FHIR-Based Exchange Implementation Guide

Version 1.0
Dec 1, 2020

Table Of Contents

1. Introduction

This Implementation Guide outlines policy, technical, and process requirements for Implementers of the FHIR-Based Exchange Use Case, under the terms of the Carequality Connected Agreement (CCA), and their Carequality Connections (CCs), under the Carequality Connection Terms.

The FHIR-Based Exchange Use Case addresses the need for FHIR Resources/Bundles containing relevant healthcare information to be made available to appropriate parties across the healthcare ecosystem. A hospital may need information held by a primary care physician, who in turn may need information from a specialist or emergency department. A payer may need information from any of these clinical settings. Government agencies may need information from private sector organizations. This Implementation Guide provides for flexibility across a wide array of access purposes and healthcare settings. Access to information for treatment purposes may have some additional requirements, but widespread exchange across a broad swath of permitted purposes is envisioned for this FHIR ecosystem. Sections 3-6 outline the policy and process requirements, while Sections 7-8 detail the technical pieces.

2. Roles

The concept of a role within this Use Case is central to this Implementation Guide and to defining the rights, obligations, and responsibilities of Carequality Implementers and CCs. Implementers and CCs play a declared role or roles, and Implementers must indicate to Carequality, during the application process for each use case, which role or roles the Implementer will fill, and which role or roles each of its CCs fill.

By default, any requirement specified herein applies to any Implementer or CC regardless of role. Requirements that apply only to those Implementers or CCs with a particular role or roles will clearly indicate the role or roles to which they apply.

An Implementer may fill different roles than its CCs, or may not actually fill any role at all. For example, an Implementer may provide network support, services, and oversight but play no direct role in the transactions specified for this Use Case. The only roles defined in this Use Case are those who initiate queries (“clients”) and those who respond to queries (“servers”).

2.1 Query Initiator

An Implementer or CC with the declared role of a Query Initiator institutes queries to retrieve information held by Implementers or CCs in the Query Responder role. An Implementer or CC with the declared role of a Query Initiator shall support the technical actor(s) specified in Sections 7-8 of this Guide, and comply with any other requirements throughout this Guide that are specifically described as applying to the Query Initiator (client) role.

2.2 Query Responder

An Implementer or CC with the declared role of a Query Responder provides information in response to queries by Implementers or CCs in the Query Initiator role. An Implementer or CC with the declared role of a Query Responder shall support the technical actor(s) specified in Sections 7-8 of this Guide, and comply with any other requirements throughout this Guide that are specifically described as applying to the Query Responder (server) role.

3. Customizable Principles of Trust

3.1 Permitted Purposes

Carequality Implementers and Carequality Connections (CCs) represent a diverse set of stakeholders that wish to exchange health information for a variety of reasons. In order to establish trust, it is important to identify a shared set of acceptable reasons to initiate a query for information (Permitted Purposes). The Permitted Purposes for queries to be made under this Use Case are:

1. Treatment
2. Payment
3. Health Care Operations
4. Public Health Activities
5. Patient Request
6. Coverage Determination
7. Other Authorization-Based Disclosures

The first four terms are used as defined in the Health Insurance Portability and Accountability Act (“HIPAA”) and its implementing regulations, 45 C.F.R. Part 160 and 45 C.F.R. Part 164, Subparts A and E, Standards for Privacy of Individually Identifiable Health Information, and 45 C.F.R. Part 164, Subpart C, Security Standards for the Protection of Electronic Protected Health Information. Public Health Activities are those permitted pursuant to 45 C.F.R. Part 164.512(b).

An Implementer or CC may claim the Patient Request permitted purpose for queries that are directly initiated by the patient or the patient’s personal representative as defined by 45 CFR 164.502(g), via a personal health record or other consumer-facing application. Note that any requests initiated by individuals other than the patient or personal representative may not use the Patient Request permitted purpose, even if the patient has indicated that he or she wishes for the request to occur. For queries initiated directly by the patient’s personal representative, the Query Initiator is responsible for ensuring that the individual initiating the query is, in fact, authorized and appropriate to act as the personal representative as defined by HIPAA.

An Implementer or CC that is not a Covered Entity as defined by HIPAA may claim the Coverage Determination permitted purpose if the request is pursuant to an authorization as defined by HIPAA, and the request is for the purpose of making a determination of eligibility for, or ongoing administration of, disability benefits, life insurance, or other insurance or similar benefits. Note that a health plan or other Covered Entity must claim the Payment permitted purpose when making requests for similar purposes. Note that the primary intent of the Coverage purpose of use is to inform Query Responders that the particular request is being made by an organization that is not a Covered Entity. Providing this level of detail allows responders to make fully informed access policy decisions.

An Implementer or CC may claim the Other Authorization-Based Disclosures permitted purpose if the request is pursuant to an authorization as defined by HIPAA, and the request does not qualify for the Coverage Determination permitted purpose as defined above.

Not every Implementer will support all of the Permitted Purposes allowed for the FHIR-Based Exchange Use Case. Therefore, each Implementer shall identify to Carequality the Permitted Purposes that it and each of its CCs support.

3.2 Permitted Users

Implementers SHALL require users to be identity proofed at a minimum of Identity Assurance Level two (IAL2)[1] prior to issuance of credentials. Non-patient request users that are not identity proofed to IAL2, but were proofed to Level of Assurance three (LOA3)[2] and have maintained that level of identity proofing will be sufficiently identity proofed for Carequality. Exception: When using credentials from a data holder system for a Patient Request to that data holder, IAL2 identity proofing of the user by the client app operator is not required.

Requests for equipment information via FHIR queries, including but not limited to bed availability, SHALL be restricted to agencies or authorities of a State, a territory, or a political subdivision of a State or territory, that are responsible for public health matters as part of their official mandate. Such agencies or authorities shall be responsible for identifying individuals within the agency or authority that SHALL have access to such data via a Carequality Implementer.

When a user is authenticating to a datasource, the authentication MUST follow one of the two following flows:

1. The user MUST present the data source credentials provided by that data source and client app implementer MAY additionally present proof of IAL2 identity proofing of the user
2. The client app implementer which completed IAL2 identity verification of the user SHALL provide proof of IAL2 identity proofing and demographic attributes sufficient for a demographic match

For Patient Requests, a Query Responder(1) MUST accept its own credentials to allow exchange of data or (2) MAY accept IAL2 credentials plus demographic data as defined in 3.3.2 to allow exchange of data. When a Query Responder supports the latter method (2), the Query Responder SHALL NOT refuse to

[1] NIST Special Publication 800-63-3 Digital Identity Guidelines, available at: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63-3.pdf
[2] NISTIR 8062 An Introduction to Privacy Engineering and Risk Management in Federal Systems, available at: https://nvlpubs.nist.gov/nistpubs/ir/2017/NIST.IR.8062.pdf

exchange data when sufficient demographic attributes have been provided to obtain an accurate demographic match.

3.3 Data Integrity

It is clear to all stakeholders that the health information stored in EHRs would be more easily transacted over data sharing networks if the information was better structured into universally accepted formats. As of 2020, these formats do not exist or, if they exist, they are not universally accepted. The clear goal of Carequality is to make progress toward greater structure over time. While that work is being done, Implementers that are Query Responders are allowed to decide whether they share information that the Implementer, or its CCs, hasn't yet been confirmed as being accurate or clinically relevant. Some refer to the process of confirming the accuracy or clinical relevance of information as "vetting". An Implementer that is a Query Responder may choose not to share with Query Initiators information that has not been vetted. A Query Responder that does respond to a query with information will assure that whatever information is sent is an accurate representation of the information contained in the responding system.

3.3.1 Provenance

Implementers SHALL use the Provenance Resource[3] as a means to define the source of the data.

3.3.2 Patient Matching

Query Responders SHOULD have the capability to return more than one potential patient match when the patient search yields more than one match. Query Responders SHALL NOT return more than one potential match when such action would be a violation of HIPAA or other Applicable Law, as with the Patient Request Permitted Purpose for example. When Query Initiators request only "certain" matches of operation $match (i.e. require "onlyCertainMatches"=true), Query Responders SHALL honor that request by returning only a unique match. Query Responders SHALL NOT return more than 100 potential matches when onlyCertainMatches is set to false. Query initiators SHOULD (to the fullest extent possible) normalize all patient demographic data elements according to USCDI standards before attempting to query. The lone exception to this rule is for address, which MUST conform to the USPS standard[4]. Queries for information MUST include all demographic parameters that are available and can be sent and are not constrained by local policy. Demographics SHOULD follow, at a minimum, USCDI defined demographics. [5] Query responders SHALL NOT require more than all USCDI demographics plus administrative gender before returning a patient list response.

[3] For information on the Provenance resource, see: https://www.hl7.org/fhir/provenance.html

[4] USPS Publication 28, Postal Address Standards, available at: https://pe.usps.com/cpim/ftp/pubs/Pub28/pub28.pdf

[5] *United States Core Data for Interoperability* (USCDI v1 Summary of Data Classes and Data Elements) - available at: https://www.healthit.gov/isa/united-states-core-data-interoperability-uscdi

3.3.3 Propagating Corrections

Implementers SHOULD propagate corrections, after the fact, upon discovery of sending an incorrect patient correlation. Example: Yesterday, a Query Responder sent John Smith as a patient match to a Query Initiator. Today, the Query Responder identified that Jake Smith was actually the correct patient for that query. Query Responders SHALL make all reasonable efforts to contact the Query Initiator, make it aware of the error, and provide the correct patient information.

3.4 Best Practices

While this guide does contain Service Level Agreements (SLAs), the following "Best Practices" SHOULD be followed and may be converted to SLAs at a later date. To the extent that Implementers or CCs discover that any of these Best Practices are impractical or would benefit from modification, Carequality should be informed promptly so that the feedback can be considered in the future conversion of these Best Practices into SLAs.

3.4.1 Error Responses

FHIR errors SHOULD use the OperationOutcome capability to return both human readable and machine processable information with sufficient detail to allow the client to determine if the error can be corrected at the client side, such as via a retry operation due to the resource being busy, or a fatal error. Implementers MAY choose to obscure some of these details for security reasons. Any such choices SHOULD be linked to identified security concerns.

3.4.2 Version Compatibility

Implementers and Connections shall continue to support any capabilities previously supported for Carequality purposes under a particular FHIR version, until support for that FHIR version has been officially sunsetted by Carequality. Carequality will provide advance notice of such sunsetting and will collaborate with the Implementer community to develop reasonable timelines for such sunsetting.

3.4.3 Response times

Implementers SHOULD achieve the quickest response time possible per resource. Implementers MAY prioritize response times based on the Permitted Purpose and/or relevant metadata, if present, of the request.

3.4.4 Access Token Lifetime

Authorization Servers SHALL issue access tokens with a lifetime no longer than 60 minutes. An Authorization Server MAY also issue a refresh token to an application using the authorization code grant type. If the Authorization Server issues a refresh token to an application that has requested and has been authorized to use the "offline_access" scope, the refresh token lifetime SHALL be no less than three months unless a shorter lifetime aligns with applicable institutional policies.

3.5 Service Level Agreements (SLAs)

Carequality believes the following SLAs are reasonable and Implementers SHALL make every attempt to comply with them. However, because the industry has very limited experience with FHIR deployed across a national exchange ecosystem, Carequality will undertake a one-year evaluation of these SLAs. During the evaluation period and until otherwise determined by Carequality, a violation of the SLAs will not be treated as a breach of the Carequality Connected Agreement or Carequality Connection Terms. The one-year evaluation period will begin on the date of the first production transaction governed by the terms of this Implementation Guide. Upon completion of this evaluation period, Carequality MAY provide further guidance, adopt updates to the SLAs in a subsequent version of this Guide, or extend the evaluation period at its discretion.

3.5.1 Planned downtime

Implementers and CCs SHOULD schedule planned downtime for time periods with the lowest transaction volume, ideally after 3 a.m. Eastern Time and before 6:00 a.m. Eastern Time, as long as this time period in fact has the lowest transaction volume in the Implementer's or CC's experience with FHIR transactions enabled by Carequality. Downtime is considered to be "planned" if it occurs with at least 48 hours advance notice and SHALL be denoted as such in the Carequality Directory, once the necessary data fields are supported by the Carequality Directory and further guidance is provided in Carequality's Directory Policy. Planned downtime SHALL NOT be scheduled to exceed 72 hours, although Carequality acknowledges that downtime may last longer than anticipated when the downtime was planned, due to unexpected events.

3.5.2 Unplanned downtime notification

A CC MUST attempt to communicate an inability to respond to a request, for any reason, to itsImplementer within a reasonable amount of time. An implementer MUST disseminate a CC's or Implementer's own inability to respond within a reasonable time of discovery of the outage via an update to their Carequality Directory record indicating their status. (This requirement becomes operational only when the Carequality Directory supports such a status indication.)

3.5.3 Uptime

Implementers SHOULD measure uptime on a monthly basis at the Gateway level. Such measurements should only take into account unplanned downtime. Implementers SHOULD strive to achieve 99.9% uptime. The proposed uptime for enforcement by a future SLA is 99.5%

3.6 Full Participation

It is important that all Implementers, CCs and their End Users understand that others are committed to participate in this Use Case so that all those who participate can realize value for their investment of time and resources. An Implementer or CC that plays the role of Query Responder for this Use Case, as defined in Section 2 of this Guide, SHALL provide information in response to valid queries for the Permitted Purpose of Treatment, unless doing so would violate Applicable Law or the Implementer's or CC's local access policies, or unless the data available through the Implementer or CC is of a nature such that it is inappropriate. Further, an Implementer or CC that plays the role of Query Responder for this Use Case, as defined in Section 2 of this Guide, SHALL provide information in response to valid queries for the Permitted Purpose of Patient Request, if the user making the request authenticates using valid credentials issued by the Query Responder, and unless doing so would violate Applicable Law or the Implementer's or CC's local access policies, or unless the data available through the Implementer or CC is of a nature such that it is inappropriate. An Implementer or CC may provide information in response to queries for other Permitted Purposes, but is not required to do so. An Implementer or CC is permitted to serve ONLY in the role of Query Initiator for the Permitted Purposes of Treatment and Patient Request only if that Implementer or CC is a government agency, is a provider organization with no clinical information that could reasonably be made available for response as defined in Section 3.7.1 below, or is an EMS provider with alternative provision of data, as defined in Section 3.7.2 below. An Implementer or CC, other than a government agency or those defined below in subsections of Section 3.7, who wishes to be a Query Initiator for treatment and patient request purposes must also play the role of Query Responder for treatment and patient request purposes.

An Implementer who is, or who provides access to, directly or via its CCs, one or more organizations that are subject to the exceptions listed in the previous paragraph, MUST list each such organization – as defined in this specific case to be the smallest separate business entity that as a whole meets the exception requirements – in the Carequality Directory as a distinct, separate entry. For clarity, note that an individual in solo practice could be an "organization" for purposes of this requirement. These entries must label the organization, in the Organization Type (Org-Type) field, as one of the following values, as appropriate based on that organization's exception:

- Government Agency (Initiator Only)
- Provider Organization (Initiator Only)
- EMS Provider (Initiator Only)

Organizations that do not qualify for the exceptions listed in the previous paragraph MUST NOT be assigned these Org-Type values, so that the Carequality community can immediately discern which organizations are claiming an exception.

3.6.1 Provider Organizations Without Electronic Clinical Information

An Implementer or CC that is a healthcare provider organization is considered to have no available clinical information for response when clinicians within that Implementer or CC primarily maintain patient documentation on paper or otherwise outside of an EHR system, and the organization's staff are only able to initiate queries through a web portal or other mechanism provided by a third party. For clarity, an organization that maintains patient clinical documentation and supports clinician workflows with an electronic system does NOT qualify as having no clinical information for response, if the inability to respond is due to such electronic system's lack of support for the specifications outlined in this Guide.

3.6.2 Emergency Medical Services (EMS) Providers with Alternative Data Sharing Methods

An Implementer or CC is considered to be an EMS Provider if its primary healthcare activity is patient transport with paramedic support. For clarity, taxi and other transport services lacking skilled support are not EMS Providers. Additionally, organizations providing patient transport in addition to other healthcare services, such that patient transport is not the organization’s primary healthcare activity, are not EMS Providers. Further, such EMS Provider is considered to have an alternative data sharing method if the organization to which it is transporting the patient can reasonably expect to receive a summary of any care provided in the course of transport in a format such that the summary can be included in the organization’s electronic record for the patient. Such formats include but are not limited to Direct message and fax. Failure to provide a summary in isolated cases does not disqualify an EMS Provider from having an alternative data sharing method, as long as the organization to which the patient is being transported can reasonably expect such a summary.

3.7 Access Policies

This Section outlines requirements for Implementers and CCs who wish to communicate access policy requirements and their fulfillment within transactions for this Use Case. Implementers and CCs have discretion under Carequality’s local autonomy principle to define access policies that may restrict the release of information for specific patients to other Implementers and CCs, with the limitation that such access policies may only be based on clinical or legal sensitivity of the information, or on the required patient permission that may be needed for the information to be released.

3.7.1 Patient Permission

Throughout this and other sections of the Implementation Guide, the term “Patient Permission Form” refers to a form that provides the Query Responder with the requisite legal authority to exchange or release the patient’s records. Depending on the circumstances, a Patient Permission Form may be a consent form or an authorization, as the two terms are defined by HIPAA. Patient Permission Forms must be signed by the patient in question or by their personal representative (as defined by 45 CFR 164.502(g)). Signatures may be in electronic form. To be clear, this section refers to access policy decisions made for individual patients rather than agreements between organizations. The internal application of these access policies may be quite complex and highly variable among Query Responders, based on each Query Responder’s definition of clinical and legal sensitivity of different elements of patient records. In general, however, there are four possible categories into which the access policies will fall for any given permitted purpose:

1) The Responder’s access policies do not support access for the specific permitted purpose of the query, at all.
2) The Responder’s access policies never allow the release of information for the asserted permitted purpose, without specific additional permission or other mitigating circumstances such as a medical emergency.

3) The Responder’s access policies may prohibit the release of information for the asserted permitted purpose, without additional permission or other mitigating circumstances, based on attributes of the particular patient record being queried.
4) The Responder’s access policies always allow the release of information to valid Carequality requesters for the asserted permitted purpose

If a Query Responder’s policies for a permitted purpose fall into categories (1) or (4), there is no role for additional information from the Query Initiator and the remainder of this Section is largely inapplicable for that permitted purpose. For Query Responders whose policies fall into categories (2) or (3), however, additional input from the Query Initiator could be essential in determining whether or not information may actually be released in response to any individual query. In order to provide such additional input in a consistent way, such that Query Responders may evaluate whether or not it aligns with local access policies, Carequality defines a set of specific policy assertions that are available to Query Initiators.

These two options generally do not require any special behavior on the part of the Responder. While generally discouraged, Outcome 1 is the most restrictive access policy wherein all requests made for a specific permitted purpose are denied. Outcomes 2 & 3 require the Responder to make specific access decisions for specific initiator’s request(s). For example, suppose that Peaceful Valley Hospital has a record for John Smith. Peaceful Valley Hospital has been queried for this record by physician practice Adventist Medical.As a matter of policy, PeacefuI Valley Hospital will release patient files only if it receives a signed consent from the patient or the patient’s personal representative (category 2). Upon receiving the query without an indication of a signed document, Peaceful Valley will request additional documentation in response or will not release John Smith’s information to Adventist Medical.

3.7.2 Policy Assertions

In addition to asserting a Permitted Purpose, Implementers and Connections may also assert Access Policies. Access Policy Assertions are concepts defined by Carequality which represent standardized policy constructs accessible to all Implementers. These assertions provide detailed information to the Query Responder about the Initiator’s capabilities and permissions. Policy assertions SHALL be in the form of the OIDs as defined below:

Policy Assertion	Access Consent Policy Identifier	Requirements for the Initiator
Verbal Consent	urn:oid:2.16.840.1.113883.3.7204.1.1.1.1.1	The patient who is the subject of the transaction must be physically present at the facility initiating the query and have provided clear verbal confirmation of their consent to have records released by the Query Responder to the Query Initiator. The verbal consent must have been provided directly to the staff member initiating the query

Collected Initiator’s Signed Patient Permission Form (available in band)	urn:oid:2.16.840.1.113883.3.7204.1.1.1.1.2	The Query Initiator must have collected a Patient Permission form containing all of the elements required for it to be a valid consent or authorization, as appropriate, under HIPAA, signed by the patient or an authorized representative. The specific text of the form is at the Query Initiator’s discretion, as long as it contains at a minimum the HIPAA required elements. An electronic copy of the Patient Permission form must be available for retrieval by the Query Responder via a link in the OAuth acp_reference field. Note that technical issues preventing the retrieval of an individual document do not constitute a failure of the Query Initiator to meet the requirements for this Policy Assertion, as long as a pattern of consistent failures does not emerge such that the Query Initiator must reasonably expect that Query Responders may be unable to retrieve Patient Permission documents

Collected Initiator’s Signed Patient Permission Form (**Unavailable** in band)	urn:oid:2.16.840.1.113883.3.7204.1.1.1.1.3	The Query Initiator must have collected a Patient Permission form containing all of the elements required for it to be a valid consent or Authorization, as appropriate, under HIPAA, signed by the patient or an authorized representative. The specific text of the form is at the Query Initiator’s discretion, as long as it contains at a minimum the HIPAA required elements. The Query Initiator does not support a mechanism for retrieving an electronic copy of the Patient Permission document and the Query Responder shall not assume that it will be able to retrieve the Patient Permission document prior to making its access policy decision on whether or not to release records in response to the Query Initiator’s request. The Query Initiator shall, however, provide a copy of the form to the Query Responder in response to reasonable requests after the fact
Collected Responder’s Signed Patient Permission Form (available in band)	urn:oid:2.16.840.1.113883.3.7204.1.1.1.1.5	The Query Initiator must have collected an unaltered Patient Permission form signed by the patient or an authorized representative, with the text of the form being specified by the Query Responder to meet the Query Responder’s access policy requirements. The Query Initiator must have documented

		evidence of the Query Responder’s intent for the form to be used in this manner, either directly in the form of an email or other communication, or indirectly through the Query Responder’s submission of the form or form text to a system or service that the Query Responder knows will distribute the form or form text for purposes of facilitating the use of this Policy Assertion. An electronic copy of the Patient Permission form must be available for retrieval by the Query Responder. Note that technical issues preventing the retrieval of an individual document do not constitute a failure of the Query Initiator to meet the requirements for this Policy Assertion, as long as a pattern of consistent failures does not emerge such that the Query Initiator must reasonably expect that Query Responders may be unable to retrieve Patient Permission documents
Collected Responder’s Signed Patient Permission Form (**Unavailable** in band)	urn:oid:2.16.840.1.113883.3.7204.1.1.1.1.6	The Query Initiator must have collected an unaltered Patient Permission form signed by the patient or an authorized representative, with the text of the form being specified by the Query Responder to meet the Query Responder’s access policy requirements. The Query Initiator must have documented

		evidence of the Query Responder's intent for the form to be used in this manner, either directly in the form of an email or other communication, or indirectly through the Query Responder's submission of the form or form text to a system or service that the Query Responder knows will distribute the form or form text for purposes of facilitating the use of this Policy Assertion. If the Query Initiator does not support a mechanism for retrieving an electronic copy of the Patient Permission form, the Query Responder shall not assume that it will be able to retrieve the Patient Permission form prior to making its access policy decision on whether or not to release records in response to the Query Initiator's request. The Query Initiator must, however, provide a copy of the Patient Permission form to the Query Responder in response to reasonable requests after the fact
Collected Initiator's Signed Patient Permission Form (**Available** for electronic request within 10 days)	urn:oid:2.16.840.1.113883.3.7204.1.1.1.1.4	The Query Initiator must have collected a Patient Permission form containing all of the elements required for it to be a valid authorization as defined by HIPAA, signed by the patient or an authorized representative. The specific text of the form is at the Query Initiator's discretion, as long as it contains at a minimum the

		HIPAA required elements. The Query Initiator supports a mechanism for retrieving an electronic copy of the Patient Permission form, but is not able to provide a copy at the time of the request, and the Query Responder shall not assume that it will be able to retrieve the Patient Permission form prior to making its access policy decision on whether or not to release records in response to the request. The Query Initiator must, however, make a copy of the Patient Permission form available to the Query Responder in response to an appropriate document query after no more than 10 business days
Collected Responder's Signed Patient Permission Form (**Available** for electronic request within 10 days)	urn:oid:2.16.840.1.113883.3.7204.1.1.1.1.7	The Query Initiator must have collected an unaltered Patient Permission form signed by the patient or an authorized representative, with the text of the form being specified by the Query Responder to meet the Query Responder's access policy requirements. The Query Initiator must have documented evidence of the Query Responder's intent for the form to be used in this manner, either directly in the form of an email or other communication, or indirectly through the Query Responder's submission of the form or form text to a system or service that the Query Responder knows will distribute

		the form or form text for purposes of facilitating the use of this Policy Assertion. The Query Initiator supports a mechanism for retrieving an electronic copy of the Patient Permission form, but is not able to provide a copy at the time of the request, and the Query Responder shall not assume that it will be able to retrieve the Patient Permission form prior to making its access policy decision on whether or not to release records in response to the request. The Query Initiator must, however, make a copy of the Patient Permission form available to the Query Responder in response to an appropriate document query after no more than 10 business days
Public Health Emergency	urn:oid:2.16.840.1.113883.3.7204.1.1.1.1.8	The Query Initiator must be making its request for information in the context of a state of emergency that has been declared by state or federal officials. The specific patient who is the subject of the query must reasonably be associated with the declared emergency. For example, an outbreak of measles reaches an extent that it is declared a Public Health Emergency by local authorities. From this point on, queries in the affected area should include the Public Health Emergency policy assertion for patients who are

		impacted by the measles outbreak. Most such queries will likely be for Treatment, but could also be for the Public Health purpose of use. Other purposes of use are less likely to be aligned with this policy assertion, but the use of this assertion is not forbidden for other purposes, as long as the Query Initiator can reasonably claim that the query is associated with the declared emergency
Emergency	urn:oid:2.16.840.1.113883.3.7204.1.1.1.1.9	The Query Initiator must be making its request in the context of an imminent threat to the health and safety of a patient or others as defined in 45 CFR 164.512(j)(1)(i). The Query Initiator must comply with reasonable follow-up requests from the Query Responder in order to comply with the Query Responder's regulatory obligations, including without limitation collecting a signed form after the fact, or providing information on the nature of the emergency
Patient Verified NIST Identity Assurance Level 2	urn:oid:2.16.840.1.113883.3.7204.1.1.1.1.10	The Query Initiator must be making a request on behalf of the patient that is directly initiated within the Query Initiator's system by the patient. The Query Initiator must have verified the patient's

		identity in a manner compliant with NIST Identity Assurance Level 2, as described in NIST publication SP 800-63A. The Query Initiator may rely on a third party registration authority's identity verification but takes full responsibility for the identity verification complying with the NIST Identity Assurance Level 2
Authorized Personal Representative Verified NIST Identity Assurance Level 2	urn:oid:2.16.840.1.113883.3.7204.1.1.1.1.11	The Query Initiator must be making a request on behalf of the patient as requested by the patient's authorized personal representative as described in 45 C.F.R. § 164.502(g) of the HIPAA Regulations. The personal representative's request must be directly initiated within the Query Initiator's system. The Query Initiator must have verified the personal representative's identity in a manner compliant with NIST Identity Assurance Level 2, as described in NIST publication SP 800-63A. The Query Initiator may rely on a third party registration authority's identity verification but takes full responsibility for the identity verification complying with the NIST Identity Assurance Level 2
Patient Verified NIST Identity Assurance Level 3	urn:oid:2.16.840.1.113883.3.7204.1.1.1.1.12	The Query Initiator must be making a request on behalf of the patient that is directly

		initiated within the Query Initiator's system by the patient. The Query Initiator must have verified the patient's identity in a manner compliant with NIST Identity Assurance Level 3, as described in NIST publication SP 800-63A. The Query Initiator may rely on a third party registration authority's identity verification but takes full responsibility for the identity verification complying with the NIST Identity Assurance Level 3
Authorized Personal Representative Verified NIST Identity Assurance Level 3	urn:oid:2.16.840.1.113883.3.7204.1.1.1.1.13	The Query Initiator must be making a request on behalf of the patient as requested by the patient's personal representative as described in 45 C.F.R. § 164.502(g) of the HIPAA Regulations. The personal representative's request must be directly initiated within the Query Initiator's system. The Query Initiator must have verified the personal representative's identity in a manner compliant with NIST Identity Assurance Level 3, as described in NIST publication SP 800-63A. The Query Initiator may rely on a third party registration authority's identity verification but takes full responsibility for the identity verification complying with the NIST Identity Assurance Level 3 (IAL 3). Note: All policy assertions

		should be asserted individually, even when one policy implies compliance with another. In the case of the Policy Assertions related to NIST IALs, while asserting IAL 3 implies compliance with IAL 2, the Query Initiator must assert both IAL 2 AND IAL 3
Information from Substance-Abuse Facilities Covered Under 42 CFR Part 2 Can Be Accepted	urn:oid:2.16.840.1.113883.3.7204.1.1.1.1.14	The Query Initiator must be able to comply with requirements for handling information from substance abuse treatment facilities covered under 42 CFR Part 2, and specifically must be able to prevent the unauthorized disclosure of any such information outside the entity specifically identified as the requesting entity by virtue of the Home Community Identifier used in the query transactions

3.7.3 Requirements for Query Responders

Query Responders are permitted to make access denial decisions based on the Initiator's Permitted Purpose as well as Access Policy Assertions it asserts. If the Query Responder finds that its access policies allowing the release of records have not been satisfied by internal action, such as by collection of a form that generally authorizes such releases, and are not satisfied by the combination of the Query Initiator's Permitted Purpose and any Access Policy Assertions included with the query, it may indicate to the Query Initiator which of the Carequality Policy Assertions, if any, would allow access to the identified patient's records.

If the Query Responder indicates that one or more Policy Assertions would allow access to a patient's records, and the Query Initiator completes the requirements for the relevant Policy Assertion(s) and includes the Policy Assertion(s) in a subsequent request for that patient's records, the Query Responder MUST provide access to the records unless there has been a change to the patient's record in the meantime such that the particular Policy Assertion(s) no longer satisfy the Query Responder's access policies. It is expected that such an occurrence would generally be rare, and that Query Responders MUST generally release records if a Query Initiator asserts a Policy Assertion that the Query Responder recently indicated would allow access to these records. If the Query Responder has received an opt-out from exchange by the patient or their personal representative, this should override any Patient Permission assertion from a Query Initiator. As long as the Query Responder supports a particular query's Permitted Purpose/User, Query Responders MUST perform patient matching on the query prior to responding, in the absence of any technical error.

If a patient match is identified, the Query Responder MUST assess its access policies for that patient to determine if they have already been satisfied by the Query Responder's internal actions, for example by collecting a form authorizing the release of information. If this assessment reveals access policy requirements that are still outstanding, the Query Responder MUST then assess any Carequality Access Policy Assertions made by the Query Initiator, to see if they satisfy the outstanding requirements.

Query Responders are permitted to never release information for a supported specific Permitted Purpose or to refuse to release information, including the fact that a record exists, without specific permission. However, the practice of an Implementer or CC refusing in their response to disclose that a matching record exists is discouraged, to the extent allowed by HIPAA, for all Implementers and CCs that are not substance abuse treatment facilities covered under 42 CFR Part 2, or other mental and behavioral health facilities that have significant restrictions placed on their release of information under applicable law.

3.8 The Role of Vendor App Stores

Notable in the value proposition discussion for Carequality is the lack of any mention of integration into user workflows within EHRs. Carequality can provide standardization and access to data, but does not provide any assurance of integration at the workflow level into an individual EHR. For some use cases, including virtually all use cases involving cross-organization exchange, access to the data is all that is needed. For many intra-organization use cases, however, there is a need for not only access to data, but meaningful integration at a workflow level with an organization's core IT systems.

Some EHR vendors have developed "app store" constructs to address this factor, with a process for vetting an app's ability to meaningfully interoperate with the EHR. For a subset of potential Carequality participants that can loosely be described as "provider apps", this process can be expected to continue to provide value independent of the value provided by Carequality. Stated differently, having access to data, while necessary, may not be sufficient to actually enroll users to request that data. The target users of provider apps are far more likely to adopt a product if it has been validated by the relevant vendor from a workflow and integration standpoint. For this reason, Carequality SHALL permit a vendor validation process, including the charging of fees by the vendor to those app providers who go through such a process, as long as the process carries no implication for data access via Carequality.

4. Non Discrimination

Interoperability is impaired if organizations are free to impose whatever terms they choose as a condition of exchanging information. All Carequality Implementers and CCs that choose to participate in FHIR-Based Exchange will do so without imposing unfair or unreasonable conditions that would limit exchange or interoperability with other Carequality Implementers and CCs that are similarly situated. A condition is unfair or unreasonable if it results in similarly situated Implementers, or their CCs, being treated differently. Whether two Implementers or CCs are similarly situated is determined primarily by the purpose for which the information is being exchanged, although other considerations may apply in specific circumstances as described below.

4.1 Non Discrimination – Treatment and Patient Request

Carequality has the goal of enabling widespread exchange of health information on a nationwide scale, between many partners who do not have any direct relationship with one another outside of Carequality. Recognizing that the time and effort required to reach individual contractual agreements, including those whose purpose is to define fee payment terms, between all of these potential partners can be a barrier to widespread exchange, Implementers and CCs cannot impose any additional fees, terms, or conditions on other Implementers or CCs with respect to queries/responses for treatment or patient request purposes. No additional agreements beyond the Carequality legal framework are allowed to be required in order to honor queries for these two permitted purposes.

With respect to treatment, the type of organization initiating the query is not a factor (although organizations claiming treatment must actually be providing treatment, or be making the request on behalf of a network member that is providing treatment.)

With respect to patient requests, Implementers MUST respond to these types of requests, if the request is successfully authenticated via OAuth. Honoring queries without credentials, i.e. based solely on demographics matching, will be permitted but not required. Additional policy details regarding patient requests are noted in Section 3.2 of this Implementation Guide.

4.2 Non Discrimination – Other Permitted Purposes

Implementers and CCs are permitted, but not required, to impose fees, terms and conditions on the Implementers or CCs with respect to queries or responses for any permitted purpose other than treatment and patient requests (as noted in section 4.1). Any fees, terms, and conditions must comply with the entirety of Section 4 of this Implementation Guide.

Implementers that play the role of Query Responder are not REQUIRED to honor queries other than for the treatment or patient request permitted purposes. However, Query Responders may CHOOSE to honor queries for other permitted purposes. If a Query Responder does choose to honor queries beyond treatment and patient requests purposes, it must honor said queries (for that permitted purpose) from all Query Initiators, unless (i) to do so would violate applicable law; (ii) it has chosen to honor queries only from particular government agencies as further outlined in Section 4.3; (iii) it has chosen to impose terms and conditions on Query Initiators, and has not reached agreement on such terms and conditions with a particular Query Initiator, as further described in Section 4.3; or (iv) the permitted purpose is Other Authorization-Based Disclosures.

For example, Peaceful Valley Hospital has received queries for John Smith's record from payers Acme Healthcare and Insure America for the purpose of payment. Peaceful Valley Hospital has a contract with Acme Healthcare outlining additional terms for the exchange, including data element requirements. Peaceful Valley Hospital may choose to honor payment queries from Acme Healthcare, but not Insure America, if Insure America has not agreed to similar terms, subject to the additional requirements of Section 4.3 below.

Note, Carequality anticipates further work to more fully define the Other Authorization-Based Disclosures permitted purpose. Until such additional definition is completed, however, different Query Initiators may, in good faith, make Other Authorization-Based Disclosure queries using the same PurposeOfUse value that in fact stem from very different circumstances. Given this uncertainty, Query Responders are free to choose which queries to honor under this permitted purpose. Query Responders are strongly encouraged, however, to honor queries for this permitted purpose equally from any organization, when the circumstances for the queries are generally similar.

4.3 Non Discrimination – Consistency in Additional Terms and Conditions

If an Implementer or CC chooses to impose additional terms and conditions on other Implementers and CCs with respect to performing or responding to queries for permitted purposes beyond treatment and patient requests, such terms and conditions MUST NOT vary based on the type of organization that the other Implementer or CC is. For example, a Query Responder cannot impose one set of conditions on health care providers and another set of conditions on health care payers for queries based on the same permitted purpose. However, acknowledging that some permitted purposes are quite broad, a Query Responder's terms and conditions may limit its responses to queries for that permitted purpose to specific workflows or types of data use, which may in turn result in the Query Responder only exchanging, in practice, with specific types of organizations. For example, queries by health plans for case management, queries by home health services in support of administrative intake processes, and queries by EMS services in support of post-event staff training follow-up, could all arguably fall under the permitted purpose of "Operations". A Query Responder's terms and conditions restricting responses for a permitted purpose to one or more workflows are acceptable under these Non-Discrimination requirements so long as they (i) focus on a particular workflow as elucidated by the examples above, although not limited to the examples above; (ii) do not result in differential treatment of similarly situated organizations that engage in the relevant workflow; and (iii) are not specific to the approach of a particular exchange partner or partners, such that others who engage in the same conceptual workflow would be excluded.

In addition, it is acceptable for a Query Responder to treat local, state or federal government agencies differently from other Implementers and CCs. For example, a Query Responder can choose to respond to queries for payment from CMS but not from commercial insurers. Also, a Query Responder may accept a fee for providing information in response to a query from the Social Security Administration without charging a fee to other Query Initiators.

Except as noted above with respect to government agencies, additional terms and conditions must be imposed consistently on all other Implementers and CCs that perform or respond to queries for the same permitted purpose. An Implementer or CC may impose different fees on different Implementers and CCs, but the differences must be based on a consistently-applied set of objective, economically relevant criteria such as organization size or transaction volume.

If an Implementer or CC offers particular terms to one party, it must make good faith efforts to reach similar terms with other parties who perform or respond to queries for the same Permitted Purpose, subject to the exception for government agencies noted above. If a party feels that good faith efforts to reach terms are not being made, it may file a dispute under the Carequality Dispute Resolution Process.

4.4 Non Discrimination – Policy Assertion Acceptance

If a Query Responder will accept a particular Policy Assertion(s) from one Query Initiator, it MUST accept that Policy Assertion(s) from any other Query Initiator for the same permitted purpose. This requirement applies equally to unsolicited Policy Assertions from the Query Initiator and to those assertions made after the Query Responder has indicated which Policy Assertions would satisfy its access requirements. For example, suppose that Peaceful Valley Hospital has a record for John Smith. Peaceful Valley Hospital has been queried for this record by physician practices Adventist Medical and Children First. If Adventist Medical asserts that it meets the requirements of the Verbal Consent policy, and Peaceful Valley considers this assertion from Adventist Medical to satisfy its access policies for John Smith, then Non Discrimination requires that it must also consider a Verbal Consent assertion from Children First to satisfy its access policies.

4.5 Non Discrimination – Access Policies

Query Responders are prohibited from enforcing different access policies based on attributes of the organization making the request. Stated differently, if a Query Initiator can legitimately claim a particular permitted purpose, the Query Responder must treat the request the same as any other for that permitted purpose, regardless of the Query Initiator's organization type or other attributes. Note that this requirement relates to the access policy itself, not necessarily the outcomes of evaluating that access policy for any individual request, since such outcomes may vary based on a number of factors including attributes of the particular patient records being requested. Also note that this requirement refers to general access policies set by the organization, and does not prevent a Query Responder from honoring an individual patient's wishes to restrict release of his or her records to particular organizations.

Similarly, a Query Responder can't waive access policy requirements for a particular Query Initiator, or enforce additional access policy requirements for a particular Query Initiator, relative to what is required of other Query Initiators for a particular permitted purpose. While restricting access to patient data based on asserted Access Policy Assertions provides responders with additional flexibility, it is not intended (and is, in fact, not permitted) to be used to discriminate against any particular Query Initiator in accordance with the rest of the Non Discrimination section of this guide.

Several of the Access Policy Assertions – those referring to a Patient Permission form being "available in band" – apply to situations in which the Query Initiator has collected a consent form, and is able to provide a copy of that form to the Query Responder, upon request. The following requirements for Query Initiators apply specifically and only to those Query Initiators who are listed in the Carequality Directory as supporting the inclusion of Policy Assertions in messages. Query Initiators MUST assert all policy assertions for which the Query Initiator meets the requirements. Note: All policy assertions SHOULD be asserted individually, even when one policy implies compliance with another. For example, in the case of the Policy Assertions related to NIST Identity Assurance Levels (IALs), meeting the requirements for IAL3 implies that the requirements for IAL2 have also been met. Nonetheless, Query Initiators who can assert IAL3 SHOULD also assert IAL2. Compliance with this practice will remove complexity, and allow for forward compatibility, in the Query Responder's rule evaluation. If Carequality has not provided a field in its Directory that identifies Query Initiators who support the inclusion of Policy Assertions, a Query Initiator MAY choose to send policy assertions with some queries but not with others, but if any policy assertions are asserted, the Query Initiator MUST assert all policy assertions for which the requirements are met.

5. Resource Usage

If an Implementer or CC updates its endpoints listed in the Carequality Directory for any reason other than FHIR version support, the Implementer or CC MUST continue to support transactions received at its previously listed endpoint(s) for a minimum of 14 days after updating its endpoints in the Carequality Directory.

6. Evidence of Compliance

Prior to implementing production connectivity for the workflows/transactions specified for this Use Case, each Implementer SHALL complete a non-production test with 3 other Implementers whose connectivity relies on software provided by a different technology vendor or provider (the Test Partner). Implementers who themselves do not play a role in this Use Case may designate a CC to perform the test, or perform the test using an internal environment as long as that environment has the same code base that will be delivered to the Implementer's CCs.

The non-production partner test will consist of successful execution of each transaction required for the role or roles declared by the Implementer as being played either directly by that Implementer or by its CCs. The success of the test will be at the discretion of the Test Partner, but Test Partners SHOULD NOT report success unless each transaction has been completed and data returned to the other party in that transaction. Specifically, matching patient data MUST be found, at least one FHIR resource MUST be available, and one or more resources MUST be retrieved. Data should be coordinated among the test partners such that patient matching is successful. Upon completion of the test to the Test Partner's satisfaction, the Test Partner will independently inform Carequality that the Implementer's non-production partner test was successfully completed.

After completing the non-production partner test and meeting the applicable requirements of the Carequality Application Process, an Implementer MAY configure its production system for connectivity via the transactions specified for this Use Case. Prior to being recognized as a live Implementer of this Use Case, the Implementer must complete connectivity validation in production. Until this validation is successfully completed, Implementers are not considered live and MAY NOT claim such status. Further, until this validation process is successfully completed, other Implementers are not obligated to engage in exchange activities with the Implementer, other than those required for the connectivity validation as described in this Section.

The connectivity validation will consist of two steps. In the first step, basic connectivity is confirmed through authentication. Implementers in the Query Initiator role, or who support CCs in the Query Initiator role, must then be able to retrieve a FHIR resource with at least 50% of all other live Implementers. The aforementioned Initiator requirements/validation rules also apply to Implementers in the Query Responder role.

7. Use Cases/Workflows

7.1 Patient Discovery

Assumptions:

- The Query Initiator knows a sufficient number of the patient's demographics for a successful match
- The Service Directory has all endpoints for the Query Responder
- The user is either a Healthcare Provider or other Healthcare system user with acceptable Purpose of Use, or, a Patient/Caregiver with allowed access to the Patient record
- If a Patient/CareGiver is using Query Responder portal credentials, those credentials have been granted to that user and sufficient identity proofing has been done by the Query Responder prior to granting the credentials

Nominal Flow:

1. The workflow begins when the Query Initiator queries the Carequality Service Directory for the Endpoint and information for the Query Responder.
2. If the Query Initiator does not have a valid client_id for use with the Query Responder, then:
    a. The Query Initiator asserts a dynamic registration to the Query Responder authorization server providing their Carequality certificate, a software statement and other JWT metadata as per section 8.3.3. The Query Initiator's Purpose of Use matches one of those listed in the Query Initiator's Service Directory Entry and is one supported by the Authorization server, if required.

   b. The Authorization Server returns the client_id assigned or, in case of renewal, re-assigned according to the capabilities of the server.
3. Query Initiator, using the provided client_id, requests an access token as per Section 8.3.4 or 8.3.5. The request includes, as part of the authorization extension object, one of the six supported NHIN Purpose of Use codes. If the authorization_code grant type is used as specified in Section 8.3.4, the credentials supplied follow one of the two following options:
   a. The Query Initiator provides the user credentials and may include proof of IAL2 proofing. These credentials are accepted by the Query Responder and an access token is granted
   b. If supported by the Query Responder, the Query Initiator provides proof of IAL2 identity proofing and all known demographics
      i. If the Query Responder does not accept demographic authorization, the Query Responder will return a Invalid Request error.
      ii. If supported, the Query Responder executes a user match against the demographics collected by the Query Responder out of band and one of the following outcomes occurs:
         1. Accepts the match and proofing as sufficient for access and grants an access token.
         2. Requests additional demographics from the Query Initiator. In this case, the Query Initiator may collect and provide further demographics and re-initiate the access request.
         3. Deny the access request due to insufficient confidence in the user demographic match. The workflow ends.
4. The Query Initiator executes a Patient search using the $match operation as per Section 8.5.2
   a. Patient Resource includes all known demographics. The following optional results may occur
      i. The Query Responder returns the requested Patient Resource identifier
      ii. The Query Responder rejects the request, requiring additional demographics. In this case, the Query Initiator can collect and provide further demographics and re-initiate the query
5. The Query Initiator, once the Patient search has been successfully executed, begins an Information Query
6. An audit log of the request is made by both the Query Responder and Query Initiator

Alternate flow 1: The Query Initiator includes Access Consent

1. The Query Initiator includes, as part of their access request, an Access Consent Policy as follows:
   a. The acp element includes OID as appropriate from Policy Assertion Table in Section 3.8
   b. The acp_reference element includes a link to a Consent or DocumentReference resource which holds the information as needed according to the OID in the acp field.
   c. One of the two following results occurs:
      i. The ACP is accepted and the access token returned

        ii. The ACP is not accepted and, as part of the response, an OID from the Policy Assertion Table in Section 3.8 and a URL to the appropriate policy or form required is returned, indicating further or different requirements.

2. The workflow continues.

Post Conditions:

- The Query Initiator has an access token necessary for a follow-up information query.
- The Access Token is valid for the period as per section 8.3.5.

7.2 Information Query

Assumptions:

- The Query Initiator has fulfilled the requirements of Section 7.1 and has a valid access token.

Nominal Flow:

1. The Query Initiator queries the Query Responder for their FHIR server's CapabilityStatement.
2. The CapabilityStatement is checked against the Query Initiator's requirements for Resource, Profile and FHIR Implementation Guide requirements. The following options result:
    a. No match between the Query Initiator's requirements is found, the workflow ends The Query initiator may need to contact the Query Responder out-of-band to resolve the mis-match or may need to fall back to core/US Core profiles for the information needed. How this is done is out of scope for this IG.
    b. The CapabilityStatement matches, in whole or in part, the requirements and the workflow continues.
3. The Query Initiator executes query(ies) for the information regarding the Patient queried in Section 7.1 as limited by the scopes in the software statement provided in the authorization request.
4. The Query Responder returns Resources that match the Query Initiator's request.
5. An audit log of the request(s) is made by both the Query Responder and Query Initiator.

Post Conditions:

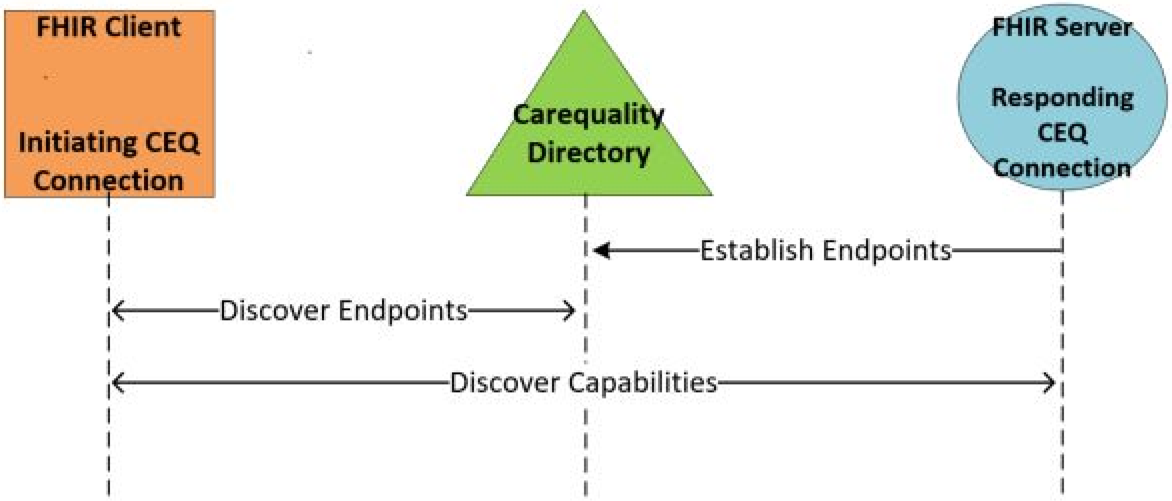
The Query Initiator has the information needed for proper patient care.

8. Infrastructure

8.1 FHIR Endpoints

To enable a lookup of Carequality Connections that support HL7® FHIR® based access and exchange, Carequality needs to establish a common place to query for such information. Carequality SHALL deploy a single Carequality FHIR directory covering both FHIR based and Document Exchange based transactions instead of spinning up a brand new directory for FHIR. A value SHALL be added, called "FHIR R4", to the existing directory extension named "UseCases." This will allow Implementers to distinguish between the Query Based Document Exchange and FHIR based endpoints.

The FHIR CapabilityStatement resource SHALL be used to define the FHIR capabilities for an endpoint. This CapabilityStatement will only be made available by the server and will not be copied into the Carequality Directory. This approach will minimize redundant data and associated maintenance, thus reducing out-of-date/sync capability statements while reducing the number of centralized points to establish a connection that could fail. Further implementation experience MAY yield adding other data to the Carequality Directory, but that will be addressed later based on implementation feedback.



Link to Directory IG: https://carequality.org/healthcare-directory/index.html

8.2 Discovery of Endpoint Capabilities

Discovery of Endpoints shall be executed by a query to the Carequality directory service which will have the FHIR endpoint(s) for the implementers and connections. If multiple endpoints exist for an implementer or connection, the CapabilityStatement for each endpoint would list the capabilities of the instance. Carequality Implementers supporting the FHIR-Based Exchange Use Case shall deploy at least one FHIR CapabilityStatement that is publicly discoverable, where CapabilityStatement.kind=”instance”, and is supported/endorsed by Carequality. Implementers must also support at least one FHIR resource per Capability Statement that they deploy.

8.3 Authentication/Trust

The goal is to establish an approach that supports establishing trust at scale, i.e., when an organization is ready to become a Carequality Connection endpoint for FHIR based access/exchange, there should be minimal steps for that connection and implementer to “connect” that organization to Carequality, while connecting to existing Carequality Connections that can use FHIR.

To support scaling of authentication, Carequality will deploy a decentralized authentication approach where the Carequality Implementer establishes one or more authorization servers to support their connections or may share an authorization server with one or more other Carequality Implementers.

Carequality trusted X.509 certificates will be used to enable establishing trust of the calling application. Note that no “client secret” will be utilized as part of this approach as interactions described above using the Carequality certificate sufficiently enable the server to assess whether the calling application is trusted and that trust has not expired.

8.3.1 Carequality Certificates

Introduction

The Use of Carequality certificates for the FHIR ecosystem SHALL conform to the certificate policy and profile requirements described in the Carequality Technical Trust Policy, with the additions and exceptions as noted below.

Issuance

Certificates SHALL be issued to the Implementer or Connection responsible for the security of a FHIR Application, as determined by the Implementer’s deployment model, also referred to as the Operator. A FHIR Application is a client application and/or a service that makes or responds to requests described in this guide.

If multiple instances of the same application are secured by different Operators, then each Operator MUST be issued a separate certificate. However, for convenience, a single Operator MAY group various client and/or server functions together as a single Application using a single certificate, or divide them into separate Applications using separate certificates, subject to the restrictions below. Depending on organization policy, certificates issued to a single Operator MAY be issued on a per-organization basis (e.g. when one Operator secures the same application on behalf of multiple organizations) or MAY be issued more granularly on a per-application basis (e.g. when one Operator wishes to use separate certificates for software components that run on different servers or perform different functions).

Note that grouped software components that share one certificate will be treated as one application by Authorization Servers and, thus, MUST be able to use a single client_id assigned by the Authorization Server. If using a single client_id is not practicable, then using separate certificates for each component would be an appropriate alternative.

TLS certificates used in the Carequality QBDE environment SHALL NOT be used by FHIR Applications.

Structure

The value of the Common Name SDN attribute SHALL be a human readable name for the Application as provided by the Operator. The Operator's legal business name, city, and state SHALL be included in the Subject Distinguished Name (SDN) of the certificate as the values of the Organization, Locality, and State attributes.

The Operator and the FHIR Application SHALL be jointly identified by a unique URI listed in a uniformResourceIdentifier entry in a certificate's Subject Alternative Name (SAN) extension, i.e. each certificate is issued in the context of an Operator and an Application. If a certificate is used to secure an https service endpoint, then the host's DNS name SHALL also be included in the SAN extension as a dnsName entry.

The subject key SHALL be an RSA key (2048 bit) or an elliptic curve key (P-256 or P-384 curves). Note that implementations are not required to support elliptic curve signatures at this time.

Server Considerations

On the server side, Carequality Implementers can deploy as either a single-tenant gateway or multi-tenant gateway. In the single-tenant case, there is a one-to-one relationship between X.509 certificates and Carequality Connections (CCs). In the multi-tenant case, where a single certificate is used and there is more than one CC per host, the Implementer SHALL be identified as the Operator. Both scenarios are allowed.

A Carequality Implementer with multiple CCs hosted behind a single gateway MAY be deployed with only one certificate for all of their CCs. In this case, a single certificate will be issued for that Implementer and that Implementer will be entered into the Directory. Subsequently, as that Implementer's CCs become ready to exchange, each CC will be added to the Directory, but no additional certificate will need to be issued since it is behind the same gateway. Stated differently, multi-tenant scenarios will result in one Carequality Directory entry per CC.

Unlike the SOAP-based QBDE transactions in which a client authenticates to the server during a mutual TLS handshake with the server, client authentication for the workflows described in this guide is achieved using tokens that are digitally signed by the client as described in section 8.3.2. Thus, a server SHALL NOT additionally require client authentication at the time of the TLS handshake to access the OAuth 2.0 and FHIR endpoints identified for these workflows.

8.3.2 Use of JWT Signatures

This guide makes use of the JSON Web Token (JWT) and JSON Web Signature (JWS) specifications to create digitally signed JWTs that establish the authenticity of participants requesting client registration or client authentication. All JWTs defined in this guide MUST:

1. conform to the mandatory requirements in RFC 7515 and RFC 7519,
2. conform to the additional JWS header and JWT claims requirements in this guide,
3. be digitally signed using the signer's private key that corresponds to the public key listed in the signer's Carequality X.509 certificate,
4. be digitally signed using one of the permitted signature algorithms listed in this guide,
5. include the signer's Carequality X.509 certificate in the 'x5c' JWS header parameter array as per Section 4.1.6 of RFC 7515, and
6. be serialized using JWS Compact Serialization as described in Section 7.2 of RFC 7515.

All JWTs defined in this guide SHALL contain a Javascript Object Signing and Encryption (JOSE) header conforming to the following requirements:

Carequality JWT Header Values		
alg	required	A string containing the JWA algorithm used for signing the JWT. All implementations SHALL support RS256, SHOULD support ES256, and MAY support ES384 and RS384. For example: "RS256"

x5c	required	An array of one or more strings containing the X.509 public key certificate or certificate chain [RFC5280] corresponding to the key used to digitally sign the JWS. Each string in the array is a base64-encoded DER representation of the certificate, with the key used to sign the JWS being first. See https://tools.ietf.org/html/rfc7515#section-4.1.6

8.3.3 Client Registration

These requirements are based this based on Unified Data Access Profiles DRAFT 2019-05-15
as specified at http://www.udap.org

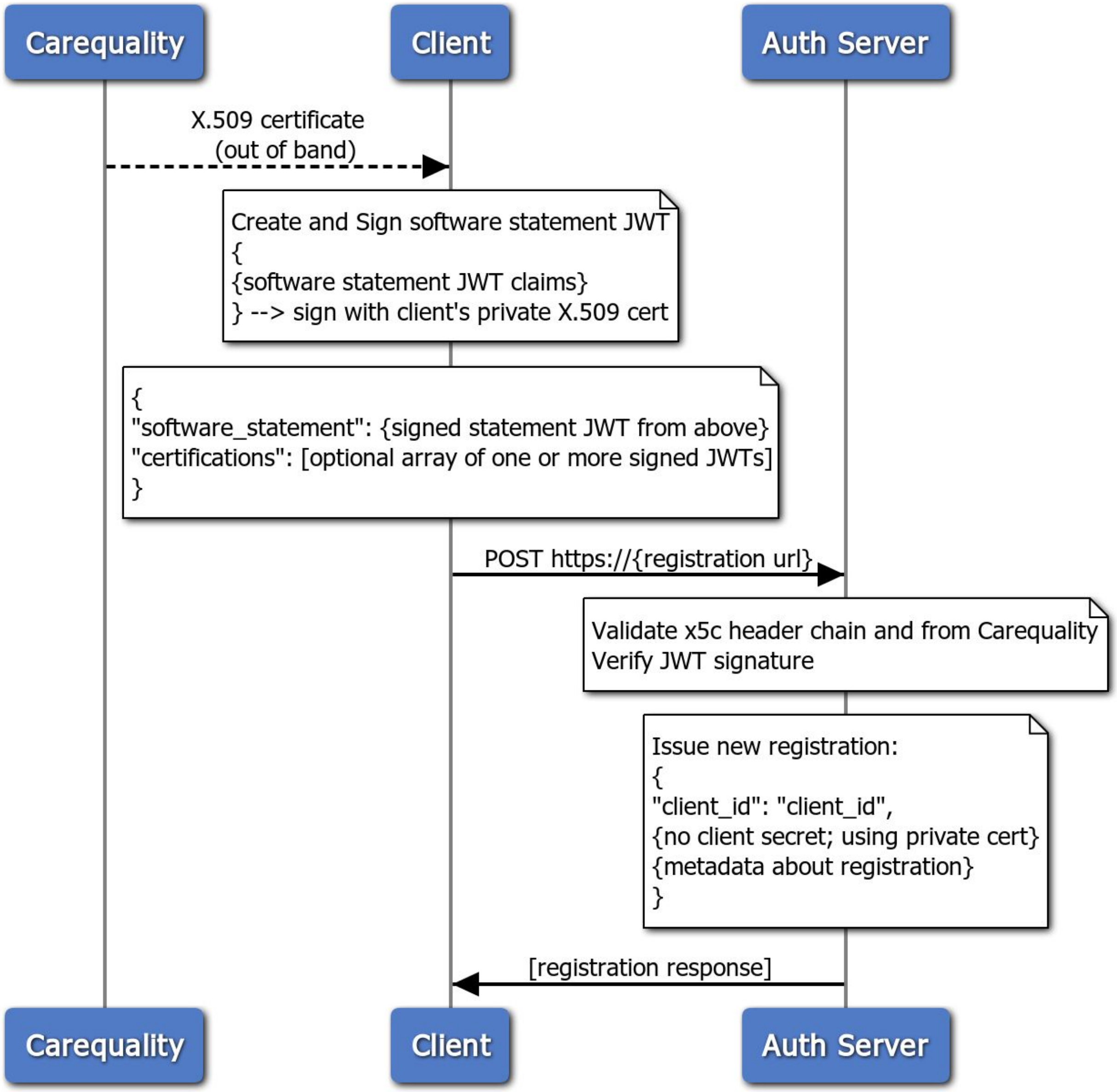
Before proceeding, the Initiating Carequality Connection's solution MUST have been registered with the Responding Carequality Connection's authorization server. This process MUST be scalable and MUST NOT require manual steps for every Responding Carequality Connection's authorization server. Carequality Implementers SHALL support dynamic registration as described in this section to enable the necessary scaling without manual intervention. Carequality participants MUST register their applications with the authorization servers that protect the FHIR servers with whom they wish to exchange data. Beyond establishing details about application names and ownership, registration also establishes intended authorization workflows and FHIR resources that they wish to exchange.

Dynamic Registration

Carequality FHIR implementation does not allow for Public clients, which do not have a private key, in this version. Future versions of the Guide may allow for this use case. Two use cases are currently supported:

1. Confidential clients: Conventional server-based web applications that can maintain a secret,
2. Backend services: for business-to-business connections, backend services can access data directly, without a user directly in the loop.

OAuth 2.0 Dynamic Registration


Carequality
Client
Auth Server
X.509 certificate
(out of band)
Create and Sign software statement JWT
{
{software statement JWT claims}
} --> sign with client's private X.509 cert
{
"software_statement": {signed statement JWT from above}
"certifications": [optional array of one or more signed JWTs]
}
POST https://{registration url}
Validate x5c header chain and from Carequality
Verify JWT signature
Issue new registration:
{
"client_id": "client_id",
{no client secret; using private cert}
{metadata about registration}
}
[registration response]
Carequality
Client
Auth Server

8.3.3.1 Registration API

Discovery

A FHIR Server MUST make its Authorization Server's authorization, token, and registration endpoints available to client applications as follows:

1. include the endpoints in its CapabilityStatement available at [baseURL]/metadata using the OAuth 2.0 URIs Extension on the rest.security element as per Section 3.1 of the HL7 SMART App Launch Framework, using the "authorize", "token", and "register" components,
2. include the endpoints in the JSON object available at [baseURL]/.well-known/smart-configuration as defined in Section 4 of the HL7 SMART App Launch Framework (referred to in this guide as the SMART configuration data), using the "authorization_endpoint", "token_endpoint", and "registration_endpoint" keys,
3. if it includes a "token_endpoint_auth_methods" key in its SMART configuration data, include "private_key_jwt" as one of the elements in the array value of this key,
4. indicate UDAP support in its CapabilityStatement by including the UDAP code http://fhir.udap.org/CodeSystem/capability-rest-security-service|UDAP in the rest.security.service element as in the example below, and
5. include the required metadata defined below at [baseURL]/.well-known/udap as per section 1 of UDAP Dynamic Client Registration.

The URLs listed above MUST be accessible to client applications without requiring client authentication.

Example CapabilityStatement excerpt showing items 1 and 4 from the list above:

```
{
 "resourceType": "CapabilityStatement",
 ...
 "rest": [
  {
   "mode": "server",
   "security": {
    "extension": [
     {
      "url": "http://fhir-registry.smarthealthit.org/StructureDefinition/oauth-uris",
      "extension": [
       {
        "url": "token",
        "valueUri": "https://baseurl.example.com/token"
       },
       {
        "url": "authorize",
        "valueUri": "https://baseurl.example.com/authz"
       },
       {
        "url": "register",
        "valueUri": "https://baseurl.example.com/register"
       },
       ...
      ]
     },
     ...
    ],
    "service": [
     {
      "coding": [
       {
        "system": "http://hl7.org/fhir/restful-security-service",
        "code": "SMART-on-FHIR"
       }
      ],
      "text": "OAuth2 using SMART-on-FHIR profile (see http://docs.smarthealthit.org)"
     },
     {
      "coding": [
       {
        "system": "http://fhir.udap.org/CodeSystem/capability-rest-security-service",
        "code": "UDAP"
       }
      ],
      "text": "OAuth 2 using UDAP (see http://www.udap.org)"
     },
     ....
    ],
    ...
   },
   ...
  },
  ...
 ],
 ...
}
```

Required UDAP Metadata

The metadata returned from the UDAP metadata endpoint defined above SHALL conform to the requirements listed in the table below and SHALL represent the server's capabilities for the workflows described in this guide. For elements that are represented by arrays, returning an empty array SHALL be interpreted by clients to mean that the corresponding capability is NOT supported by the server.

udap_versions_supported	required	A fixed array with one string element: ["1"]
udap_certifications_supported	recommended	An array of zero or more certification URIs supported by the Authorization Server, e.g.: ["https://wiki.carequality.org/udap/profiles/basic-app-certification"]
udap_certifications_required	recommended	An array of zero or more certification URIs required by the Authorization Server, e.g.: ["https://wiki.carequality.org/udap/profiles/basic-app-certification"]
udap_authorization_extensions_supported	recommended	An array of zero or more key names for authorization extension objects supported by the Authorization Server, e.g.: ["carequality"]
udap_authorization_extensions_required	recommended	An array of zero or more key names for authorization extension objects required by the Authorization Server, e.g.: ["carequality"]

grant_types_supported	recommended	An array of one or more grant types supported by the Authorization Server, e.g.: ["client_credentials"]
scopes_supported	recommended	An array of one or more strings containing scopes supported by the Authorization Server. The server MAY support different subsets of these scopes for different client types or entities, e.g.: ["system/Patient.read", "system/AllergyIntolerance.read", "system/Procedures.read"]
authorization_endpoint	recommended	A string containing the URL of the Authorization Server’s authorization endpoint
token_endpoint	recommended	A string containing the URL of the Authorization Server’s token endpoint.
token_endpoint_auth_methods_supported	recommended	Array of one or more authentication methods supported by the Authorization Server, e.g. ["private_key_jwt"]
token_endpoint_auth_signing_alg_values_supported	recommended	Array of strings listing one or more JWA algorithm identifiers supported by the Authorization Server for validation of signed JWTs submitted to the token endpoint for client authentication. All

		implementations SHALL support RS256, SHOULD support ES256, and MAY support ES384 and RS384. For example: ["RS256", "ES384"]
registration_endpoint	recommended	A string containing the URL of the Authorization Server’s registration endpoint
registration_endpoint_jwt_signing_alg_values_supported	recommended	Array of strings listing one or more JWA algorithm identifiers supported by the Authorization Server for validation of signed software statements, certification, and endorsements submitted to the registration endpoint. All implementations SHALL support RS256, SHOULD support ES256, and MAY support ES384 and RS384. For example: ["RS256", "ES384"]

Software Statement

The software statement is a JWT signed by the client using the private key that corresponds to the public key listed in its X.509 certificate. The JOSE header and payload of the JWT are constructed as per section 2 of UDAP Dynamic Client Registration. The JOSE Header shall contain the required elements specified in Section 8.3.2 of this guide. The client signs the software statement using one of the RS256, ES256, RS384, or ES384 signature algorithms as defined in RFC 7518; the algorithm used will depend on whether the client app's X.509 certificate contains an RSA or EC key. All implementations SHALL support RS256, SHOULD support ES256, and MAY support ES384 and RS384.

The unique client URI used for the 'iss' claim SHALL match the uriName entry in the Subject Alternative Name extension of the client app's X.509 certificate. The software statement is intended for one-time use with a single OAuth 2.0 server. As such, the 'aud' claim SHALL list the URL of the OAuth Server's registration endpoint, and the lifetime of the software statement ('exp' minus 'iat') SHALL be 5 minutes.

Inclusion Of Certifications And Endorsements

This model supports the optional certifications framework outlined in UDAP Certifications and Endorsements for Client Applications. Authorization Servers MAY support the inclusion of certifications by Application operators. Authorization Servers SHALL ignore unsupported or unrecognized certifications, i.e., the inclusion of an unsupported or unrecognized certification SHALL NOT be a reason for an Authorization Server to return an error response.

Authorization Servers MAY require client apps to include one or more certifications that are referenced in this guide in a registration request. If a certification is required, the Authorization Server SHALL communicate this by including the corresponding certification URI in the udap_certifications_required metadata element defined above.

This guide defines the Carequality Basic App Certification Profile in Section 8.3.3.2. Application operators MAY include a self-signed certification as defined by that profile. Operators of consumer-facing applications MAY include a self-signed certification as defined by the Carequality Consumer-Facing App Certification Profile. We welcome Early Adopter feedback to this documentation. Please follow the link for more details. Carequality may publish additional certification profiles.

Software Statement JWT Claims		
iss	required	Issuer of the JWT -- unique identifying client URI. This MUST match the value of a uniformResourceIdentifier entry in the Subject Alternative Name extension of the client's certificate included in the 'x5c' JWT header
sub	required	Same as 'iss'. In typical use, the client application will not yet have a client_id from the Authorization Server
aud	required	The Authorization Server's "registration URL" (the same URL to which the registration request will be posted)
exp	required	Expiration time integer for this software statement, expressed in seconds since the "Epoch"

		(1970-01-01T00:00:00Z UTC). This time SHALL be no more than five minutes in the future
iat	required	Issued time integer for this software statement, expressed in seconds since the “Epoch”
jti	required	A nonce string value that uniquely identifies this software statement. This value SHALL NOT be reused by the client app in another software statement or authentication JWT before the time specified in the “exp” claim has passed
client_name	required	A string containing the human readable name of the client application
redirect_uris	conditional	An array of one or more redirection URIs used by the client application. This claim MUST be present if grant_types includes "authorization_code" and this claim MUST be absent otherwise. Each URI MUST use the https scheme
contacts	required	An array of URI strings indicating how the data holder can contact the app operator regarding the application. The array SHALL contain at least one valid email address using the mailto scheme, e.g. ["mailto:qa@example.com"]
logo_uri	conditional	A URL string referencing an image associated with the client application, i.e. a logo. If grant_types includes "authorization_code", client applications SHALL include this field, and the authorization server SHOULD display this logo to the user during the authorization process. The URL SHALL use the https scheme and reference an image file (PNG, JPG, or GIF), e.g. "https://www.example.com/HealthApp.png"

grant_types	required	Array of strings, each representing a requested grant type, from the following list: "authorization_code", "refresh_token", "client_credentials". The array MUST NOT contain both "authorization_code" and "client_credentials". The value "refresh_token" MUST NOT be present in the array unless "authorization_code" is also present
response_types	conditional	Array of strings. This claim MUST be present with a value of "code" as the only array element if grant_types contains "authorization_code", and MUST be omitted otherwise
token_endpoint_auth_method	required	Fixed string value: "private_key_jwt"
scope	required	String containing a space delimited list of scopes requested by the client application for use in subsequent requests. The Authorization Server MAY consider this list when deciding the scopes that it will allow the application to subsequently request

Example software statement, prior to Base64URL encoding and signature (non-normative, the “.” between the header and claims objects is a convenience notation only):

```
{
  "alg": "RS256",
  "x5c": ["MIEE8DCCA.....remainder omitted for brevity"]
}.{
  "iss": "http://example.com/my-application",
  "sub": "http://example.com/my-application",
  "aud": "https://as.example.net/register",
  "exp": 1525209377,
  "iat": 1525209077,
  "jti": "random-jti-generated-by-client"
  "client_name": "My Application",
  "redirect_uris": ["https://example.com/redirect"],
  "contacts": ["mailto:qa@example.com"],
  "logo_uri": "https://www.example.com/HealthApp.png",
  "grant_types": ["authorization_code"],
  "response_types": ["code"],
  "token_endpoint_auth_method": "private_key_jwt",
  "scope": "user/Patient.read", "user/Procedure.read"
}
```

Request Body

The registration request is submitted by the client to the Authorization Server’s registration endpoint.

```
POST /register HTTP/1.1
Host: as.example.net
Content-Type: application/json

{
  "software_statement": "...the signed software statement JWT...",
  "certifications": ["...a signed certification JWT..."]
  "udap": "1"
}
```

The Authorization Server validates and processes the registration request as per Sections 4 and 5 of UDAP Dynamic Client Registration[6]. This includes validation of the JWT payload and signature, validation of the X.509 certificate chain, and validation of the requested application registration parameters. If the registration is successful, the Authorization Server SHALL return an HTTP 201 response which includes the unique client_id that the client app will use to interact with the Authorization Server's authorization and token endpoints and the registration metadata that was accepted. The Authorization Server SHALL grant registration requests received from clients submitting software statements signed with valid Carequality certificates assuming all authorization logic and purpose of use requirements have been reviewed and accepted. Example success and failure responses can be found in sections 5.1 and 5.2 of UDAP Dynamic Client Registration[7].

[6] UDAP Dynamic Client Registration refers to the 2019-05-15 Draft published at http://www.udap.org/udap-dynamic-client-registration.html

[7] http://www.udap.org/udap-dynamic-client-registration.html

8.3.3.2 Carequality Basic App Certification Profile

In most cases, the Application Operator will provide additional authorization metadata to the data holder’s Authorization Server at the time of the token request using the Carequality Authorization Extension Object, as discussed in Sections 8.3.4 and 8.3.5. When some or all of this metadata will not vary across subsequent token requests, e.g. all requests will be for a single purpose of use, then the Application Operator MAY also provide some or all of this authorization metadata to the data holder’s Authorization Server at the time of registration. This is accomplished by including a Carequality Basic App Certification in the certifications array of the registration request. This self-asserted certification re-uses the Carequality Authorization Extension Object keys defined in Section 8.3.5, with the additional requirements described throughout this section. The certification JWT included by the client app MUST conform to the header requirements in Section 8.3.2 and the claims requirements in the following table:

Carequality Basic App Certification JWT Claims		
iss	required	Issuer of the JWT -- unique identifying client URI. This must match the value of a uniformResourceIdentifier entry in the Subject Alternative Name extension of the client’s certificate included in the ‘x5c’ JWT header and MUST match the value of the ‘iss’ claim of the software statement with which this certification is submitted See https://tools.ietf.org/html/rfc5280#section-4.2.1.6
sub	required	Same as ‘iss’
exp	required	Expiration time integer for this self-assertion, expressed in seconds since the "Epoch" (1970-01-01T00:00:00Z UTC). Since this certification may be reused for multiple registrations, the expiration time SHALL be no more than three years after the time the JWT is issued
iat	required	Issued time integer for this authentication JWT, expressed in seconds since the “Epoch”
jti	required	A nonce string value that uniquely identifies this certification JWT. This value SHALL NOT be reused by the client app in another JWT

		with the same ‘iss’ value before the time specified in the ‘exp’ claim has passed
certification_name	required	String with fixed value: "Carequality Basic App Certification"
certification_uris	required	Fixed array with single string element: ["https://wiki.carequality.org/udap/profiles/basic-app-certification"]
extensions	required	A JSON Object containing the key “carequality” with a value equal to a Carequality Authorization Extension Object, as defined in Section 8.3.5, with the additional requirements discussed below.

For the purposes of this certification profile, inclusion of the ‘version’ element of the Carequality Authorization Extension Object defined in Section 8.3.5 SHALL be required. However, inclusion of all other extension object elements SHALL be optional for this certification profile. Specifically, the App Operator MAY include only those extension object keys whose values will not vary across subsequent token requests. For example, if all subsequent data requests by the App will be made by one organization and only for the purpose of treatment, then the ‘organization_id’, ‘organization’, and ‘purpose_of_use’ keys and the corresponding values could be included in the Carequality Authorization Extension Object within this self-declaration. Elements whose value may vary in subsequent token requests SHALL NOT be included in this certification. An App Operation SHOULD NOT submit a Carequality Basic App Certification with the authorization object containing only the ‘version’ element, as such a certification provides no additional information to the data holder’s Authorization Server. When an Application Operator includes a Carequality Basic App Certification in its registration request, an Authorization Server MAY reject subsequent token requests by this app that contain authorization metadata that does not match the corresponding values declared in the certification. In addition, an Authorization Server MAY require that one or more elements of the Carequality Basic App Certification, such as purpose_of_use, be supplied at registration time. Authorization servers that reject a registration request due to a missing element SHOULD respond with an informative error identifying that element.

8.3.3.3 Modifying Registrations

The client URI in the Subject Alternative Name of an X.509 certificate uniquely identifies a single application and its operator over time. A previously registered client application MAY request a modification of its previous registration with an Authorization Server by submitting another registration request to the same Authorization Server’s registration endpoint using a certificate with a Subject Alternative Name client URI entry matching the original registration request. Note that this may be a different certificate than the one used in the previous registration, such as in the case of certificate renewals or re-keyings. The registration request SHALL include all parameters required for a new registration and all parameters for which modification is requested.

If an Authorization Server receives a valid registration request with a software statement containing the same ‘iss’ value as an earlier software statement but with a different set of claims or claim values (other than ‘exp’, ‘iat’, and ‘jti’), or with a different (possibly empty) set of optional certifications and endorsements, the server MUST treat this as a request to modify the registration parameters for the client application by replacing the information from the previous registration request with the information included in the new request. For example, an Application operator may use this mechanism to update a redirection URI or to remove or update a certification. If the registration modification request is accepted, the Authorization Server SHOULD return the same client_id in the registration response as for the previous registration. If it returns a different client_id, it MUST cancel the registration for the previous client_id.

If an Authorization Server receives a valid registration request with a software statement containing the same set of claims and claim values as an earlier software statement (excluding ‘exp’, ’iat’, and ‘jti’), and with the same (possibly empty) set of optional certifications and endorsements, i.e. the client application is requesting registration with unmodified registration parameters, the server SHOULD return the same client_id. If it returns a different client_id, it MUST cancel the registration for the previous client_id.

If an Authorization Server receives a valid registration request with a software statement that contains an empty grant_types array from a previously registered application, the server SHOULD interpret this as a request to cancel the previous registration. A client application SHALL interpret a registration response that contains an empty grant_types array as a confirmation that the registration for the client_id listed in the response has been cancelled by the Authorization Server.

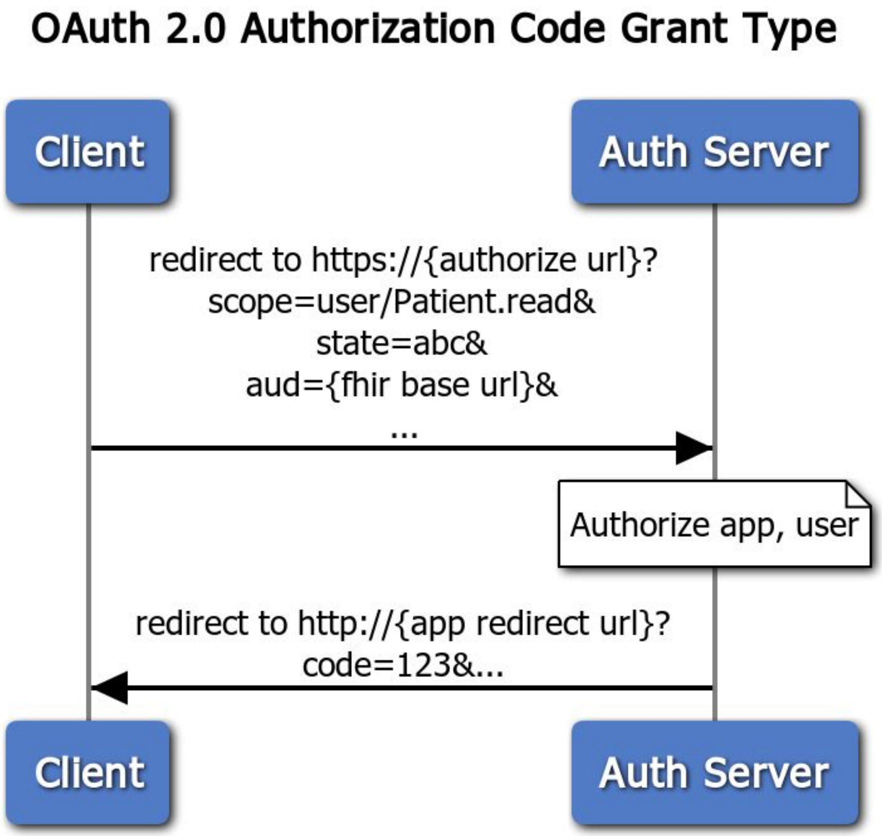
If the Authorization Server returns the same client_id in the registration response for a modification request, it SHOULD also return an HTTP 200 response code. If the Authorization Server returns a different client_id in the registration response, the client application SHALL use only the new client_id in subsequent transactions with the Authorization Server.

8.3.4 Authorization Code Grant Type (3-legged OAuth 2.0)

Applications that wish to exchange data with a FHIR server must authenticate and authorize themselves and their users with an Authorization Server first in order to obtain an FHIR access token. This flow can vary depending on whether an application is a public app or a confidential app. This section will outline the flows for user-facing applications. Only confidential apps are supported in this version of this guide. Client and servers MAY optionally support UDAP Tiered OAuth for User Authentication.

8.3.4.1 Obtaining an Authorization Code

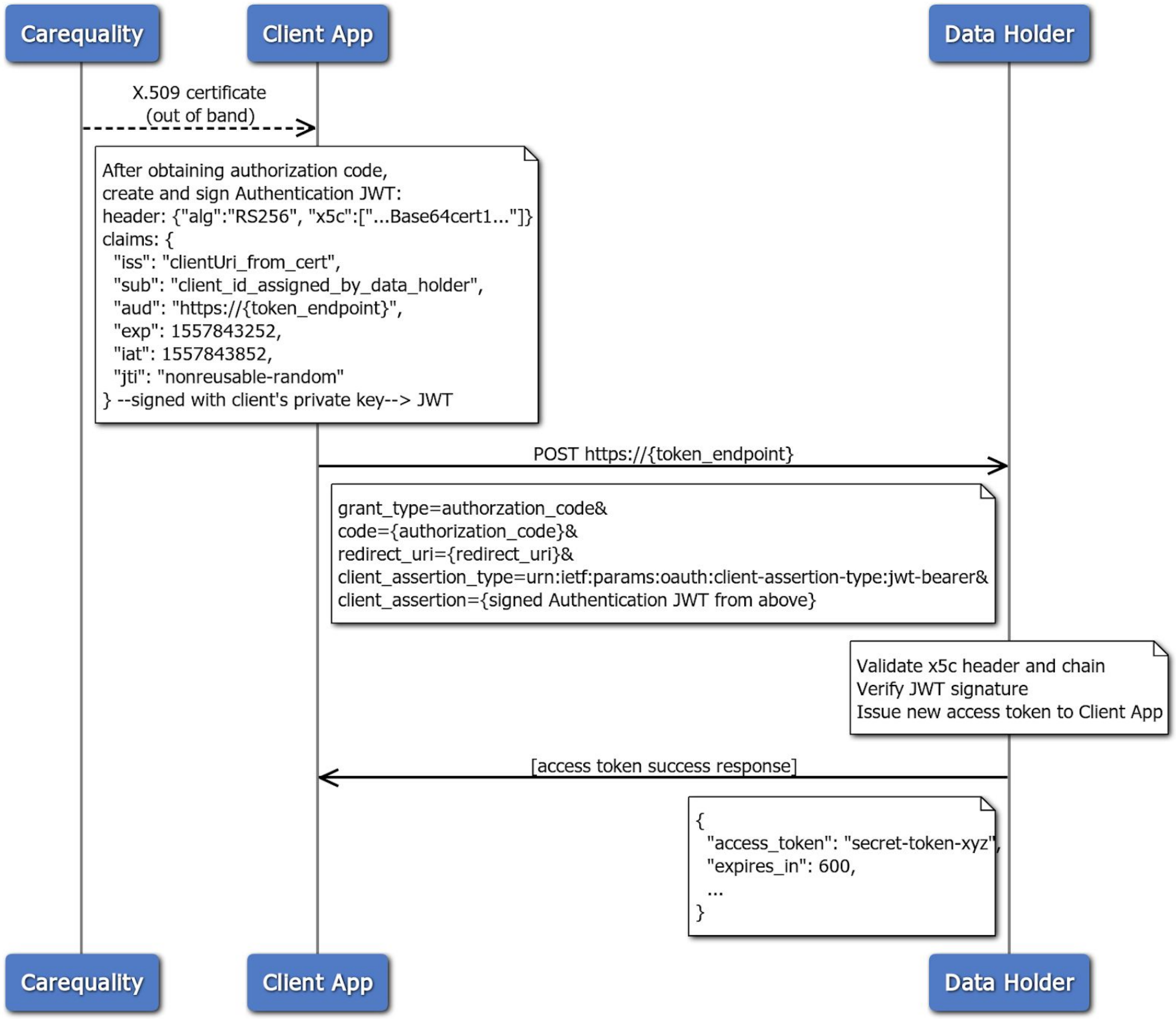
The client application SHALL obtain an authorization code as per the profile outlined by the HL7 SMART App Launch Framework at: http://hl7.org/fhir/smart-app-launch/#step-1-app-asks-for-authorization



8.3.4.2 Obtaining an Access Token

The approach for confidential clients will follow the profile outlined by SMART here (http://hl7.org/fhir/smart-app-launch/#step-3-app-exchanges-authorization-code-for-access-token), but with the following key difference. Rather than confidential clients presenting a client_secret, the client SHALL use its Carequality certificate to sign a client_assertion to prove its identity as described in section 6.1 of http://www.udap.org/udap-jwt-client-auth.html and detailed further below.

OAuth 2.0 Authorization Code flow (Confidential Client)



```
After obtaining authorization code,
create and sign Authentication JWT:
header: {"alg":"RS256", "x5c":["...Base64cert1..."]}
claims: {
  "iss": "clientUri_from_cert",
  "sub": "client_id_assigned_by_data_holder",
  "aud": "https://{token_endpoint}",
  "exp": 1557843252,
  "iat": 1557843852,
  "jti": "nonreusable-random"
} --signed with client's private key--> JWT
```

```
grant_type=authorzation_code&
code={authorization_code}&
redirect_uri={redirect_uri}&
client_assertion_type=urn:ietf:params:oauth:client-assertion-type:jwt-bearer&
client_assertion={signed Authentication JWT from above}
```

```
Validate x5c header and chain
Verify JWT signature
Issue new access token to Client App
```

```
{
  "access_token": "secret-token-xyz",
  "expires_in": 600,
  ...
}
```

The responder typically authenticates the end user during the authorization step when using the authorization_code grant type. For Patient Requests that follow workflow (1) described in Section 3.2 of this guide, identity proofing of the user by the client application is not required. However, if the Implementer operating the requesting application has additionally identity-proofed the end user of its application, then the requesting application MAY provide metadata about the user to the data holder as additional authorization information at the time of the token request by adding this information to the authentication JWT in the form of a Carequality-specific authorization extension as per section 6.2 of http://www.udap.org/udap-client-authorization-grants.html, and as detailed further below.

The user metadata submitted by the requesting application in the extension object SHALL correspond to the verified identity attributes of the permitted user (verified as per Section 3.2) who is making the request. Note that for patient requests (i.e. where the purpose of use code is REQUEST), this user is not necessarily the patient who is the transaction subject, i.e., the verified user MAY instead be a patient’s authorized representative.

A responder SHOULD indicate support or lack of support for this extension object by including or omitting the "carequality_user" key from its list of supported authorization extension objects in its UDAP metadata. If a responder does not support this extension object, it MAY ignore the associated metadata. Alternatively, if the responder has explicitly indicated in its UDAP metadata that this extension object is not supported, it MAY instead return an invalid_request error response for a token request containing this extension object.

If the responder supports the use of the Carequality User Authorization Extension object, the responder is responsible for validating that verified user identity metadata submitted by the application reasonably matches the responder’s own records for the app user that was authenticated by the responder before issuing an access token. If the data holder cannot validate the user information, it SHALL return an invalid_grant error in response to the token request.

The authentication JWT submitted by the client app MUST conform to the header requirements in Section 8.3.2 and the claims requirements in the following table:

Authentication JWT Claims		
iss	required	Issuer of the JWT -- unique identifying client URI. This must match the value of a uniformResourceIdentifier entry in the Subject Alternative Name extension of the client’s certificate included in the ‘x5c’ JWT header See https://tools.ietf.org/html/rfc5280#section-4.2.1.6

sub	required	The service's client_id, as determined during registration with the FHIR authorization server
aud	required	The FHIR authorization server's "token endpoint URL" (the same URL to which this authentication JWT will be posted -- see below)
exp	required	Expiration time integer for this authentication JWT, expressed in seconds since the "Epoch" (1970-01-01T00:00:00Z UTC). This time SHALL be no more than five minutes after the time the JWT is issued
iat	required	Issued time integer for this authentication JWT, expressed in seconds since the “Epoch”
jti	required	A nonce string value that uniquely identifies this authentication JWT. This value SHALL NOT be reused by the client app in another authentication JWT before the time specified in the “exp” claim has passed
extensions	optional	A JSON Object containing the key "carequality_user" with a value equal to a Carequality User Authorization Extension Object, defined below

Carequality User Authorization Extension object **Extension Name: “carequality_user”**		
version	required	Fixed string value: "1"
purpose_of_use	required	Fixed string value: "REQUEST" The purpose for which the data is requested, from the code set of permitted purposes in the NHIN PurposeOfUse code system as per Section 3.1 of the Carequality QBDE IG

user_name	required	JSON Object, value is the requesting user’s verified name represented as a FHIR HumanName element; the element SHALL contain a first name, middle name or middle initial, and last name, and optionally a suffix, e.g. {"family":"Smith", "given":["William", "J."]}
user_address	required	JSON Object, value is the requesting user’s verified address represented as a FHIR Address element; the element SHALL contain a street address, city, state, and postalCode, e.g. {"line":"202 C St", "city":"San Diego", "state":"CA", "zip":"92101"}
user_contact	required	JSON array of JSON Objects, the value of each element is a verified contact point for the requesting user represented as a FHIR ContactPoint element; at least one verified email address and at least one one verified phone number SHALL be included, e.g. [{"system":"email", "value":"wsmith@example.com"}, {"system":"phone", "value":"+1-619-555-1212"}]
user_dob	required	String containing the verified date of birth of the requesting user, formatted as YYYY-MM-DD, e.g. "1976-08-04"
acp	required	The Access Consent Policy Identifier corresponding to the asserted Access Policy that represents the identity proofing level of assurance of the user, array of string values from the subset of valid policy OIDs in section 4.4.1 of the Carequality QBDE IG that represent identity proofing levels of assurance, each expressed as a URI, e.g. ["urn:oid:2.16.840.1.113883.3.7204.1.1.1.1.12"] to represent identity proofing of the user at IAL2

The parameters for the POST request to the Authorization Server’s token endpoint MUST conform to the requirements in the following table:

grant_type	required	Fixed value: authorization_code
code	required	Code that the app received from the authorization server
redirect_uri	required	The same redirect_uri used in the initial authorization request. The redirect_uri SHALL use the https scheme.
client_assertion_type	required	Fixed value: urn:ietf:params:oauth:client-assertion-type:jwt-bearer
client_assertion	required	Signed Authentication JWT value (see above)
udap	required	Fixed value: 1

Authorization Servers SHALL issue access tokens with a lifetime no longer than 60 minutes. An Authorization Server MAY also issue a refresh token to an application using this grant type. If the Authorization Server issues a refresh token to an application that has requested and has been authorized to use the “offline_access” scope, the refresh token lifetime SHALL be no less than three months unless a shorter lifetime aligns with applicable institutional policies. If an application that has requested and has been authorized to use the “offline_access” scope presents a valid refresh token to an Authorization Server to obtain a new access token, the Authorization Server SHOULD also issue a new refresh token valid for a new period of no less than three months unless a shorter lifetime aligns with applicable institutional policies.

Non-normative example (white space and line breaks added for clarity, not URL-encoded):

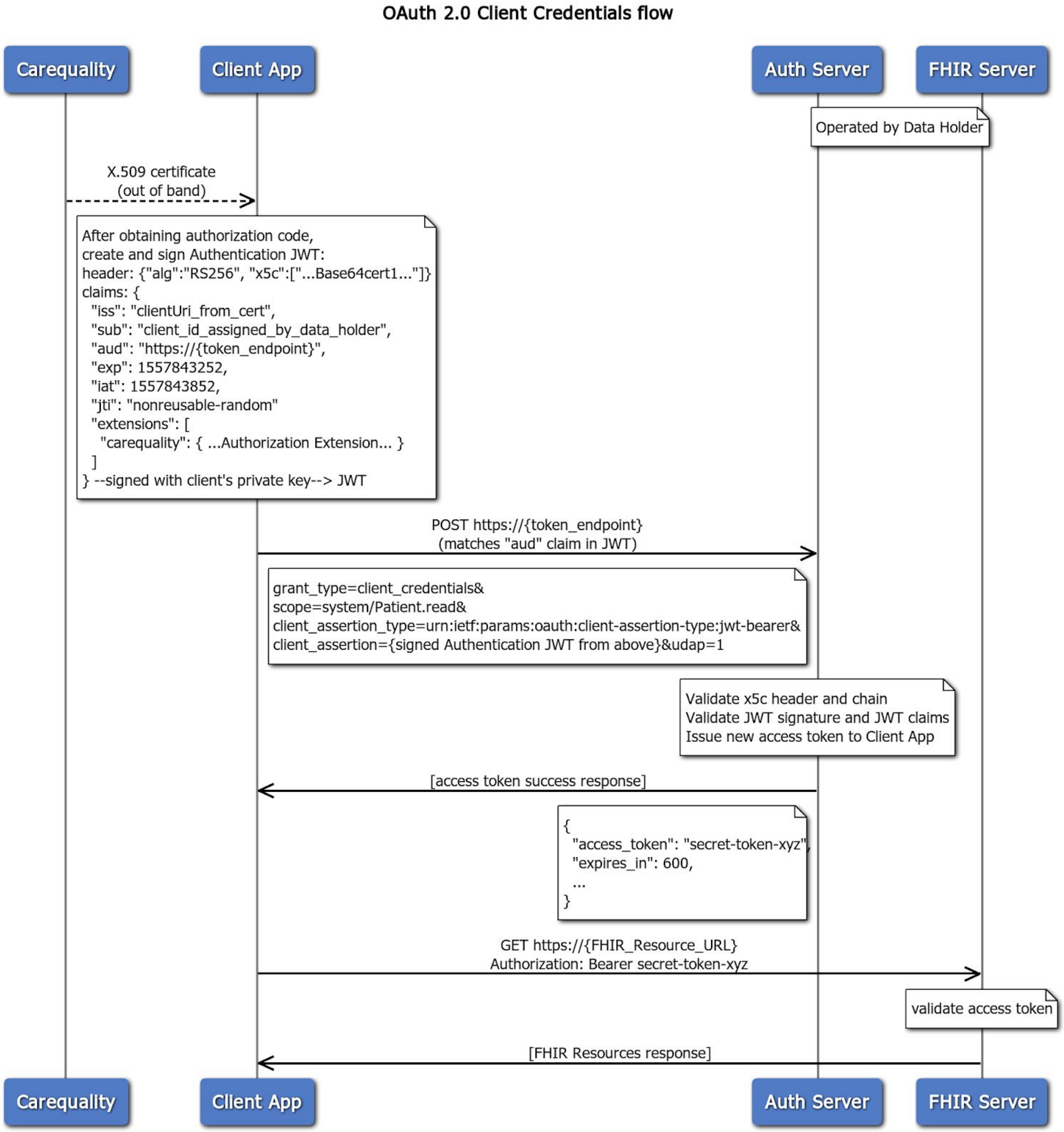
Request:

```
POST /token HTTP/1.1
Host: as.example.com
Content-type: application/x-www-form-urlencoded

grant_type=authorization_code&
  code=authz_code_from_resource_holder&
  client_assertion_type=urn:ietf:params:oauth:client-assertion-type:jwt-bearer&
  client_assertion=eyJh[…remainder of AnT omitted for brevity…]&
  udap=1
```

Response (success):

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "access_token": "example_access_token_issued_by_AS",
  "token_type": "Bearer",
  "expires_in": 3600
}
```

8.3.5 Client Credentials Grant Type (2-legged OAuth 2.0)


OAuth 2.0 Client Credentials flow
Carequality
Client App
Auth Server
FHIR Server
Operated by Data Holder
X.509 certificate
(out of band)
POST https://{token_endpoint}
(matches "aud" claim in JWT)
[access token success response]
GET https://{FHIR_Resource_URL}
Authorization: Bearer secret-token-xyz
validate access token
[FHIR Resources response]
Carequality
Client App
Auth Server
FHIR Server

```
After obtaining authorization code,
create and sign Authentication JWT:
header: {"alg":"RS256", "x5c":["...Base64cert1..."]}
claims: {
  "iss": "clientUri_from_cert",
  "sub": "client_id_assigned_by_data_holder",
  "aud": "https://{token_endpoint}",
  "exp": 1557843252,
  "iat": 1557843852,
  "jti": "nonreusable-random"
  "extensions": [
    "carequality": { ...Authorization Extension... }
  ]
} --signed with client's private key--> JWT
```

```
grant_type=client_credentials&
scope=system/Patient.read&
client_assertion_type=urn:ietf:params:oauth:client-assertion-type:jwt-bearer&
client_assertion={signed Authentication JWT from above}&udap=1
```

```
Validate x5c header and chain
Validate JWT signature and JWT claims
Issue new access token to Client App
```

```
{
  "access_token": "secret-token-xyz",
  "expires_in": 600,
  ...
}
```

Privileged applications acting on their own behalf, or without direct user action required via the data holder’s authorization endpoint, use the client_credentials grant type. In this flow, the authorization endpoint is not used and the application communicates directly to the token endpoint. Authorization

Servers SHALL NOT issue refresh tokens to clients using this grant type. Authorization Servers SHALL issue access tokens with a lifetime no longer than 60 minutes.

Since users do not interact directly with the data holder's authorization endpoint when using the client_credentials grant type, it may be necessary to provide additional authorization information to the data holder at the time of the token request by adding this information to the authentication JWT in the form of a Carequality-specific authorization extension as per section 5.2 of http://www.udap.org/udap-client-authorization-grants.html, and as detailed further below.

The authentication JWT submitted by the client app MUST conform to the header requirements in Section 8.3.2 and the claims requirements in the following table. Note that the header requirements are identical to those used for the authorization code flow:

Authentication JWT Claims		
iss	required	Issuer of the JWT -- unique identifying client URI. This MUST match the value of a uniformResourceIdentifier entry in the Subject Alternative Name extension of the client's certificate included in the 'x5c' JWT header See https://tools.ietf.org/html/rfc5280#section-4.2.1.6
sub	required	The service's client_id, as determined during registration with the FHIR authorization server
aud	required	The FHIR authorization server's "token endpoint URL" (the same URL to which this authentication JWT will be posted -- see below)
exp	required	Expiration time integer for this authentication JWT, expressed in seconds since the "Epoch" (1970-01-01T00:00:00Z UTC). This time SHALL be no more than five minutes after the time the JWT is issued
iat	required	Issued time integer for this authentication JWT, expressed in seconds since the "Epoch"
jti	required	A nonce string value that uniquely identifies this authentication JWT. This value SHALL NOT be reused by the client app in another

		authentication JWT before the time specified in the “exp” claim has passed
extensions	required	A JSON Object containing the key "carequality" with a value equal to a Carequality Authorization Extension Object, defined below. For Patient Requests, the key "carequality_user" SHALL be used instead, with a value equal to a Carequality User Authorization Extension Object, as discussed below

Carequality Authorization Extension object **Extension Name: “carequality”**		
version	required	Fixed string value: "1"
organization_id	required	String containing the URL of requestor’s Organization resource at the Carequality Directory server:https://prod-dir-ceq-01.sequoiaproject.org/fhir-stu3/1.0.1/ "https://https://prod-dir-ceq-01.sequoiaproject.org/fhir-stu3/1.0.1//Organization/2.16.840.1.113883.19.347473"
organization	required	String containing the requestor’s human readable organization name, e.g. "ABC Hospital"
subject_id	conditional	String containing the human readable name of the person responsible for originating the request. SHALL be present when applicable, e.g. "Dr. Mary Johnson"
purpose_of_use	required	String containing the purpose for which the data is requested, from the code set of permitted purposes in

		the NHIN PurposeOfUse code system as per Section 3.1 of the Carequality QBDE IG, e.g. TREATMENT | PAYMENT | OPERATIONS | PUBLICHEALTH | REQUEST | COVERAGE
acp	optional	The Access Consent Policy Identifier corresponding to the asserted Access Policy, array of string values from the list of valid policy OIDs in section 3.8 of this IG, each expressed as a URI, e.g. ["urn:oid:2.16.840.1.113883.3.7204.1.1.1.1.1"]
acp_reference	optional	An array of FHIR DocumentReference or Consent resources where the supporting access consent documentation can be retrieved, each expressed as an absolute URL, e.g. ["https://implementer1.example.com/fhir/R4/DocumentReference/consent-12345"]

When the Carequality Authorization Extension object is included in a token request and the data holder determines that the authorization metadata submitted is insufficient for the data holder to grant access because the data holder requires one or more Access Consent Policies to be asserted but the requestor has omitted the acp parameter or has asserted a policy that is not acceptable to the data holder, then the Authorization Server SHALL return an invalid_grant error response to the token request, and this error response SHOULD include the Carequality Authorization Extension Error object in the 'extensions' object of the error response.

Carequality Authorization Extension Error object **Extension Name: "carequality"**		
acp_required	required	The list of acceptable Access Consent Policy Identifier(s) corresponding to the asserted Access Policy required for authorization, an array of string

		values from the list of valid policy OIDs in section 3.8 of this IG, each expressed as a URI, e.g. ["urn:oid:2.16.840.1.113883.3.7204.1.1.1.1.1"]
acp_form	optional	A URL as a string where the required consent form may be downloaded, if applicable, e.g. "https://implementer1.example.com/consentForms/sample1.pdf"

Responders supporting use cases that require transmission of consent information SHALL support the acp and acp_reference claims and SHALL be able to resolve a DocumentReference or Consent resource included in the acp_reference array.

If the requested purpose of use is not supported by the responder, the responder SHALL return an invalid_grant error response to the requesting application.

Patient Requests

Responders MAY support the client credentials grant type for Patient Requests (i.e. where the purpose_of_use code is REQUEST) but are not required to do so. This corresponds to authorization workflow (2) defined in Section 3.2. If the responder does support this workflow, the responder SHALL support the Carequality User Authorization Extension object, defined in Section 8.3.4.2 and identified by the extension key "carequality_user".

A client application requesting a token for Patient Requests using the client credentials grant type SHALL include the Carequality User Authorization Extension Object in its token request instead of the Carequality Authorization Extension object. The user metadata submitted by the requesting application in the Carequality User extension object SHALL correspond to the verified identity attributes of the permitted user (verified as per Section 2.2) who is making the request. Note that this user is not necessarily the patient who is the transaction subject, i.e., the verified user MAY instead be a patient’s authorized representative. Before issuing an access token, the responder SHALL validate that the verified user identity metadata submitted by the application matches the responder’s own records for a person that is authorized to make patient requests in accordance with Section 3.2 and SHALL limit the patient data accessible using the access token accordingly. If the submitted user information does not sufficiently match a person known to the responder, or if the responder does not support this workflow for Patient Requests, it SHALL return an invalid_grant error in response to the token request.

Example:

Below is an example of complete authentication JWT header and claims with authorization information prior to Base64URL-encoding and signing (non-normative, the “.” between the header and claims objects is a convenience notation only):

```
{
  "alg": "RS256",
  "x5c": ["MIIEczCCA1ugA…remainder of Base64 encoded certificate omitted for brevity…"]
}.{
  "iss": "http://implementer1.example.com/cq-fhir-app",
  "sub": "myClientID",
  "aud": "https://implementer2.example.net/token",
  "exp": 1557843252,
  "iat": 1557843852,
  "jti": "Q1E6g2PY91nmj5bSJJ-CZQ",
  "extensions": {
    "carequality": {
      "version": "1",
      "organization_id": "https://directory.carequality.org/Organization/2.16.840.1.113883.19.347473",
      "organization": "ABC Hospital",
      "subject_id": "Dr. Mary Johnson",
      "purpose_of_use": "TREATMENT",
      "acp": ["urn:oid:2.16.840.1.113883.3.7204.1.1.1.1.5"],
      "acp_reference":
["https://implementer1.example.com/fhir/R4/DocumentReference/consent-12345"]
     }
  }
}
```

After generating an authentication JWT, the client requests a new access token via HTTP POST to the FHIR authorization server's token endpoint URL, using content-type application/x-www-form-urlencoded with the following parameters:

POST Body Parameters		
grant_type	required	Fixed value: client_credentials
scope	required	The scope(s) of access requested, expressed as a space delimited list of SMART clinical scopes

client_assertion_type	required	Fixed value: urn:ietf:params:oauth:client-assertion-type:jwt-bearer
client_assertion	required	The signed authentication JWT value (see above)
udap	required	Fixed value: 1

Examples (non-normative, white space, and line breaks added for clarity, not URL-encoded):

Request:

```
POST /token HTTP/1.1
Host: as.example.com
Content-type: application/x-www-form-urlencoded

grant_type=client_credentials&
scope=system/Patient.read&
client_assertion_type=urn:ietf:params:  :client-assertion-type:jwt-bearer&
client_assertion=eyJh[…remainder omitted for brevity…]&
udap=1
```

Response (success):

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "access_token": "example_access_token_issued_by_AS",
  "token_type": "Bearer",
  "expires_in": 3600
}
```

Response (failure, Authorization Server requires a specific Access Consent policy to be asserted):

```
HTTP/1.1 400 Bad Request
Content-Type: application/json


{
  "error": "invalid_grant",
  "error_description": "An Access Consent policy must be asserted for this purpose of use.",
  "extensions": {
    "carequality" : {
      "acp_required": ["urn:oid:2.16.840.1.113883.3.7204.1.1.1.1.1"]
    }
  }
}
```

8.4 Record Location Service (RLS)

Record Location Services are out of scope for Version 1 of this document. Future versions may address this issue.

8.5 Clinical Data Exchange

8.5.1 FHIR Version and FHIR Implementation Guide Support Requirements

For participation in the Carequality FHIR information exchange, Carequality FHIR implementers MUST support FHIR US Core Implementation Guide V3.1.0 where data is available (e.g., US Core Pediatric BMI for Age Observation Profile need not be supported if the information is not collected) and MAY support subsequent version(s). In addition, FHIR Implementation Guides MUST be supported to the requirement levels specified in this link.

8.5.2 Patient Discovery

Except for the SMART on FHIR auth code flow in which the `launch/patient` SMART scope is requested, granted and the Patient ID is subsequently provided, Patient Discovery SHALL be performed using the FHIR Patient Resource $match operation. Each query SHALL include, but is not limited to, all available USCDI patient demographics with a minimum of (where known): first name, last name, date of birth, birth sex, current address (normalized as per section 3.3.2), phone number(s), and email address(es) plus administrative gender. All implementers SHALL support these demographics. A responder MAY ignore any other demographic not supported.

The $match request operation SHALL have the following parameters:

Parameter	Required Value
onlyCertainMatches	SHALL be set to true for Patient Requests; when set to true, the server SHALL return only certain matches; when absent or set to false, the server MAY return probable matches, but is not required to do so if its organizational policy allows only certain matches to be returned
count	Optional, server MAY send fewer results than specified Note that clients should be careful when using this, as it may prevent

	probable - and valid - matches from being returned
resource	Patient resource with included demographic parameters formatted as per US Core Patient profile