines variables (for example, time from check in to time of antibiotic administration) used to score particular aspects of performance. Measure observations are not population criteria, in that they do not determine whether or not a patient is to be counted in a measure. Rather, measure observations are data elements that are to be collected on patients meeting the population criteria within a continuous variable measure.

Figure 12 is an example of the "Measure Observations" section from the hospital measure CMS55/NQF0495, median time from arrival to departure for admitted emergency room patients, and Figure 13 is the related CQL expressions that define the continuous variable.

**Figure 12 - Continuous Variable / Measure Observation Example**

```
<Query xmlns="http://hl7.org/fhir">
  <!-- abridged -->
  <parameter url="http://hl7.org/fhir/query#expressionDocument">
    <valueUri value="http://hl7.org/expressions/CMS55v1_NQF0495.cql"/>
    <extension url="http://hl7.org/fhir/query#expressionDocument-ELM">
      <valueUri value="http://hl7.org/expressions/CMS55v1_NQF0495.elm"/>
    </extension>
  </parameter>
  <!-- abridged -->
  <parameter url="http://hl7.org/fhir/query#measureScore">
    <valueString>CMS55.MeasureScore</valueString>
  </parameter>
  <!-- abridged -->
</Query>
```

Similar to the FHIR examples in Figure 10 and Figure 11, this measure references an external CQL document (and the optional ELM representation) in lines 3-8. The standard parameter defining a "Measure Observation" is measureScore and is declared in line 9, and the namespaced expression CMS55.MeasureScore is referenced in line 10.

**Figure 13 - CQL defining the measure observation (CMS55v1_NQF0495.cql)**

```
define MeasureObservation = EDEncounters E
  where E.facilityLocationArrivalDateTime is not null 
  and E.facilityLocationDepartureDateTime is not null
  return minutes between E.facilityLocationArrivalDateTime
  and E.facilityLocationDepartureDateTime
context POPULATION

define MeasureScore = Median(MeasureObservation)
```

The expression CMS55.MeasureScore is declared in line 72 of the CQL document (it appears in the code listing without the namespace). That expression in turn references other expressions (for example, MeasureObservation in line 64-68). Other expressions being referenced in Figure 12 appear elsewhere in the CQL document.
3.3 Conformance Requirements

A FHIR Query conforming to this implementation guide:

- **SHALL** include a text narrative element that:
  - **SHALL** include an HTML narrative, tagged with metadata (full definition of tags is TBD) using microdata.
  - **SHALL** include one or more expressionDocument parameter elements that:
    - **SHALL** specify the url as http://hl7.org/fhir/query#expressionDocument
    - **SHALL** include a child valueString element whose value contains a URI (relative or absolute) that identifies the CQL expression document
    - **SHOULD** include an extension element with a url attribute value of http://hl7.org/fhir/query#expressionDocument-ELM that:
      - **MUST** include a child valueUri element whose value contains a URI (relative or absolute) that identifies an ELM expression document whose content **MUST** be semantically equivalent to the corresponding CQL expression document referenced by the parent valueUri element.
  - **SHALL** include zero or one (0..1) http://hl7.org/fhir/query#initialPopulationCriteria parameters, referencing a library/namespace qualified CQL expression.
  - **SHALL** include zero or one (0..1) http://hl7.org/fhir/query#numeratorCriteria parameters, referencing a library/namespace qualified CQL expression.
  - **SHALL** include zero or one (0..1) http://hl7.org/fhir/query#numeratorExclusionCriteria parameters, referencing a library/namespace qualified CQL expression.
  - **SHALL** include zero or one (0..1) http://hl7.org/fhir/query#denominatorCriteria parameters, referencing a library/namespace qualified CQL expression.
  - **SHALL** include zero or one (0..1) http://hl7.org/fhir/query#denominatorExclusionCriteria parameters, referencing a library/namespace qualified CQL expression.
  - **SHALL** include zero or one (0..1) http://hl7.org/fhir/query#denominatorExceptionCriteria parameters, referencing a library/namespace qualified CQL expression.
  - **SHALL** include zero or one (0..1) http://hl7.org/fhir/query#measurePopulationCriteria parameters, referencing a library/namespace qualified CQL expression.
  - **SHALL** include zero or one (0..1) http://hl7.org/fhir/query#measurePopulationExclusionCriteria parameters, referencing a library/namespace qualified CQL expression.
  - **SHALL** include zero or more (0..1) http://hl7.org/fhir/query#measureScore parameters, referencing a library/namespace qualified CQL expression, if the eCQM represents a continuous variable measure.
  - **SHALL** include zero or one (0..1) http://hl7.org/fhir/query#stratification parameters, as a comma-separated list of named CQL expressions and FHIR search parameters.
  - **SHALL** include zero or more (0..*) http://hl7.org/fhir/query#supplementalData parameters, each specifying additional FHIR data attributes or elements to be included in the results.
4 Recommendations

The original intended scope for this document was to define the mechanism for combining CQL expressions with HQMF R2.1 documents. However, while developing this approach, we realized that the problem was essentially defining measure metadata and high level population structure, and that it would be straightforward to represent this information in a simpler form by profiling resources inside FHIR that reference the same CQL expressions.

Our recommended approach is to create a FHIR eCQM profile that uses CQL for expression of logic.

---

1. CMS Guide for Reading Eligible Professional (EP) and Eligible Hospital (EH) eMeasures, Version 4, May 2013