HL7 Version 3 Implementation Guide:
Clinical Quality Language (CQL)-based
Health Quality Measure Format (HQMF)
Release 1, STU 2.1 - US Realm

September 2017

HL7 Standard for Trial Use (STU)

Volume 1 - HQMF representation of CQL based Measures

Sponsored by:
Clinical Quality Information Work Group

Acknowledgements

Copyright

Contents

List of Figures

List of Tables

List of Snippets

List of Conformance Requirements

September 2017

Specification Materials

Filename	Description	Ballot Applicability
V3_IG_CQL_HQMF_R1 D2.1 _2017AUG_Vol1.pdf	This guide - HQMF representation of CQL based Measures	STU
V3_IG_CQL_HQMF_R1_D2.1 _2017AUG_Vol2.pdf	Using QDM in CQL based Measures	STU
V3_IG_CQL_HQMF_R1_D2.1 _2017AUG_Vol3.pdf	QDM based HQMF templates	STU
HQMF N1 XSD EXT/	CQL specific extensions to HQMF	Informative
examples/		
sample_eMeasure.xml	HQMF structure	Informative
Terminology_CQL.cql	Sample CQL file used to illustrate Terminology	Informative
Terminology_eMeasure.xml	Sample HQMF file used to illustrate Terminology	Informative
examples/EXM146v4/		
Common-2.0.0_CQL.cql	Sample CQL file that contains the CQL expressions referenced by EXM146v4_CQL.cql	Informative
Common-2.0.0_ELM.xml	elm+xml representation of Common-2.0.0_CQL.cql	Informative
Common-2.0.0_ELM.json	elm+json representation of Common-2.0.0_CQL.cql	Informative
EXM146v4_CQL.cql	Sample CQL file	Informative

`EXM146v4 ELM.xml`	`elm+xml` representation of `EXM146v4 CQL.cql`	Informative
`EXM146v4_ELM.json`	`elm+json` representation of `EXM146v4_CQL.cql`	Informative
`EXM146v4_eMeasure.xml`	HQMF representation of `EXM146v4_CQL.cql`	Informative
`EXM146v4_eMeasure.html`	Human Readable representation of `EXM146v4_CQL.cql`	Informative
`examples/TestCMS55v5 Artifacts/`		
`TestCMS55v5_CQL.cql`	Sample CQL file for a continous variable measure -	Informative
`TestCMS55v5_ELM.xml`	`elm+xml` representation of `TestCMS55v5_CQL.cql`	Informative
`TestCMS55v5_eMeasure.html`	Human Readable representation of `TestCMS55v5_CQL.cql`	Informative
`TestCMS55v5_eMeasure.xml`	HQMF representation of `TestCMS55v5_CQL.cql`	Informative
`examples/TestComposite/`		
`Composite_eMeasure.xml`	Sample HQMF file for a Composite Measure	Informative
`Test122v5 Artifacts/`		
`TestCMS122v5_CQL.cql`	Sample CQL file used to define part of a composite measure	Informative
`TestCMS122v5_ELM.xml`	`elm+xml` representation of `TestCMS122v5_CQL.cql`	Informative
`TestCMS122v5_ELM.json`	`elm+json` representation of `TestCMS122v5_CQL.cql`	Informative
`TestCMS122v5_eMeasure.xml`	HQMF representation of `TestCMS122v5_CQL.cql`	Informative
`TestCMS122v5_eMeasure.html`	Human Readable representation of `TestCMS122v5_CQL.cql`	Informative
`Test131v5 Artifacts/`		
`TestCMS131v5_CQL.cql`	Sample CQL file used to define part of a composite measure	Informative
`TestCMS131v5_ELM.xml`	`elm+xml` representation of `TestCMS131v5_CQL.cql`	Informative
`TestCMS131v5_ELM.json`	`elm+json` representation of `TestCMS131v5_CQL.cql`	Informative
`TestCMS131v5_eMeasure.xml`	HQMF representation of `TestCMS131v5_CQL.cql`	Informative
`TestCMS131v5_eMeasure.html`	Human Readable representation of `TestCMS131v5_CQL.cql`	Informative
`examples/TestRiskAdj v5 1/`		
`TestRiskAdj_CQL.cql`	Sample CQL file illustrating risk adjustment	Informative

`TestRiskAdj_ELM.xml`	elm+xml representation of `TestRiskAdj_CQL.cql`	Informative
`TestRiskAdj_ELM.json`	elm+json representation of `TestRiskAdj_CQL.cql`	Informative
`TestRiskAdj_eMeasure.xml`	HQMF representation of `TestRiskAdj_CQL.cql`	Informative
`TestRiskAdj_eMeasure.html`	Human Readable representation of `TestRiskAdj_CQL.cql`	Informative

Table 1: Contents of the Packages.

1 Introduction

1.1 Purpose

The Institute of Medicine (IOM) defines quality as: "The degree to which health services for individuals and populations increase the likelihood of desired health outcomes and are consistent with current professional knowledge." [1] For care quality to be evaluated, it must be standardized and communicated to the appropriate organizations. To that end, this Implementation Guide has been written to provide guidance for authoring electronic Clinical Quality Measures (eCQMs) utilizing the following standards:

- Quality Data Model (QDM) v5.3 [2]

- Clinical Quality Language (CQL) R1.1 [3]

- Health Quality Measures Format Release 1 Normative (HQMF R1 Normative) [4]

Although the specification is based on the 1.1 version of CQL, backwards compatible future versions of CQL can be used as well.

Note that HQMF releases have typically been referred to by their STU version, so HQMF R2.1 was referring to the STU version (2.1), not the full release version, which is still 1. Now that HQMF has been released as a normative specification, the STU version is dropped. Except where noted specifically, references to HQMF in this guide are to the normative release 1 version.

Except where noted, material from the above specifications is not reproduced here.

1.2 Structure of this Guide

Three volumes comprise this *HL7 Version 3 Implementation Guide: Clinical Quality Language (CQL)-based Health Quality Measure Format (HQMF), Release 1 STU 2.1 (US Realm), Standard for Trial Use*:
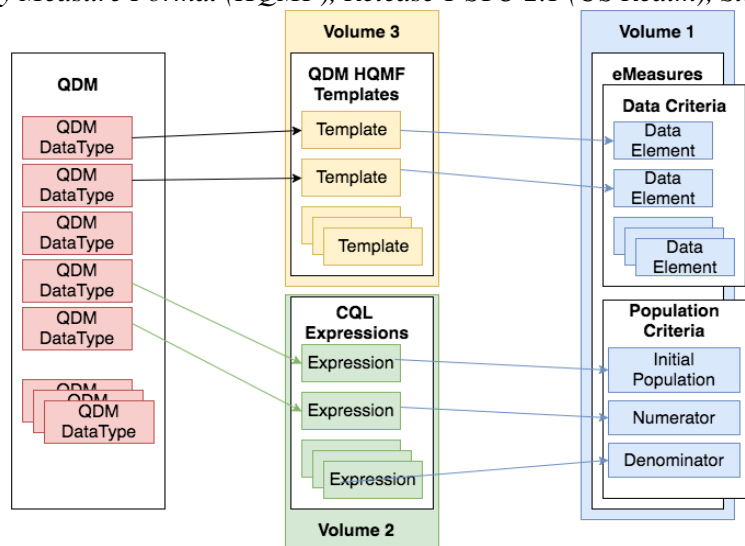
Figure 1: Relationship between QDM, CQL, eMeasure, and the volumes of this IG.

Volume 1 : Provides narrative introduction, background material, and conformance requirements for representing CQL-based eCQMs in HQMF.

Volume 2 : Describes how to incorporate version 5.3 of the Quality Data Model into a CQL-based eCQM in accordance with accepted formatting and usage conventions.

Volume 3 : Contains the HQMF templates for QDM data elements, necessary for constructing QDM+CQL-based HQMF measures.

1.3 Structure of this Volume

In this section we present an outline of this volume of this *HL7 Version 3 Implementation Guide: Clinical Quality Language (CQL)-based Health Quality Measure Format (HQMF), Release 1 STU 2 (US Realm), Standard for Trial Use*.

This volume is divided into 6 chapters. Chapters 2 - 6 describe how to construct a CQL based HQMF document and follow the structure of an HQMF document (metadata, data criteria, population criteria, stratification criteria).

Chapter 1 provides an introduction to this IG, gives a brief history of the IG, describes some of the standards upon which this IG was built, and briefly references other standards and tools present in the ecosystem of which this IG is part.

Chapter 2 provides an overview of HQMF structure, how to reference CQL documents in the HQMF document, and how to specify control variables (measure period).

Chapter 3 describes how to reference codes and valuesets in CQL and the accompanying HQMF.

Chapter 4 describes how to construct the `dataCriteriaSection` of the HQMF document.

Chapter 5 discusses measure scoring types, how to specify population criteria in the HQMF document using CQL, and how to specify measure populations in CQL. There are also sections discussing stratification, inclusion of supplemental data, and defining risk adjustment variables.

Chapter 6 contains a discussion of composite measures and HQMF examples of composite measures.

1.4 Audience

The audience for this IG includes software developers of the Measure Authoring Tool (MAT); measure developers who will specify clinical quality measures in HQMF; software developers and implementers who will implement the quality measures specified in HQMF in their institutions or in their vendor products; and local, regional, and national quality reporting agencies who wish to receive and process quality report documents that are based on measures specified in HQMF.

1.5 Approach

The approach taken here is consistent with balloted IGs for Clinical Document Architecture (CDA). These publications view the ultimate implementation specification as a series of layered constraints. HQMF itself

is a set of constraints on the Health Level Seven (HL7) Reference Information Model (RIM). IGs such as this add constraints to HQMF through conformance statements that further define and restrict the sequence and cardinality of HQMF objects and the vocabulary sets for coded elements.

This IG is STU 2.1 of the CQL-based HQMF Standard for Trial Use (STU). Section 1.8 describes the development of this STU.

1.6   Scope

This IG is a conformance profile, as described in the "Refinement and Localization" [9] section of the HL7 Version 3 Interoperability Standards. The base standard for this IG is the HL7 Health Quality Measures Format. This IG (Volumes 1, 2, and 3) does not describe every aspect of HQMF Release 1 Normative. Rather, it defines constraints on the base HQMF used in a CQL-based HQMF document in the US Realm. Additional optional HQMF elements, not included here, can be included and the result will be compliant with the specifications in this guide.

1.7   Conventions

The keywords SHALL, SHALL NOT, SHOULD, SHOULD NOT, MAY and NEED NOT in this document are to be interpreted as described in the HL7 Version 3 Publishing Facilitator's Guide.

- **SHALL**: an absolute requirement for the particular element. Where a SHALL constraint is applied to an XML element, that element must be present in an instance, but may have an exceptional value (i.e., may have a nullFlavor), unless explicitly precluded. Where a SHALL constraint is applied to an XML attribute, that attribute must be present, and must contain a conformant value.

- **SHALL NOT**: an absolute prohibition against inclusion

- **SHOULD/SHOULD NOT**: best practice or recommendation. There may be valid reasons to ignore an item, but the full implications must be understood and carefully weighed before choosing a different course

- **MAY/NEED NOT**: truly optional; can be included or omitted as the author decides with no implications

1.8   Background

This Implementation Guide (IG) defines an approach to using CQL with Health Quality Measures Format Release 1 Normative (HQMF) [4] for defining eMeasures.   This IG is split into three volumes, volume 1 (this volume) contains the instruction on how to use HQMF with CQL, volume 2 describes how to use QDM with CQL, and volume 3 contains all the necessary QDM based HQMF templates for defining a QDM based eCQM.

This Implementation Guide is the successor of the QDM-based HQMF IG R1.4 (Figure 2a) and the CQL-Based HQMF IG R1 STU1 (Figure 2b).
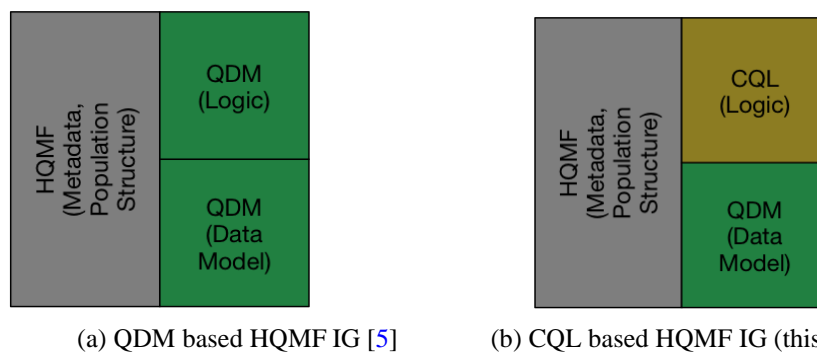
(a) QDM based HQMF IG [5]   (b) CQL based HQMF IG (this IG)
Figure 2: Relationship between QDM based and CQL based HQMF IG's.

1.8.1   Clinical Quality Language R1.1

Clinical Quality Language R1.1 (CQL) is an HL7 draft standard for trial use (DSTU)[3]. It is part of the effort to harmonize standards between electronic clinical quality measures (eCQMs) and clinical decision support (CDS). CQL provides the ability to express logic that is human readable yet structured enough for processing a query electronically.

1.8.2   QDM based HQMF IG R1.4

The QDM based HQMF IG R1.4 [5] published October 2016 described how to construct an HQMF measure using QDM data elements and QDM logic (Figure 2a). That IG was built using QDM version 4.3.

1.8.3   CQL based HQMF IG R1 STU1

The first version of the CQL based HQMF IG was released in September 2015 and was intended to be used in conjunction with the QDM based HQMF R1 IG. Since 2015, the community and the standards have evolved and QDM v5.02 no longer contains expression logic, ceding this functionality to CQL. As such, no stand alone QDM based HQMF IG will be built upon future versions of QDM starting with QDM v5.02. Rather, this IG is intended to be the sole guide describing how to use QDM, CQL, and HQMF in combination (Figure 2b).

A result of replacing QDM-based logic with CQL is that all QDM logic elements previously encoded in HQMF are replaced with CQL. This means that QDM data criteria specify only the data of interest (e.g. value sets, effective time, properties) for the eMeasure, and the previous use of QDM expressions that captured interrelationships between data criteria (such as "`starts after end of`") or identified subsets of data (such as `min`, `max`, `last`, and `first`) are now represented with CQL expressions. This IG documents the full approach in detail starting in Chapter 2.

A separate HL7 initiative will produce an IG that covers the use of Fast Healthcare Interoperability Resources (FHIR), CQL, FHIR Quality Profiles, and other emerging approaches to define eMeasures.

1.8.4   HQMF

HQMF is a structured document markup standard[*] ". . . for representing a health quality measure as an electronic document. A quality measure is a quantitative tool to assess the performance of an individual or organization's performance in relation to a specified process or outcome via the measurement of an action, process, or outcome of clinical care. Quality measures are often derived from clinical guidelines and are designed to determine whether the appropriate care has been provided given a set of clinical criteria and an evidence base." [4, §1.1]

HQMF defines a header for classification and management of the quality measure as well as important metadata. HQMF also defines a document body that carries the content of the quality measure.

Through standardization of a measure's structure, metadata, definitions, and logic, the HQMF ensures measure consistency and unambiguous interpretation. A health quality measure encoded in the HQMF format is referred to as an "eMeasure" or an electronic clinical quality measure (eCQM). Standardization of document structure (e.g., sections), metadata (e.g., author, verifier), and definitions (e.g., numerator, initial population) enables a wide range of measures currently existing in a variety of formats to achieve consistency. This formal representation of the clinical, financial, and administrative concepts and logic within an eMeasure produce unambiguous interpretation and consistent reporting.

During the past few years, National Quality Forum (NQF), through the Health Information Technology Expert Panel (HITEP), developed the Quality Data Model (QDM) for data representation in quality measures; and HL7 developed the HQMF Release 1 (R1) Draft Standard For Trial Use (DSTU). NQF, working with CMS, applied the QDM to HQMF R1, and implemented this solution in the Measure Authoring Tool (MAT) [7] . The team did this by creating patterns for each QDM data type and QDM attribute, mapping them to the HL7 Reference Information Model (RIM), and using standard vocabularies. The QDM-based HQMF R1 was further refined (and the HQMF R1 DSTU was extended) in collaboration with measure developers through the eMeasure Issues Group (eMIG), a consensus body of eMeasure developers and stewards convened by CMS. The resulting QDM-based (extended) HQMF R1 was implemented in the MAT, and served as the basis for the creation of Meaningful Use 2014 eMeasures.

This approach was subsequently standardized in the QDM-based HQMF IG [5] for HQMF R1 STU2 [4] along with the full list of templates for the QDM data types and QDM attributes in Volume 2 of the QDM-based HQMF IG, so that they could meet the needs of Meaningful Use eMeasures, and so that the QDM-based HQMF strategy would be governed by an open HL7 consensus process (as opposed to being driven by the MAT tooling implementation).

1.8.5   HQMF Release 1 STU1 vs STU2

HQMF R1 STU1 was balloted in the September 2009 ballot cycle as a DSTU; it was supported by volunteer efforts and through the NQF contract with the US Department of Health and Human Services (HHS) to promote the effective use of EHR systems. The DSTU period for HQMF R1 STU1 was two years.

HQMF R1 STU2 was sponsored by the Center for Clinical Standards and Quality of CMS in partnership with HL7 and the Office of the National Coordinator (ONC). A driver for developing HQMF R1 STU2 was the need to make HQMF more amenable to automated machine processing. ONC's Standards and Interoperability (S&I) Framework Query Health Technical Workgroup co-hosted project meetings. This IG is developed based on HQMF R1 STU2.1 that was published in August 2014 [4].

---

[*]HQMF is not an HL7 V3 Clinical Document Architecture (CDA) standard, but is similar to CDA in being a structured document markup standard.

1.9 Other Related Tools and Standards

This section describes other tools, standards, and resources related to electronic Clinical Quality Measures.

1.9.1 Quality Data Model

Volume 1 of this IG is intended to be as model agnostic as possible. However, the examples used have incorporated QDM [2]. Further discussion of incorporating QDM into CQL based HQMF measures is discussed in Volume 2 of this IG.

1.9.2 Relationship to Quality Reporting Document Architecture

Volumes 2 and 3 discuss how to incorporate QDM into CQL based HQMF measures. A standard reporting mechanism for QDM based is the Quality Reporting Document Architecture [6]. Further discussion of QRDA is available in Volume 2 of this IG.

1.9.3 Measure Authoring Tool

The MAT is a web-based software authoring tool that measure developers use to create eMeasures [7]. The authoring tool allows measure developers to create eMeasures in a highly structured format using the QDM and healthcare industry standard vocabularies. The MAT was developed by NQF under a contract with HHS, and has been publicly available through NQF since September 2011. All Meaningful Use Stage 2 measures are authored in MAT to ensure consistency in creating header metadata, population criteria, data criteria, etc. Effective January 2013, CMS assumed ownership of the MAT and has contracted with Health Care Innovation Services, a joint venture between Telligen and Net-Integrated Consulting for the ongoing development, maintenance, and support.

The QDM-based building-block approach to eMeasures, which is described in this IG, was implemented in the MAT. It will be updated in accordance with this guide.

1.9.4 NLM Value Set Authority Center

The Value Set Authority Center (VSAC) [10] is provided by the National Library of Medicine (NLM), in collaboration with the ONC and CMS. The VSAC currently serves as the authority and central repository for the official versions of value sets that support Meaningful Use eCQMs. Through the VSAC, NLM draws upon the UMLS Metathesaurus and its responsibility as the central coordinating body for clinical terminology standards within the HHS to assure the ongoing validity and accuracy of the value sets. NLM launched the VSAC Authoring Tool on October 31, 2013. Value sets for eMeasures can now be authored directly in VSAC.

1.9.5 CMS Measures Management Blueprint

CMS has developed a standardized approach for the development and maintenance of the quality measures it uses in its various quality initiatives and programs. The Measures Management System is composed of

a set of business processes and decision criteria that CMS-funded measure developers follow in the creation, implementation, and maintenance of quality measures. Measures developed following the Measures Management System meet the high standards required by the NQF for consensus endorsement. The full Measures Management System set of business processes and decision criteria are documented and described in A Blueprint for the MCS Measures Management System (the Blueprint). Updates to the Blueprint have been made every year since its first release in 2003.

To support the need of eMeasure development, the "Measures Specifications" section was added to Version 8.0 of the Blueprint (August 2011) to guide CMS- contracted measure developers on how to develop and document an eMeasure for either a retooled measure or a de novo measure. The "Measure Specifications" section has since gone through several updates and has been evolved to become the "Measure Lifecycle" section with the latest being published on CMS' website [8].

1.9.6   HITSC Recommended Vocabularies

In 2012, the Health IT Standards Committee (HITSC) Clinical Quality Technology Workgroup and Vocabulary Task Force of the ONC published their recommendations for the use of vocabulary standards by measure developers. The list of QDM categories and their applicable HITSC recommended vocabulary standards are included in the Blueprint's "Measure Lifecycle" section.

2   HQMF Basics

In HQMF, an eMeasure is formatted in XML as a `QualityMeasureDocument` containing metadata (the rest of this section) and terminology (Section 3), a data criteria section (Section 4), and a population criteria section (Section 5). The population criteria section typically contains initial population criteria, denominator criteria, and numerator criteria sub-components, among others. Snippet 1 shows the structure of HQMF.

```
1  <QualityMeasureDocument>
2    <!-- metadata for the measure -- snipped for brevity -->
3    <component>
4      <dataCriteriaSection>
5        <entry>...</entry>
6      </dataCriteriaSection>
7    </component>
8    <component>
9      <populationCriteriaSection>
10       <component>
11         <initialPopulationCriteria>...</initialPopulationCriteria>
12       </component>
13       <component>
14         <denominatorCriteria>...</denominatorCriteria>
15       </component>
16       <component>
17         <numeratorCriteria>...</numeratorCriteria>
18       </component>
19     </populationCriteriaSection>
20   </component>
21 </QualityMeasureDocument>
```

Snippet 1: HQMF document structure - abridged for clarity (from `sample_eMeasure.xml`)

2.1   MetaData

The header of an eMeasure document identifies and classifies the document and provides important metadata about the measure. The Blueprint includes a list of header data elements that are specified by CMS for use by all CMS measure contractors. The Blueprint header requirements have been implemented in the Meaningful Use 2014 eCQMs and their 2015 and 2016 annual updates. This IG further constrains the header in the base HQMF standard by including the Blueprint header requirements. Details are as shown in Volume 3 of this IG package.

The rest of this section describes some of the more important components to the header, such as "Related Documents" (Section 2.2), "Control Variables" (Section 2.3), and "Data Criteria" (Section 4).

2.2   Related Documents

The Clinical Quality Language R1.1 [3] can be used in conjunction with HQMF to construct CQL-based HQMF measures. CQL is a domain specific language used in the Clinical Quality and Clinical Decision Support domains. Measures written in CQL leverage the expressibility and computability of CQL to define the `populationCriteria` used in the HQMF.

Any included CQL library must contain a library declaration line as its first line as in Snippet 2.

```
1   library EXM146 version '4.0.0'
```

Snippet 2: Library declaration line from (`EXM146v4_CQL.cql`)

When using multiple CQL libraries to define a measure, refer to the "Nested Libraries" section of Volume 2 of this guide.

Inclusion of CQL into an HQMF document is accomplished through the use of `relatedDocument` elements. `relatedDocument` elements such as Snippet 3 are incorporated into the HQMF in the metadata section (line: 2 of Snippet 1). CQL expression documents are included by reference using the HQMF `expressionDocument` element as described in *S*4.4 of HQMF [4] – note that *S*4.4.1.5 of HQMF prohibits embedding of expression documents. Snippet 3 shows an example of this.

Lines 21–32 in Snippet 3 identify a CQL expression document (`EXM146v4_CQL.cql`) and assign an internal `root` identifier to it (`22688A59-B73C-4276-9E83-778214E1CA3C`). This identifier is later used when referencing CQL expressions from HQMF population criteria.

> **Conformance Requirement 1 (Referencing CQL Documents):**
> - Measures utilizing CQL libraries **SHALL** include exactly 1 `expressionDocument` element per CQL library refenced in the HQMF.
>   - Libraries implicitly referenced through nesting of libraries **MAY** be included.
>
> - The `expressionDocument` element **SHALL** contain a child `text` element that **SHALL** have a `mediaType` attribute value of `text/cql` and **SHALL** include a child `reference` element whose value contains a URI (relative or absolute) that identifies the CQL expression document.
>
> - Any referenced CQL library **SHALL** contain a library declaration line.
>
> - The library declaration line **SHALL** be the first line in the library.

```
20  <relatedDocument typeCode=COMP>
21     <expressionDocument>
22        <id root="22688A59-B73C-4276-9E83-778214E1CA3C"/>
23        <text mediaType="text/cql">
24           <reference value="EXM146v4_CQL.cql"/>
25           <translation mediaType="application/elm+xml">
26              <reference value="EXM146v4_ELM.xml"/>
27           </translation>
28           <translation mediaType="application/elm+json">
29              <reference value="EXM146v4_ELM.json"/>
30           </translation>
31        </text>
32     </expressionDocument>
33  </relatedDocument>
34  <relatedDocument>
35     <expressionDocument>
36        <id root="22688A59-B73C-4276-9E83-778214E1CA3D"/>
37        <text mediaType="text/cql">
38           <reference value="Common-2.0.0_CQL.cql"/>
39           <translation mediaType="application/elm+xml">
40              <reference value="Common-2.0.0_ELM.xml"/>
41           </translation>
42           <translation mediaType="application/elm+json">
43              <reference value="Common-2.0.0_ELM.json"/>
44           </translation>
45        </text>
46     </expressionDocument>
47  </relatedDocument>
```

Snippet 3: Referencing a CQL file in HQMF (from `EXM146v4_eMeasure.xml`)

Inclusion of CQL libraries within the HQMF framework must conform to Conformance Requirement 1.

2.2.1 Including ELM

CQL defines both a human-readable text representation and a machine-oriented XML representation called the Expression Logical Model (ELM). The human-readable text representation is optimized for authoring while the ELM XML representation offers a canonical, simplified representation that is easier to implement in software. Any CQL expression can be directly translated to its ELM equivalent. It is expected that authoring tools will perform the translation from CQL to ELM and that measure authors will never directly work with ELM.

Both CQL and ELM representations should be referenced from the HQMF to follow the HL7 V3 approach of supporting human readability at a minimum (in this case, the high-level CQL syntax) and a canonical representation for machine processing (in this case, CQL's Expression Logical Model (ELM)). This approach supports easy human review of measure logic via CQL and easy implementation of that logic in tools via ELM.

> **Conformance Requirement 2 (Referencing ELM Documents):**
> - Any `expressionDocument/text` elements that reference a CQL document **SHOULD** include a `translation` element that includes a child `reference` element whose value contains a URI (relative or absolute) that identifies an Expression Logical Model (ELM) expression document.
>
>   – If an ELM translation is provided, both an XML and JSON representation of the ELM **SHOULD** be included.
>
>   * The XML representation of the ELM **SHALL** have a `mediaType` attribute value of `application/elm+xml`
>   * The JSON representation of the ELM **SHALL** have a `mediaType` attribute value of `application/elm+json`
>
> - Any `translation`-referenced ELM documents **SHALL** be semantically equivalent to the corresponding parent CQL expression document.

Lines 25–30 of Snippet 3 shows an example of how an HQMF document would reference both the CQL and the ELM. More on ELM can be found in Chapter 4.1. For examples of ELM using the XML and JSON representations please see the included examples, `EXM146v4_ELM.xml` and `EXM146v4_ELM.json`.

2.3 Control Variables

HQMF introduces the concept of control variables as metadata that influences the computation of measures. Currently HQMF defines only one control variable: the measurement period, see *§4.1.3.12 of HQMF [4]*. Snippet 4 demonstrates how to indicate a `controlVariable` in the HQMF (line: 2 of Snippet 1).

The values of HQMF control variables are accessible to CQL libraries as CQL parameters. The HQMF measurement period control variable is accessible to CQL libraries as the value of a parameter called `"Measurement Period"`. Snippet 5 shows an example of a CQL library declaring this parameter.

```
48  <controlVariable>
49     <measurePeriod>
50        <code code="MSRTP" codeSystem="2.16.840.1.113883.3.560">
51           <displayName value="Measurement Period"/>
52        </code>
53        <value xsi:type="PIVL_TS">
54           <phase lowClosed="true" highClosed="true">
55           <low value="201201010000"/>
56              <high value="201212312359"/>
57              <width xsi:type="PQ" value="1" unit="a"/>
58           </phase>
59        </value>
60     </measurePeriod>
61  </controlVariable>
```

Snippet 4: HQMF representation of control variables from (`EXM146v4 eMeasure.xml`)

```
20  parameter "Measurement Period" Interval<DateTime>
```

Snippet 5: CQL declaration of the measurement period parameter (from `EXM146v4 CQL.cql`)

> **Conformance Requirement 3 (HQMF Measurement Period):**
> - The value of the HQMF measurement period control variable **SHALL** be made available to CQL libraries as the value of the `measurePeriod` element.
>
> - CQL libraries that require access to the HQMF measurement period control variable **SHALL** either declare the type of the `"Measurement Period"` parameter as an interval of **DateTime** or provide a default value as an interval of **DateTime**.

3   Terminology

This chapter describes how to use codes and valuesets from codesystems like LOINC, SNOMED-CT, and others within the CQL and HQMF files of a measure package.

Valuesets and direct referenced codes are declared in the header section of the CQL using the CQL `valueset` and `code` constructs. In the case of direct referenced codes a `codesystem` declaration must precede the code declaration (per CQL v1.2 specification). Examples of valueset and code declarations can be seen in the accompanying "examples/TerminologyExample.cql".

```
5  codesystem "SNOMED-CT": 'urn:oid:2.16.840.1.113883.6.96'
6     version 'urn:hl7:version:201609'
7  valueset "Encounter Inpatient SNOMEDCT Value Set":
8     'urn:oid:2.16.840.1.113883.3.666.7.307' version 'urn:hl7:version:20160929'
9  code "Venous foot pump, device (physical object)": '442023007' from "SNOMED-CT"
```

Snippet 6: CQL declaration of codesystem, valueset, and code (from `Terminology CQL.cql`)

Further discussion of **codesystem**, **valueset**, and **code** can be found in Volume 2 of this IG, sections 2.3, 2.4, and 2.5.

All declared valuesets and codes can be found in the definition elements in the HQMF (line: 2 of Snippet 1). Examples of `valueSet` and `cql-ext:code` definitions can be found in the accompanying "examples/TerminologyExample.xml".

```
58  <defintion typeCode="DRIV">
59      <valueSet classCode="OBS" moodCode="DEF">
60          <!-- id of the value set from the CQL library with the URI prefixes removed -->
61          <id root="2.16.840.1.113883.3.666.5.307"/>
62          <!—local identifier of the ValueSet from the CQL library -->
63          <title value="Encounter Inpatient SNOMEDCT Value Set"/>
64          <!-- Version attribute from the CQL valueset definition, without URI prefixes -->
65          <cql-ext:version value="20160929"/>
66      </valueSet>
67  </definition>
68  <definition>
69      <cql-ext:code code="442023007"
70      codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED-CT" codeSystemVersion="201609">
71          <!—local identifier of the Code from the CQL library -->
72          <displayName value="Venous foot pump, device (physical object)"/>
73      </cql-ext:code>
74  </definition>
```

Snippet 7: Example HQMF terminology definitions (from `Terminology_eMeasure.xml`)

Measures using valueset and/or direct referenced codes must conform to Conformance Requirement 4.

Conformance Requirement 4 (Terminology Inclusion):
- All valuesets and codes referenced in the CQL **SHALL** be included in the HQMF using definition elements.

- `valueset.id@root` **SHALL** be the OID of the **valueset**

- `valueset.title@value` **SHALL** be the CQL identifier for the **valueset**

- `cql-ext:code@codeSystemName` **SHALL** be the CQL identifier for the **codesystem**

- `cql-ext:code.displayName@value` **SHALL** be the CQL identifier for the **code**

---

4   Data Criteria

The data criteria section (lines 4–6 of Snippet 1) 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. Snippet 8 shows an example of a data criteria entry indicating a "Laboratory Test, Performed".

Volume 3 of this implementation guide contains templates for the `entry`'s in the data criteria section (as in Snippet 8) and the mappings from the HQMF templates to their respective QRDA templates. For an example of how to include direct referenced codes in the data criteria section please see line 110 of the accompanying `examples/Terminology_eMeasure.xml`.

```
351  <entry typeCode="DRIV">
352     <observationCriteria classCode="OBS" moodCode="EVN">
353        <templateId>
354           <item root="2.16.840.1.113883.10.20.28.3.42" extension="2016-12-01"/>
355        </templateId>
356        <id root="05f0a7c8-371e-4e77-aad4-22ff9fdf7fa3"/>
357        <code valueset="2.16.840.1.113883.3.464.1003.198.12.1012"
358           valueSetVersion="20160929"/>
359        <title value="Laboratory Test, Performed"/>
360        <statusCode code="completed"/>
361     </observationCriteria>
362  </entry>
```

Snippet 8: Example HQMF data criteria (from `EXM146v4_eMeasure.xml`)

Conformance Requirement 5 (Data Criteria Inclusion):
- Data criteria entries **SHALL** be included in the HQMF for each QDM element referenced by the measure logic.

- Data criteria entries **SHALL** conform to the templates defined in Volume 3 of this IG.

- Data criteria entries **SHALL NOT** include `excerpt` or `temporallyRelatedInformation` elements.

Note that CQL defines its own method for referencing data and that there is no direct link between the data criteria included in the HQMF and the data used by the CQL expressions. The HQMF data criteria are retained by this implementation guide to promote limited backwards compatibility with existing implementations of the QDM-based HQMF IG for the following use cases:

- Determining the set of data used by a particular eMeasure.

- Limited "scoop-and-filter" for creation of QRDA category 1 reports. The elimination of temporal relationships from HQMF data criteria may result in the inclusion of more data than actually required. Implementations desiring or required to comply with privacy policies that mandate or recommend fine-grained filtering should examine the CQL or ELM to determine additional data constraints nec- essary for adherence to those policies.

Section 4.1 describes a means for deriving HQMF data criteria from CQL data references.

4.1   Use of ELM

The canonical representation of ELM makes it straightforward to derive HQMF data criteria for CQL data references to comply with Conformance Requirement 6:

(a) ELM elements with an `xsi:type` of `Retrieve` are equivalent to the simplified HQMF data criteria defined in Chapter 2

(b) The value of those ELM element's `dataType` attributes can be mapped to the corresponding QDM data type for that data reference

(c) The value of those ELM element's `codes` child elements identify the value set for the concept for that data reference

(d) The corresponding HQMF data criteria template can be looked up in Volume 3 of this IG using the QDM data type identified in item (b) above

(e) For each ELM element identified in item (a) above, an HQMF data criteria should be included using the template identified in item (d) that references the value set identified in item (c)

To illustrate the mapping, Snippet 9 shows an ELM data reference and Snippet 10 shows the corresponding HQMF data criteria.

```
20   <def name="Acute Pharyngitis" id="2.16.840.1.113883.3.464.1003.102.12.1011"
        accessLevel="Public"/>

65   <operand xmlns:ns2="urn:healthit-gov:qdm:v5_0_2" dataType="ns2:Diagnosis"
        xsi:type="Retrieve">
66     <codes name="Acute Pharyngitis" xsi:type="ValueSetRef"/>
67   </operand>
```

Snippet 9: ELM data reference for Diagnosis: Acute Pharyngitis (from `EXM146v4_ELM.xml`)

```
363  <entry typeCode="DRIV">
364    <observationCriteria classCode="OBS" moodCode="EVN">
365      <templateId>
366        <item root="2.16.840.1.113883.10.20.28.3.1" extension="2016-12-01"/>
367      </templateId>
368      <id root="9e4e810d-3c3e-461e-86f8-6fe7a0b1ca2b"/>
369      <code code="282291009" codeSystem="2.16.840.1.113883.6.96"
370        codeSystemName="SNOMED CT">
371        <displayName value="Diagnosis"/>
372      </code>
373      <title value="Diagnosis"/>
374      <statusCode code="completed"/>
375      <value xsi:type="CD" valueSet="2.16.840.1.113883.3.464.1003.102.12.1011"
376        valueSetVersion="20160929"
377    </observationCriteria>
378  </entry>
```

Snippet 10: HQMF data criteria for Diagnosis: Acute Pharyngitis (from `EXM146v4_eMeasure.xml`)

Note: while the QDM datatypes templates in volume 3 contain specifications of QDM attributes, they are included for the purpose of formal semantic representations of the QDM datatypes. The `dataCriteriaSection entry` (and by extension the HQMF) shall not contain any references to the attributes within the measure's HQMF.

> **Conformance Requirement 6 (Data Criteria Inclusion):**
> - `dataCriteriaSection entry`'s **SHALL** only contain information related to data types
> - `dataCriteriaSection entry`'s **SHALL NOT** contain information related to data type attributes

5 Population Criteria

The `populationCriteriaSection` (lines 9 - 19 of Snippet 1) includes definitions of criteria used to specify populations. The criteria specifying these populations are described using CQL and those CQL expressions are given context in the HQMF. In this section we describe how to use CQL and HQMF to define population criteria.

CQL provides the logical expression language that is used to define population criteria. CQL-based constraints are used instead of the RIM-based data criteria constraints and population expressions defined in §4.14.3 and §2.3.2 of the HQMF R1 STU 2 specification [4] respectively.

> **Conformance Requirement 7 (Population Criteria Restrictions):**
> Population criteria **SHALL NOT** include `allTrue`, `allFalse`, `atLeastOneTrue`, `atLeastOneFalse`, `onlyOneTrue` or `onlyOneFalse` elements.

Once included in the HQMF file, expressions defined in the CQL can be used to further refine the data criteria and to define population criteria. Figure 3 illustrates the general concept. Figure 3 illustrates the
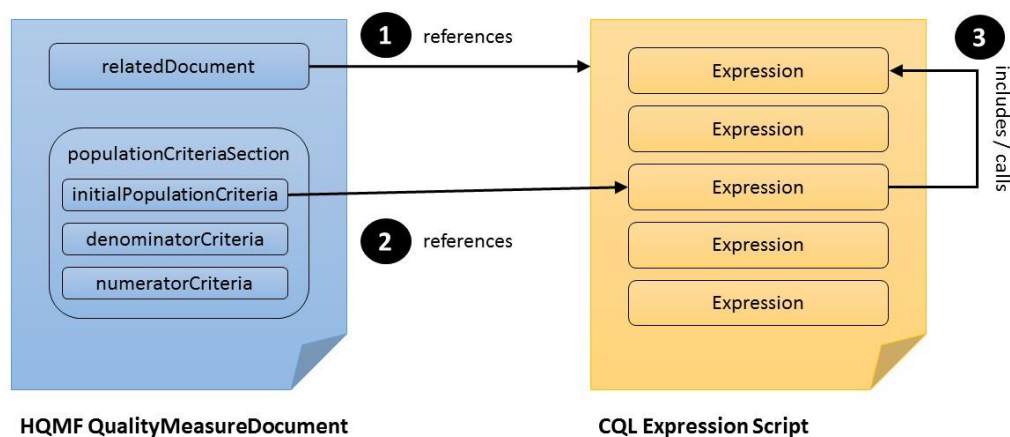
Figure 3: HQMF with linked expression document

relationship between the HQMF and CQL documents: The HQMF `QualityMeasureDocument` references a CQL expression script (#1), the population criteria sections reference a particular expression from the referenced CQL file (#2), the referenced expression in turn may include or call another expression (#3) in the same (or a different) CQL expression script. Snippet 11 and Snippet 12 demonstrate the use of HQMF and CQL in the definition of the `"Initial Population"`. Note that the `root` identifier of the criteria reference (line 424) matches the internal identifier assigned to the CQL expression document (line 22 of Snippet 3). Also note the use of the `EXM146v4` namespace and escaped quotation marks (""") in line 425.

Snippet 12 shows several examples of a CQL expression calling another, e.g. the `"Initial Population"` expression references another CQL expression: `"Pharyngitis Encounters With Antibiotics"`. In this example the referenced expressions are all contained within the same CQL file (`EXM146v4_CQL.cql`)

```
418  <initialPopulationCriteria classCode="OBS" moodCode="EVN" isCriterionInd="true">
419    <code codeSystem="2.16.840.1.113883.5.4" codeSystemName="HL7 Act Code" code="IPOP">
420      <displayName value="Initial Population"/>
421    </code>
422    <precondition typeCode="PRCN">
423      <criteriaReference moodCode="EVN" classCode="OBS">
424      <id root="22688A59-B73C-4276-9E83-778214E1CA3C"
425          extension="EXM146v4.&quot;Initial Population&quot;"/>
426      </criteriaReference>
427    </precondition>
428  </initialPopulationCriteria>
```

Snippet 11: Defining a population via reference to a CQL expression (from `EXM146v4 eMeasure.xml`)

```
 1  library EXM146 version '4.0.0'

10  using QDM version '5.02'
11
12  include Common version '2.0.0' called Common

26  define "In Demographic":
27    AgeInYearsAt(start of Measurement Period) >= 2
28      and AgeInYearsAt(start of Measurement Period) < 18

36  define "Measurement Period Encounters":
37    ["Encounter, Performed": "Ambulatory/ED Visit"] Encounter
38    where Encounter.relevantPeriod during Measurement Period
39      and "In Demographic"
40
41  define "Pharyngitis Encounters With Anitbiotics":
42    "Measurement Period Encounters" Encounters
43      with "Pharyngitis" Pharyngitis such that
44        Common."Includes Or Starts During"(Pharyngitis, Encounters)
45      with "Antibiotics" Antibiotics such that Antibiotics.authorDatetime
46        3 days or less after start of Encounters.relevantPeriod
47
48  define "Initial Population":
49    "Pharyngitis Encounters With Antibiotics"
```

Snippet 12: CQL definition of the `"Initial Population"` criteria (from `EXM146v4 CQL.cql`)

and some are included above. The `"In Demographic"` expression uses the built-in CQL function `Age-InYearsAt()`. The definition of `"Pharyngitis Encounters With Antibiotics"` includes the function `"Includes Or Starts During"`, defined in another CQL library (`Common` as described in `examples/EXM146v4/Common-2.0.0 CQL.cql`), further explanation of nested libraries is given in the "Nested Libraries" section of the Volume 2 of this IG.

> **Conformance Requirement 8 (Referential Integrity):**
> All HQMF `populationCriteriaSection` component's
>
> - **SHALL** reference exactly one CQL expression.
> - **SHALL** reference the same CQL library.

5.1 Criteria Names

To encourage consistency among measures, the following guidelines for specifying population criteria within a measure are proposed. The measure population criteria names and calculation methods used here are based on the Health Quality Measures Format (HQMF) HL7 standard [4].

> **Conformance Requirement 9 (Criteria Names):**
> The name of an expression specifying a population criteria within a measure **SHOULD** always be the name of the criteria type[††]:
>
> - `"Initial Population"`
> - `"Denominator"`
> - `"Denominator Exclusion"`
> - `"Denominator Exception"`
> - `"Numerator"`
> - `"Numerator Exclusion"`
> - `"Measure Population"`
> - `"Measure Population Exclusion"`
> - `"Measure Observation"`[†]
> - `"Stratification"`

For each type of measure, the set of applicable criteria are defined by the Health Quality Measure Format (HQMF) specification. In addition, the formula for calculating the measure score is implied by the type of the measure. The following sections describe the expected result type for population criteria for each type of measure, as well as explicitly defining the measure score calculation formula.

In addition to the measure type, measures generally fall into two categories, patient-based, and non-patient-based, such as episode-of-care-based. In general, patient-based measures count the number of patients in each population, while non-patient-based measures count the number of items (such as encounters) in each population. Although the calculation formulas are conceptually the same for both categories, for ease of expression, population criteria for patient-based measures return true or false, while non-patient-based measures return the item to be counted such as an encounter or procedure.

5.2 HQMF Population Semantics

HQMF defines a set of measure population components that are used to construct measures. HQMF populations have implicit relationships to each other as defined in the HQMF specification. For example, for proportion measures, `denominatorCriteria` have an implicit dependency on `initialPopulation-Criteria`, i.e. the criteria for inclusion in the denominator of a measure implicitly include the criteria for inclusion in the initial population. Similarly, `numeratorCriteria` have an implicit dependency on

---

[†] This is the name of a function. See the Continuous Variable Measure section for more.

[††] When using multiple populations and/or multiple population groups, see Section 5.7

[‡‡] Some ratio measures will require multiple Initial Populations, one for the numerator and one for the denominator.

Table 2: Measure populations based on types of measure scoring.

	Initial Population	Denominator	Denominator Exclusion	Denominator Exception	Numerator	Numerator Exclusion	Measure Population	Measure Population Exclusion
Proportion	R	R	O	O	R	O	NP	NP
Ratio	R‡‡	R	O	NP	R	O	NP	NP
Continuous Variable	R	NP	NP	NP	NP	NP	R	O
Cohort	R	NP	NP	NP	NP	NP	NP	NP

R=Required. O=Optional. NP=Not Permitted.

`denominatorCriteria`, i.e. the criteria for inclusion in the numerator of a measure implicitly include the criteria for inclusion in the denominator.

CQL expressions referenced by HQMF population criteria are executed within the context of these implicit dependencies.

> **Conformance Requirement 10 (HQMF Population Semantics):**
> - CQL expressions referenced by an HQMF population component **SHALL**
>   - be executed within the context of the implicit HQMF population component dependencies.
>   - CQL expressions **MAY** include explicit dependencies that duplicate the implicit HQMF population dependencies.

For example, Snippet 13 defines the `"Initial Population"` and `"Denominator"` for a measure.

```
define "Initial Population":
  "In Demographic" and "Has Target Encounter"

define "Denominator":
  "Initial Population"
```

Snippet 13: Explicit definition of the initial population and denominator.

In this snippet the relationship between the `"Denominator"` and the `"Initial Population"` is made explicit even though the HQMF defines the `"Denominator"` to be a subset of the `"Initial Population"`. With respect to the HQMF population definitions, the following CQL code has identical meaning: `define "Denominator": true`. This last bit of code defines the `"Denominator"` utilizing the HQMF dependencies but this dependency is not obvious from the CQL; this is called an implicit dependency.

5.3 Proportion Measures

An HQMF document representing a proportion measure will include one or more population criteria sections as described in Table 2.

The semantics of these components are unchanged from the HQMF specification; the only difference is that each component references a single criterion encoded as a CQL expression.

The referenced CQL expressions return either an indication that a patient meets the population criteria (patient-based measures) or the events that a particular patient contributes to the population (episode-of-care-based measures). For example, consider two measures:

Measure #	Denominator	Numerator
1	All patients with condition A that had one or more encounters during the measurement period.	All patients with condition A that underwent procedure B during the measurement period.
2	All encounters for patients with condition A during the measurement period.	All encounters for patients with condition A during the measurement period where procedure B was performed during the encounter.

Table 3: Patient-based and Episode-of-Care Measure Examples

In Table 3, 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 encounters to the denominator and numerator counts.

For measures conforming to this implementation guide, the HQMF ITMCNT measure attribute is not used to identify the items to count. Instead, CQL expressions should be written to return an appropriate value for each population depending on the measure type. Note that the CQL context indicates whether a given expression is executed in the context of a single patient or a population of patients. Therefore the Patient context is used for both patient and episode-of-care measures and it is the responsibility of the referenced expression to return an appropriate value.

> **Conformance Requirement 11 (Proportion Measures):**
> - Population criteria **SHALL** each reference a single CQL expression.
> - The CQL expression **SHALL** use the Patient context and be executed within the context of a single patient record at a time.
> - The CQL expression for patient-based measures **SHALL** return a **Boolean** to indicate whether a patient matches the population criteria (**true**) or not (**false**).
> - The CQL expression for episode-of-care measures **SHALL** return a **List** of events that match the population criteria.

5.3.1 Proportion measure scoring

Additional information on how proportion measures are scored (and the semantics behind the criteria names) can be found in the HQMF specification [4].

5.4   Ratio Measures

An HQMF document representing a ratio measure will include one or more population criteria sections as described in Table 2.

In addition, it may also include one measure observation section with one or more `measureObservation-Definition` elements. The semantics of these components are unchanged from the HQMF specification; the only difference is that each measure population component and each measure observation definition references a single criterion encoded as a CQL expression.

> **Conformance Requirement 12 (Ratio Measures):**
> - Population criteria components SHALL each reference a single CQL expression as defined by Conformance Requirement 11
> - `measureObservationDefinition` elements SHALL reference CQL expressions as defined by Conformance Requirement 13.

For patient-based ratio measures, all population criteria must return true or false (or null). For non-patient-based ratio measures, each population criteria must return the same type, such as an Encounter, or Procedure.

For ratio measures that include a Measure Observation, the measure observation is specified in the same way as it is for continuous variable measures. In particular, the Measure Observation is defined as a function that takes a single argument of the same type as the elements returned by all the population criteria, and the aggregation method is specified in the HQMF.

5.4.1   Ratio measure scoring

Additional information on how proportion measures are scored (and the semantics behind the criteria names) can be found in the HQMF specification [4].

5.5   Continuous Variable Measure

An HQMF document representing a continuous variable measure will include one or more population criteria sections as described in Table 2.

In addition, it will also include one `measureObservationSection` (seperate from the `population-CriteriaSection`, often placed between lines 20 and 21 of Snippet 1) with one or more `measure-ObservationDefinition` elements. The semantics of these components are unchanged from the HQMF specification; the only difference is that each measure population component and each measure observation definition references a single criterion encoded as a CQL expression. Note that the implicit population semantics described in Section 5.2 apply equally to continuous variable measures: measure observations are only computed for patients matching the appropriate set of population criteria.

An example `measureObservationDefinition` element is shown in Snippet 14.

The CQL expression referenced from the `measureObservationDefinition` component returns a list of events for each patient over which the calculation will be performed, an example is shown in Snippet 15.

```
481  <measureObservationSection>
482      <templateId>
483          <item root="2.16.840.1.113883.10.20.28.2.4"/>
484      </templateId>
485      <id extension="MeasureObservations" root="B525A408-F6C1-4755-97CF-E08346F3751E"/>
486      <code code="57027-5" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
487      <title value="Measure Observation Section"/>
488      <text/>
489      <!--Definition for Measure Observation 1-->
490      <definition>
491          <measureObservationDefinition classCode="OBS" moodCode="DEF">
492              <id extension="Measure Observation"
493                  root="8A9A47CF-45A4-4385-923C-5A045D8EA9F8"/>
494              <code code="AGGREGATE" codeSystem="2.16.840.1.113883.5.4"/>
495              <value nullFlavor="DER" xsi:type="INT">
496                  <expression value="TestCMS55v5.&quot;Measure Observation&quot;"/>
497              </value>
498              <component>
499                  <criteriaReference>
500                      <id extension="TestCMS55v5.&quot;Measure Population&quot;"
501                          root="DFAAF6C1-0609-49C7-BCEA-8EEDFB65DCFF"/>
502                  </criteriaReference>
503              </component>
504          </measureObservationDefinition>
505      </definition>
506  </measureObservationSection>
```

Snippet 14: Sample measure observation section from `TestCMS55v5_eMeasure.xml`

```
34  define "Measure Population":
35      "Initial Population"

60  // Measure Observation
61  define function "Related ED Visit"(Encounter "Encounter, Performed" ):
62      Last(["Encounter, Performed": "Emergency Department Visit"] ED
63        where ED.relevantPeriod ends 1 hour or less before start of Encounter.relevantPeriod
64        sort by start of  ED.relevantPeriod )
65
66  define function "Measure Observation" (Encounter "Encounter, Performed" ):
67      duration in minutes of "Related ED Visit"(Encounter).locationPeriod
```

Snippet 15: Sample CQL (from `TestCMS55v5_CQL.cql`) for a continuous-variable measure

In the example shown in Snippet 14 and Snippet 15: the measure reports the `AGGREGATE` (line 494 in the HQMF) of the result of executing the `"Measure Observation"` function (line 496 in the HQMF, line 66 in the CQL) on each of the events returned by the `"Measure Population"` expression (line 501 in the HQMF, line 34 in the CQL).

Conformance Requirement 13 (Continuous Variable Measures):

- `initialPopulationCriteria`, `measurePopulationCriteria` and `measurePopulationExclusionCriteria` **SHALL** each reference a single CQL expression as defined by Conformance Requirement 11.

- CQL expressions referenced from `measureObservationDefinition` elements **SHALL** use
    - `Patient` context and be executed within the context of a single patient.

- The CQL expression referenced from the `component` of the `measureObservationDefinition`
    - **SHALL** return a list of events over which the value will be computed.

- CQL functions referenced from the `value` of the `measureObservationDefinition` **SHALL**:

    - be in the same CQL file as the CQL expression referenced from the `component` of the `measureObservationDefinition` since the `value` element does not allow specification of the document ID

    - accept a single argument whose type matches the elements of the list returned by the CQL expression referenced from the `component` of the `measureObservationDefinition`

    - return either an **Integer**, a **Decimal**, or a **Quantity**

For continuous variable measures, the measure observation is defined as a function that takes a single parameter of the type of elements returned by the population criteria. Each population criteria other than the measure observation must return the same type.

5.5.1 Continuous variable measure scoring

Additional information on how proportion measures are scored (and the semantics behind the criteria names) can be found in the HQMF specification [4].

5.6 Cohort Definitions

For cohort definitions, only the Initial Population criteria type is used. For patient-based cohort definitions, the criteria should return a **true** or **false** (or **null**). For other types of cohort definitions, the criteria may return any type.

5.7 Measures with Multiple Populations

When a measure has multiple population groups (multiple `populationCriteriaSection`'s), the criteria names will follow the convention above, adding the number of the population group to each criteria, e.g. `"Initial Population 1"`, `"Denominator 1"`, etc. Note that when multiple population groups are present, the number of the group is added to all population groups, not just the groups other than the first.

For multiple population ratio measures that specify 2 initial populations, the populations would be named with an additional `"_X"` to distinguish the initial populations, e.g. `"Initial Population 1_1"`, `"Initial Population 1_2"`, `"Initial Population 2_1"`, `"Initial Population 2_2"`.

> **Conformance Requirement 14 (Multiple Population Indexing):**
> When specifying multiple populations and/or multiple population groups the following naming scheme **SHOULD** be used
>
> - (*Criteria Name*) (*population group number*)(*population number*)

Note when a measure has a single population group but multiple populations (such as a ratio measure), the "_" is dropped. For example, `"Initial Population 1"`, `"Initial Population 2"` refer to the populations NOT population groups.

Note also that when a measure has multiple population groups, the expectation is that the measure would have multiple scores, one for each population group. The formulas for calculation of the groups do not change, they are the same as for single group measures, just calculated using the criteria for each group.

5.8 Stratification

> **Conformance Requirement 15 (Stratification Criteria):**
> - Stratifier criteria **SHALL NOT** include HQMF logical operators.
> - Each stratifier criteria child `precondition` **SHALL** include one `criteriaReference` element referencing a single CQL expression.
> - The CQL expression **SHALL** return a **Boolean**.

Stratification is represented using a `stratifierCriteria` component. The semantics of this component are unchanged from the HQMF specification; the only difference is that each child `criteriaReference` references a CQL expression that returns a boolean to determine whether a given patient meets the criteria for that stratification. Snippet 16 shows an example stratifier that stratifies results for two sub-populations. Snippet 17 shows the CQL representation of the stratifier.

```
373  <stratifierCriteria>
374      <id extension="Stratifiers" root="F8EB3BCE-C313-49F0-B441-83F9B060FBEC"/>
375      <code code="STRAT" codeSystem="2.16.840.1.113883.5.4" codeSystemName="Act Code"/>
376      <precondition typeCode="PRCN">
377          <criteriaReference classCode="OBS" moodCode="EVN">
378              <id extension="TestCMS55v5.&quot;Stratification 1&quot;"
379                  root="DFAAF6C1-0609-49C7-BCEA-8EEDFB65DCFF"/>
380          </criteriaReference>
381      </precondition>
382  </stratifierCriteria>
```

Snippet 16: Example Stratifier from `TestCMS55v5_eMeasure.xml`

```
48  define "Stratification 1" :
49     "Inpatient Encounter" Encounter
50         where not (Encounter.principalDiagnosis in "Psychiatric/Mental Health Patient")
```

Snippet 17: Example Stratifier from `TestCMS55v5_CQL.cql`

5.9 Supplemental Data Elements

Conformance Requirement 16 (Supplemental Data Elements):
Each supplemental data element referrenced in the CQL **SHOULD** :

- return a single value when evaluated in the context of a member of the population

- have a name begining with `"SDE"`

Part of the definition of a quality measure involves the ability to specify additional information to be returned for each member of a population. Within QDM and HQMF, these supplemental data elements are specified as QDM Data Elements for patient characteristics (such as Race, Ethnicity, Payer, and Administrative Sex) and marking them with an SDE code within the underlying HQMF XML. Snippet 18 demonstrates an example supplemental data definition using the `stratifierCriteria`.

```
410  <cql-ext:supplementalDataElement>
411     <id extension="Risk Adjustment Variables"
412        root="E738B53D-3537-41C3-A24F-507853D0C905"/>
413     <code code="MSRADJ" codeSystem="2.16.840.1.113883.5.4" codeSystemName="Act Code"/>
414     <precondition typeCode="PRCN">
415        <criteriaReference classCode="OBS" moodCode="EVN">
416           <id extension="TestRiskAdj.&quot;Hepatic Failure&quot;"
417              root="FAB9DAAA-36D9-4674-8C63-6A3CB38D6BCC"/>
418        </criteriaReference>
419     </precondition>
420  </cql-ext:supplementalDataElement>
```

Snippet 18: Sample Supplemental Data Elements from `TestRiskAdj_eMeasure.xml`

```
17  define "SDE Ethnicity":
18     ["Patient Characteristic Ethnicity": "Ethinicity"]
```

Snippet 19: Example Supplemental Data Element from `TestCMS55v5_CQL.cql`

With CQL, supplemental data elements are specified using the same mechanism as any other population criteria, by defining an expression that returns the appropriate data element, and then identifying that expression within the CQL-Based HQMF. Examples of the HQMF and CQL are given in Snippet 18 and Snippet 19, respectively.

By convention, the name of each supplemental data element expression would start with `"SDE"`. The supplemental data element expressions would be required to return a single value when evaluated in the context of a member of the population. For example, patient-based measures would return the value of a supplemental data element for a given patient.

September 2017

5.10 Risk Adjustment

> **Conformance Requirement 17 (Risk Adjustment Criteria):**
> - Risk Adjustment Variables **SHALL** be included using `cql-ext:supplementalDataElement` elements as defined in `hhs-cql-hqmfn1-ext-v1.xsd`
> - Risk Adjustment Variables **SHALL** reference a single CQL expression.

Some measure may define variables used to adjust scores based on a measure of "risk" observed in the population. Such variables are referred to as risk adjustment variables. Risk adjustment variables are included in the `populationCriteriaSection` (lines 9 - 19 of Snippet 1) and defined using CQL, such inclusing must adhere to Conformance Requirement 17.

```
497  <cql-ext:supplementalDataElement>
498      <id extension="Risk Adjustment Variables"
499          root="E738B53D-3537-41C3-A24F-507853D0C905"/>
500      <code code="MSRADJ" codeSystem="2.16.840.1.113883.5.4" codeSystemName="Act Code"/>
501      <precondition typeCode="PRCN">
502          <criteriaReference classCode="OBS" moodCode="EVN">
503              <id extension="TestRiskAdj.&quot;Hepatic Failure&quot;"
504                  root="FAB9DAAA-36D9-4674-8C63-6A3CB38D6BCC"/>
505          </criteriaReference>
506      </precondition>
507  </cql-ext:supplementalDataElement>
```

Snippet 20: Sample Risk Adjustment Variable from `TestRiskAdj_eMeasure.xml`

```
43  define "Hepatic Failure":
44      exists ("Cirrhosis Dx")
45      and exists ("Bilirubin Test")
46      and exists ("Serum Albumin Test")
```

Snippet 21: Sample Risk Adjustment Variable from `TestRiskAdj_CQL.cql`

An example of risk adjustment can be found in the included directory `examples/TestRiskAdj_v5_1/`, the relevant sections of the HQMF (Snippet 20) and CQL (Snippet 21) have been included.

6 Composite Measure Development

Composite measures make use of multiple component measures to produce a combined score. In the most general case, a composite measure is akin to a continuous variable measure, where the measure observation for each population member is some combination of their component measure scores. However, the calculation logic involved is tedious, and a higher-level representation of the most common composite measure calculation approaches would be much simpler to work with and understand.

Based on an analysis of the example composite measures so far, the following calculation methods have been identified.

6.1   Opportunity Scoring

Opportunity scoring simply combines the scores from composite measures by combining the numerators and denominators for each component. In this approach, the measure score uses the same calculation logic as any proportion measure would, but the individual populations are the union of the component measure populations.

6.2   Two-Step Calculation Methods

The following approaches all involve a two step process where the first step involves computing the score for each patient using some formula involving component measure population criteria, and the second step involves aggregating these scores in the same way that a continuous variable measure is defined.

Note that these methods may be applied at any cross-section of the population (i.e. by Provider, by Department, by Institution, etc.)

All-or-nothing Scoring

All-or-nothing scoring includes an individual in the numerator of the composite measure if they are in the numerators of all of the component measures in which they are in the denominator.

- Denominator of composite is the union of component denominators

- Numerator of composite:

  - is 1 if all the component numerators are 1
  - is 0 otherwise

An example of an "All-or-nothing" scored composite measure has been in included in `examples/TestComposite/`. This directory contains "the composite measure", `Composite_eMeasure.xml`, and "the component measures" in directories `Test122v5 Artifacts/` and `Test131v5 Artifacts/`. From `Composite eMeasure.xml`, note the component measures are referenced in using `relatedDocument` elements. Within the metadata of the measure, a `subjectOf` element contains the details of the `"Composite Measure Scoring"` as shown in Snippet 22.

```
101   <subjectOf>
102      <measureAttribute>
103         <code code="CMPMSRSCR" codeSystem="2.16.840.1.113883.5.4">
104            <displayName value="Composite Measure Scoring"/>
105         </code>
106         <value xsi:type="CD" code="ALLORNONESCR"
107            codeSystem="2.16.840.1.11.20553">
108            <displayName value="All-or-nothing Scoring"/>
109         </value>
110      </measureAttribute>
111   </subjectOf>
```

Snippet 22: Example All-or-nothing scored composite measure from `Composite_eMeasure.xml`

Weighted Scoring

Weighted scoring gives and individual a score based on a weighted factor for each component numerator in which they appear.

- Denominator of composite is the union of component denominators

- Each component numerator is assigned some percentage weight and the score is the sum of the weights.

A "Weighted" scored composite measure would contain a `subjectOf` declaration indicating the measure scoring (see Snippet 22). Additionally, a composite measure using a weighted scoring scheme would need to include the weight of each measure within the `relatedDocument` elements, as in line 6 of Snippet 23:

```
 1  <relatedDocument typeCode="XCRPT">
 2     <componentQualityMeasureDocument>
 3        <id root="40280381-537c-f767-0153-c378bd7207a5"/>
 4        <setId root="9a031e24-3d9b-11e1-8634-00237d5bf174"/>
 5        <versionNumber value="1.1"/>
 6        <subjectOf>
 7           <measureAttribute>
 8              <code code="CMPMSRSCRWGHT" codeSystem="2.16.840.1.113883.5.4"/>
 9              <value xsi:type="PQ" value="0.2"/>
10           </measureAttribute>
11        </subjectOf>
12     </componentQualityMeasureDocument>
13  </relatedDocument>
14  <relatedDocument typeCode="XCRPT">
15     <componentQualityMeasureDocument>
16        <id root="40280381-51f0-825b-0152-22bd8ee41875"/>
17        <setId root="500e4792-7f94-4e34-8546-ee71c56fe463"/>
18        <versionNumber value="1.1"/>
19        <subjectOf>
20           <measureAttribute>
21              <code code="CMPMSRSCRWGHT" codeSystem="2.16.840.1.113883.5.4"/>
22              <value xsi:type="PQ" value="0.8"/>
23           </measureAttibute>
24        </subjectOf>
25     </componentQualityMeasureDocument>
26  </relatedDocument>
```

Snippet 23: Composite measure `relatedDocument`'s

References

[1] *Crossing the Quality Chasm: A New Health System for the 21st Century*. Institute of Medicine, March 2001. http://www.nationalacademies.org/hmd/Reports/2001/Crossing-the-Quality-Chasm-A-New-Health-System-for-the-21st-Century.aspx

[2] *Quality Data Model, Version 5.3 Annotated*. Centers of Medicare & Medicaid Services; Office of the National Coordinator for Health Information Technology, 2017. https://ecqi.healthit.gov/cql

[3] *Clinical Quality Language (CQL), DSTU R1.1*. HL7, April 2015. http://www.hl7.org/implement/standards/product_brief.cfm?product_id=400

[4] HL7, *Representation of the Health Quality Measures Format (eMeasure) Release 1*. HL7, June 2017. http://www.hl7.org/implement/standards/product_brief.cfm?product_id=97

[5] *HL7 Version 3 Implementation Guide: Quality Data Model (QDM)-based Health Quality Measure Format (HQMF), R1.4 – US Realm, Volume 2 (Draft Standard for Trial Use)*. HL7, October 2016. http://www.hl7.org/implement/standards/product_brief.cfm?product_id=346

[6] *HL7 Implementation Guide for CDA Release 2: Quality Reporting Document Architecture – Category I STU Release 5 (US Realm)*. HL7, ballot cycle September 2017.

[7] *Measure Authoring Tool*. CMS. https://www.emeasuretool.cms.gov/

[8] *Measures Management Blueprint v12.0*. CMS, April 2016. https://www.cms.gov/Medicare/Quality-Initiatives-Patient-Assessment-Instruments/MMS/MMS-Blueprint.html

[9] *Refinement, Constraint and Localization, Release 2*. HL7, September 2015. http://www.hl7.org/v3ballotarchive_temp_52E32C7C-1C23-BA17-0CA99EC07A928F9D/v3ballot/html/infrastructure/conformance/conformance.html

[10] *Value Set Authority Center*. U.S. National Library of Medicine. https://vsac.nlm. nih.gov/