Query-Based Document Exchange Implementation Guide

Version 2.0

Effective July 12, 2021

Table of Contents

1.0 Introduction

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

The Query-Based Document Exchange Use Case addresses the need for documents containing relevant healthcare information to be available upon request 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 multiple query purposes and healthcare settings. Queries for treatment purposes have some additional requirements, but widespread exchange over a number of permitted purposes is envisioned.

In order to facilitate such widespread exchange, with a very large number of potential exchange partners, record location services will likely play an important role. It will not be practical for an end user, or even a system through an automated process, to query all of the accessible organizations to determine which of them may have information about a patient. Record locator services can pinpoint specific targets for queries. To maintain flexibility, however, a record locator service is not assumed or required.

As noted above, this Guide covers technical specifications as well as policy and process requirements. Sections 2 through 6 outline the policy and process requirements, while Sections 7 and 8 outline technical specifications.

2.0 Definition of Roles

The concept of a role within the 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 in Sections 3 through 6 of this Guide 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.

2.1. Query Initiator

An Implementer or CC with the declared role of a Query Initiator performs queries to retrieve information held by Implementers or CCs in the Query Responder role. These queries may or may not be facilitated by an Implementer in the Record Locator Service role.

An Implementer or CC with the declared role of a Query Initiator shall support the technical actor(s) specified in Section 8.1.1 of this Guide, and comply with any other requirements throughout this Guide that are specifically described as applying to the Query Initiator 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.

Query Responders do not have direct interaction with Implementers in the Record Locator Service role, within the context of activities subject to the requirements of this Implementation Guide. Query Responders may have relationships with Implementers in the Record Locator Service role to, for example, provide data used by the Record Locator Service in the provision of its service to Query Initiators, but such a relationship is outside the scope of this Carequality Use Case and is not subject to this Implementation Guide.

An Implementer or CC with the declared role of a Query Responder shall support the technical actor(s) specified in Section 8.1.2 of this Guide, and comply with any other requirements throughout this Guide that are specifically described as applying to the Query Responder role.

2.3. Record Locator Service (RLS)

An Implementer or CC with the declared role of an RLS provides, in response to queries by Implementers or CCs in the Query Initiator role, a list of Implementers and/or CCs in the Query Responder role who potentially have, likely have, or are known to have clinical documents for the person who is the subject of the query.

The role of an RLS does not include the storage and maintenance of these clinical documents beyond transitory storage of the retrieved data during the actual processing of the transaction, to the extent that such transitory storage is necessary to affect the transmission of the retrieved data to the requester. However, note that an Implementer or CC in the RLS role may also serve in other roles defined in this implementation guide where such storage and maintenance is permitted.

An Implementer in the RLS role may have CCs in other roles, even if the Implementer itself only plays the RLS role. Any such CCs in the Query Responder role must be available to be queried directly, through the transactions supported by the Query Responder role, without the use of any RLS being required. Similarly, an Implementer or CC that has itself declared both the RLS and Query Responder roles must accept queries in its role as a Query Responder from Implementers and CCs in the Query Initiator role who have chosen not to take advantage of the Implementer's or CC's RLS function.

An Implementer or CC with the declared role of an RLS shall support the technical actor(s) specified in Section 8.1.3 of this Guide, and comply with any other requirements throughout this Guide that are specifically described as applying to the RLS role.

3.0 Customizable Principles of Trust

3.1. Permitted Purposes

Carequality Implementers and CCs represent a diverse set of stakeholders that wish to exchange health information for a variety of reasons. It is important to building trust that a common set of reasons to initiate a query for information (Permitted Purposes) be agreed to by all Implementers of this Use Case, and their CCs. The Permitted Purposes for queries to be made under this Use Case are:

- Treatment
- Payment
- Health Care Operations
- Public Health Activities
- Patient Request
- Coverage Determination
- 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.

Implementers and CCs that initiate queries for the Permitted Purpose of Patient Request ("Patient Requesters") MUST provide their users with a clear description of how the user's data is used by the Patient Requester. This description must be an accurate representation of any data use permitted by the terms and conditions to which the user agrees, in order to use the Patient Requester's personal health record or other consumer-facing application. While not specifically requiring compliance with the

current version of the CARIN Alliance Code of Conduct (the "Code of Conduct"), compliance with the Code of Conduct would fulfill this requirement.

An Implementer or CC who 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 Query Use Case. Therefore, each Implementer shall identify to Carequality the Permitted Purposes that it and each of its CCs support.

When an Implementer or CC initiates a query for information, it shall clearly identify the specific Permitted Purpose for the query in the SAML token for the message, according to the NHIN Authorization Framework 3.0 specification, section 3.2.2.6, Purpose Of Use Attribute, as referenced in Section 8.4.2 of this Guide. By asserting a Permitted Purpose, an Implementer or CC certifies that the context of its request meets the requirements for the stated Permitted Purpose as defined above.

Note that the Permitted Purposes allowed for Carequality are a subset of those defined in the NHIN Authorization Framework, with the caveat that Other Authorization-Based Disclosures provides some additional flexibility. See the table below for additional information on the Other Authorization-Based Disclosures permitted purpose. The specific NHIN PurposeOfUse values that may be used to represent the Carequality permitted purposes are as follows:

Carequality Permitted Purpose of Use	**NHIN PurposeOfUse code**
Treatment	TREATMENT
Payment	PAYMENT
Health Care Operations	OPERATIONS
Public Health Activities	PUBLICHEALTH
Patient Request	REQUEST
Coverage Determination	COVERAGE
Other Authorization-Based Disclosures	The Implementer or CC may use any NHIN PurposeOfUse code that is NOT otherwise listed in this table and is not prohibited in the following paragraph, and that the Implementer or CC in good faith believes is the best available representation of the transaction's actual

	purpose. It is acknowledged that the available PurposeOfUse codes may not include a clearly obvious value for every transaction, and Carequality anticipates future work to more clearly define specific values. NHIN PurposeOfUse codes are defined by the NHIN Authorization Framework 3.0 specification, section 3.2.2.6, as referenced in Section 8.4.2 of this Guide. Notwithstanding the previous paragraph, the following PurposeOfUse codes MUST NOT be used for Carequality: PRESENT, EMERGENCY, DISASTER.

Note that the PurposeOfUse codes defined by the NHIN Authorization Framework encompass two separate concepts – the immediate use to which the information released will be put, and other attributes of the request that may impact the responder's access policies. Carequality divides these two concepts into the Permitted Purpose, and Access Policy Assertions (the latter fully described below in Section 4.4). For example, Carequality has defined a Policy Assertion to indicate when a request is being made in an emergency situation. The information released in such a case is most likely going to be used for treatment, so in Carequality's defined structure the PurposeOfUse is Treatment, with a Policy Assertion of Emergency, potentially among others that may also apply.

3.2. 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, is strongly encouraged to provide information in response to valid queries for 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 for treatment.

3.2.1. Treatment

An Implementer or CC is permitted to serve ONLY in the role of Query Initiator for the Permitted Purpose of treatment 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 3.2.1.1 below, or is an EMS provider with alternative provision of data, as defined in Section 3.2.1.2 below. An Implementer or CC, other than a government agency or those defined below in subsections of this Section 3.2.1, who wishes to be a Query Initiator for treatment purposes must also play the role of Query Responder for treatment 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-CarequalityType) 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-CarequalityType values, so that the Carequality community can immediately discern which organizations are claiming an exception.

3.2.1.1. 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.

3.2.1.2. 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.

3.3. Permitted Users

No specific Permitted Users have been defined for this Use Case at this time. Carequality does not want to create restrictions on Implementers with respect to the querying workflow in their organizations, and

those of their CCs, for how they accomplish one of the Permitted Purposes. While not required, Carequality recommends that Implementers design their workflows such that data for a specific patient is retrieved ahead of any treatment encounter with a clinician.

3.4. Data Sufficiency and 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 2021, the industry has not yet universally adopted and implemented consistent document content standards. The clear goal of Carequality is to make progress toward greater consistency and quality in document content over time.

To this end, Carequality enforces requirements related to document content. Carequality will utilize one or more approved testing programs to validate that specific categories of organizations defined by the testing program are able to produce content that conforms with Carequality's requirements.

The Content Testing Program will define which categories of Implementer and/or their CCs are required to submit to content testing. Implementers or CCs that fall into one or more of these categories are required to test within the timelines defined by the program. Failure to submit to testing or to comply with test result recommendations may result in an organization's removal from live exchange activities. Generally, and independent of the specific document format used, Query Responders MUST ensure that documents returned in response to queries provide an accurate representation of the information contained in the responding system at the time of the documents' generation. In instances where a Responder has limited technical capacity and can only reply with an unstructured CDA or a PDF, it is preferred that the response come in the form of a PDF or other plain document (i.e., jpeg, etc.) with the appropriate format code and MIME type.

3.5. Service Level Agreements

No Service Level Agreements (SLAs) have been identified for this Use Case at this time. Carequality will collect information from Implementers about system uptime, endpoint availability, and response time. This information will be used to determine what, if any, SLAs should be developed.

3.6. Customizable Flow-downs

No additional customizable flow-downs have been identified for this Use Case.

4.0 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 a Use Case 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. Treatment

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 or responses for treatment purposes. No additional agreements beyond the Carequality legal framework may be required. 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).

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 and Children First are both querying for Treatment, Non-Discrimination requires that these two practices should have equal access to Mr. Smith's information. Mr. Smith may authorize release to a specific practice, but Peaceful Valley may not have an overall policy that treats the two practices differently.

4.2. Other Permitted Purposes

Implementers and CCs are permitted, but not required, to impose fees, terms and conditions on other Implementers or CCs with respect to queries or responses for any permitted purpose other than treatment. Any fees, terms and conditions must comply with Section 4.3 of this Implementation Guide.

Implementers or CCs that play the role of Query Responder are not required to honor queries for non-treatment permitted purposes. However, Implementers or CCs that are Query Responders may choose to honor queries for other permitted purposes.

The content provided in response to queries for non-Treatment Permitted Purposes may be the same content released in response to Treatment queries, though Implementers and CCs may also respond with an information set that is specific to a query's Permitted Purpose. Implementers and CCs may also choose to do so in order to comply with that Implementer or CC's Applicant Business Rules or Organization Business Rules, and/or Applicable Law.

Additionally, Implementers responding to non-Treatment queries may apply patient matching logic that differs from those used in response to Treatment. Implementers and CCs in the Query Responder role MAY utilize separate patient matching logic for queries with different Permitted Purposes, for example, requirements that are more stringent and/or rely on exact matches for certain fields when responding to queries for Patient Request versus Treatment.

Query Responders MAY decline to honor queries for the Permitted Purposes of Payment or Health Care Operations for those patients who have received self-pay care, although Query Responders are encouraged to respond with those portions of the record that don't relate to the self-pay care.

If a Query Responder does choose to honor queries for a non-treatment purpose, it must honor 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.

Note, Carequality anticipates further work to more fully define the Other Authorization-Based Disclosures permitted purpose. Until such additional definition is completed, however, Query Initiators may, in good faith, make queries using the same PurposeOfUse value that in fact stem from very different circumstances. Given this uncertainty, Query Responders are given broad leeway 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. 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 other than treatment, such terms and conditions cannot 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 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". As long as a Query Responder's terms and conditions focus on a particular workflow as elucidated by the examples above – although not limited to the examples above – and do not exclude a particular organization or organization type that engages in the relevant workflow, such terms and conditions are acceptable under these Non-Discrimination requirements.

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. Access and Patient Permission

This Section outlines requirements for Implementers and CCs who wish to communicate access policy requirements and their fulfillment within query and response transactions for this Use Case, as described in Section 8.0. 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. 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)).

Unlike Section 4.3, this Access and Patient Permission 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, Peaceful Valley Hospital will release patient files only if they receive 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.

4.4.1. Access Policy Assertions

In addition to asserting a Permitted Purpose, Implementers and CCs 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. A Query Initiator must assert an Access Policy Assertion by including the unique Access Consent Policy Identifier listed for the Assertion in the table below in the SAML token of a Carequality message, as described in section 8.2, flows "Initiating Gateway asserts…", if the Query Initiator meets the requirements for that Access Policy Assertion that are also outlined in the table.

Access Policy assertions are intended to provide Implementers and CCs additional flexibility in their access policies. An Initiator might assert that their Permitted Purpose of Use is "Treatment" but these options allow the Query Responder to make a distinction within those "Treatment" based requests. An example being the difference between those requests that have corresponding signed release forms from those that do not. 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 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. In such cases, as more fully outlined in Section 8.2, flows "Responding Gateway retrieves Patient Permission document…", the Query Initiator shall include the document unique ID within the Instance Access Consent Policy (IACP) element of the SAML token for its request.

Query Initiators are strongly encouraged to support the inclusion of Policy Assertions in messages as soon as possible. Carequality acknowledges, however, that a transition period will occur between the time when requirements of this Implementation Guide are published and the time when all Query Initiators have support for the inclusion of Policy Assertions in messages. During this transition period, Carequality will provide a field within the Carequality Directory entries for Query Initiators that will indicate whether or not that Query Initiator has the ability to support the Policy Assertion structure. All statements in this Implementation Guide referring to 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. Carequality is not, at this time, imposing a timeline within which all Query Initiators must support the inclusion of Policy Assertions in messages, but may do so in the future.

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.

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 as outlined in section 8.2, flows "Responding Gateway retrieves

		Patient Permission document…". 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 within the scope of the transactions outlined in Sections 7 and 8 of this Implementation Guide, 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 as outlined in section 8.2, flows "Responding Gateway retrieves Patient Permission document…". 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. The Query Initiator does not support a mechanism for retrieving an electronic copy of the

		Patient Permission form within the scope of the transactions outlined in Sections 7 and 8 of this Implementation Guide, 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 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 using the transactions outlined in Sections 7 and 8 of this Implementation Guide, 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	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

(**Available** for electronic request within 10 days)		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 using the transactions outlined in Sections 7 and 8 of this Implementation Guide, 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'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. The Query Initiator must also be able to parse and interpret information contained in document metadata identifying a document as containing substance abuse treatment information as described in section 8.7.8

In the case of any of the Policy Assertions involving a signed Patient Permission form, the Query Initiator is responsible for the thorough and accurate documentation of signatures and for the preservation of the form. Query Initiators must not assert policies related to having a signed form unless that form will remain valid, from the standpoint of an expiration date and time, for at least 24 hours after the assertion is made. Note, having the form explicitly revoked by the patient within 24 hours does not constitute a failure to meet this requirement. If the Query Initiator is unsure of a Patient Permission form's expiration date, it should not assume that a signed Patient Permission form is valid and therefore should not assert any Policy Assertions based on that form.

For example, suppose that Peaceful Valley Hospital has a signed Patient Permission form from John Smith's personal representative, Jane Doe. When initiating any query to Adventist Medical, Peaceful Valley Hospital is responsible for assuring that Jane Doe is, in fact, an authorized representative of Mr. Smith. Additionally, when Peaceful Valley Hospital asserts that it has a signed Patient Permission form, it must also maintain a record of the expiration date for that document. If Peaceful Valley cannot determine an expiration date at a system level, it should not make the assertion. Adventist Medical should not consider any Patient Permission form assertion from Peaceful Valley to take precedence over the patient's decision to opt out of releasing records to Peaceful Valley.

Queries that assert NIST IAL 2 or 3 from a third party such as a PHR or a wearable device should be regarded as queries made directly by the patient. Unlike Patient Permission form assertions, IAL assertions do not require the same expiration dates restrictions.

For example, if Jane Smith requests her records from Peaceful Valley Hospital through a PHR that has verified her identity to NIST IAL 2 or 3, this request as regarded as being made directly from the patient and therefore requires no expiration date. This extends to PHR and medical/consumer hardware as well as software/systems that may be set by the user to make periodic requests for and/or to transmit data. The PHR or device in this case is a mechanism to make and receive the request that the patient themselves operates. This differs from the requests made by an insurer or other third party, which while made on the patient's behalf, are not a direct request by the patient and the patient is not the direct recipient of any of the data gathered.

4.4.2. Requirements for Query Responders

Query Responders are permitted to make access denial decisions based on the Initiator's Permitted Purpose as well as by the Access Policy Assertion they assert. 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, using the technical approach described in section 8.4.5, Reporting Access Denials, using the QualifyingPolicies element.

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. Note that Query Initiators are under no obligation to attempt to comply with the requirements for the Query Responder's indicated Policy Assertion(s), or to attempt a follow-up request asserting such Policy Assertion(s). See Section 4.4.3 for more details on Error Responses for Access Denials.

4.4.2.1. Evaluating Policies Prior to Responding to Patient Discovery Queries

As long as the Query Responder supports a particular query's Permitted Purpose, i.e. in Outcomes #2-4 of the potential outcomes listed at the beginning of this Section 4.4, Query Responders must perform patient matching based on a Patient Discovery 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.

4.4.2.2. Patient Discovery Queries and Revealing the Existence of Records

Absent specific permission, 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. 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.

4.4.2.3. Unsolicited or Unsupported Assertions

Query Responders must be prepared to receive any Carequality Access Policy Assertions in such a way that does not negatively impact their system or workflow. This includes those policy assertions that are not utilized by the Query Responder. In these instances, Carequality Access Policy Assertions that are not relevant to the Implementer's access policy must simply be ignored by the Implementer.

With respect to unsolicited Policy Assertions from the Query Initiator, Query Responders are not required to consider them sufficient to satisfy local access policies.

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. Adventist Medical (Query Initiator) asserts that they have satisfied the requirements of the Verbal Consent policy. Peaceful Valley does not utilize Verbal Consent as a factor within their access policy decisions. Receiving this assertion must not negatively impact Peaceful Valley's system or workflow; ideally it should simply be ignored.

4.4.2.4. Reliance on Prior Policy Assertions

Query Responders must not rely on Carequality Access Policy Assertions previously asserted by the Query Initiator, i.e. should not "cache" policy assertions. Query Initiators are required to assert any Access Policy Assertions for which they meet the requirements, with each transaction, and Query Responders should assume that if a Policy Assertion is not present in a transaction, it does not apply.

4.4.2.5. Non-Discrimination With Respect to 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. Note that this requirement specifically applies to assertions made in the Access Consent Policy (ACP) field of the SAML token, as described in section 8.2, flow "Initiating Gateway

asserts Access Consent Policy". Assertions made in the Instance Access Consent Policy (IACP) field of the SAML token are outside the scope of this requirement.

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.

4.4.2.6. Non-Discrimination With Respect to 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 to the outcomes of evaluating that access policy. 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.

For example, suppose that Peaceful Valley Hospital has a record for John Smith. Peaceful Valley Hospital has been queried for this record by radiology practice Adventist Radiology and Children First Orthopedics. If Adventist and Children First are both querying for Treatment, Non-Discrimination requires that these two practices should have equal access to Mr. Smith's information. While Mr. Smith may authorize release to a specific practice, Peaceful Valley may not have an overall policy that treats the two organization types (Ex. radiology) differently.

It may be the case, however, that a Query Responder has an understanding – formal or informal – with a specific Query Initiator such that internal processes and workflows will result in access policy requirements being met for that Query Initiator. For example, Peaceful Valley Hospital and Children First may have developed a shared intake form for all patients that provides permission for the free release of records between the two organizations, that is collected from all patients upon initial registration. Despite this arrangement, if Peaceful Valley's access policy requires Query Initiators to assert that a patient has provided consent, they must apply this standard to all Query Initiators equally.

Notwithstanding the previous requirement, Implementers or CCs that comprise the same business entity, for example a health system that uses two electronic health record systems that are connected via the Carequality Framework, may enforce different access policy requirements for responses to internal queries as opposed to those queries from external entities. Queries between two agencies of the federal government, or two agencies of a particular state government, also fall under this exception,

with queries between government agencies being considered “internal” for purposes of this requirement.

4.4.2.7. Policies Relating to Individual Users and Implications for Patient Restrictions

Query Responders are also prohibited from restricting access based on the role (occupation, title, etc.) of the individual user initiating a request. Conclusions about the individuals who ultimately will have access to see and use information that is released, cannot reasonably be made in many cases based on the individual associated with a request. It is commonplace for non-clinical staff or the system itself to initiate requests so that information is available to actual clinical users. Conversely, even if a clinical user is associated with the request, the Query Responder cannot be certain that other users won’t have access to the data in the requesting system once it is released.

Similarly, Query Responders must not base access policy decisions on the User Authentication Context Field within the SAML token for an inbound message. The accuracy and consistency of this value is currently questionable in practice. Carequality may permit the use of this field for access policy decisions in the future, if its use becomes more consistent across implementations.

Given these limitations on the access restrictions that can be supported within the Carequality Framework, the practical outcome is that some patient requests for restrictions on releases must be regarded as an opt-out by the patient with respect to exchange via the Carequality Framework. Note that organizations can choose to honor patient requests regarding which individual organizations that may or may not receive their information as well as the Carequality permitted purpose(s) for which their information may be released.

The Query Responder should ensure that any permissions received from the patient or representative accurately reflect the requesting organization as identified by the Home Community Identifier in the query transaction. Query Responders should assume that any information released in response to a query asserting that the Query Initiator can accept information from a facility covered under 42 CFR Part 2 may be accessed by users within the entire requesting entity identified in the Carequality Directory by the Home Community Identifier used in the query transactions. Information related to treatment in a facility covered under 42 CFR Part 2 should not be released if permission has not been given for the entire querying entity.

4.4.3. Error Responses for Access Denials

Section 8.4.5, Reporting Access Denials, outlines possible error responses that Query Responders may employ when responding to an incoming query for which access has been denied in whole or in part. In instances in which error responses are appropriate, Query Responders must err on the side of providing the maximum information possible about the source of the error, while also limiting potential disclosures of patient data. While the most detailed available response is encouraged, Query Responders are not required to include detailed information in their error responses and may respond with an error code indicating no matching patient was found, even if a patient match was in fact found,

if the Query Responder is unable to release any information about that patient to the Query Initiator, including even the fact that the Query Responder has a record for that patient (see section 7.2.7 for more details).

For example, suppose that Peaceful Valley Hospital has a record for John Smith. Peaceful Valley Hospital has been queried for this record by Adventist Medical. Peaceful Valley first performs a patient match based on the details provided by Adventist. Mr. Smith's record is found, however, due to Peaceful Valley's policies, patient records may not be shared without a signed document policy assertion. In this instance, Peaceful Valley is encouraged to provide as much detailed information in their access denial response as possible to inform Adventist staff about what additional documentation is required. While relaying the most detailed information available to Adventist is ideal, Peaceful Valley may (as a matter of policy) respond with "no match found" whenever a query does not satisfy all access requirements.

4.5. Record Locator Services

A Record Locator Service provides a value-added service that makes querying for records easier and more efficient, but is not required in order to obtain records since the record holder can be queried directly. A Record Locator Service provides the locations of patient records, but does not provide the records themselves or the clinical data they contain, which are requested from an Implementer or CC in the Query Responder role based on the locations reported by the Record Locator Service.

A Record Locator Service for purposes of the Query-Based Document Exchange Use Case is narrowly defined in Section 2.3, and is distinguished primarily by being a Responding Gateway actor for the ITI-56 Patient Location Query transaction. Full details may be found in Section 8.1.3 below.

An Implementer or CC that is a Record Locator Service may honor patient location queries selectively based on additional agreements and charge a fee, including for patient location queries that are for treatment.

5.0 Performance Measures

In order to gauge Carequality's success in advancing widespread interoperability, Carequality will collect information from Implementers on a number of performance measures. These measures are meant to measure the impact of Carequality and specifically of this Use Case, not to evaluate individual Implementers, and the measures themselves will have no impact on an Implementer's Carequality Connected status.

Carequality will request, on a periodic basis, that Implementers provide a report on the measures outlined in this section. Implementers are required to respond for each measure with:

1. Information for that measure that is correct to the best of the Implementers' knowledge,
2. An attestation that the particular measure does not apply to that Implementer, or

3. An attestation that the Implementer cannot discover the information for that measure through commercially reasonable efforts.

5.1. Acceleration

This category addresses Carequality's effectiveness in accelerating the process of establishing connections. In this category, Carequality will have a single measure: Time in days from an Implementer's signing of the Carequality Connected Agreement, to production go-live by that Implementer or at least one CC, in at least one role specified for this Use Case.

Since the information needed for this measure will already be available to Carequality, no reporting from Implementers is necessary. This measure is included here simply for completeness.

5.2. Seamless Connectivity

This category addresses Carequality's effectiveness in broadening the scope of connectivity. There are several measures in this category, encompassing the breadth and scale of Implementers' connectivity as well as the adoption of that connectivity.

5.2.1. Breadth and Scale

1. Number of end users in production sharing information through the Implementer's network, service, or operations.
2. Types of member organizations or facilities making up the Implementer's network, or using its services. Note that these members do not all have to be CCs to be reported here, as long as they are able to take advantage of the Implementer's Carequality Connected status. (For example, hospitals, clinics, mental health centers, long term care centers, etc.)
3. Geographic areas represented by those member organizations.
4. Number of unique end users connected through the Implementer's network, service, or operations.

5.2.2. Adoption and Volume

1. Annual number of document queries performed through the Implementer's network, service, or operations.
2. If applicable: number of unique individuals included in the Implementer's master person index.

6.0 Evidence of Compliance

Applicants wishing to become Implementers of this Use Case must show evidence that they are able to comply with the requirements of the Use Case. Implementers are subject to the testing and connectivity policies listed in this guide until a transaction testing process is described by Carequality. Implementers are required to follow any and all requirements mandated by the transaction testing program at that time.

Generally, applicant requirements fall broadly into two categories:

1. The Carequality Application Process as defined for all Implementers and CCs, regardless of Use Case.
2. Compliance of the Implementer's system(s) with the technical specifications of the role or roles that it or its CCs will play, or in the case of ongoing connectivity verification, do play.

6.1. Application Process

This Guide does not add any requirements or additional steps beyond the Carequality Application Process defined for all Implementers and CCs and enforced by the Carequality Connected Agreement.

6.2. Technical Testing and Ongoing Verification

This section outlines the steps that Implementers must take in order to provide confidence that their network can connect to those of other Implementers using the technical specifications for this Use Case. The primary focus of technical testing for Carequality is on production system connectivity. Sections 6.2.2 through 6.2.5 apply to Implementers declaring the Query Initiator and/or Query Responder role, either for themselves or their CCs. These sections do not apply to those Implementers who are only in the Record Locator Service role, although such Implementers are encouraged to perform similar tests with those who will use their service.

When considering this connectivity validation approach, it is necessary to distinguish between two important but separate goals.

1) Providing reasonable confidence in the overall ability of a network to connect to others via the specifications for this Use Case.
2) Maintaining surveillance of connectivity for individual participants at all levels, including CCs.

The latter is an important topic, but is not the subject of this process, which is intended only to provide reasonable confidence in an Implementer's own systems as well as its network of CCs taken as a whole.

Nonetheless, Implementers do have a responsibility to validate that their CCs are consistently able to connect with other Implementers and CCs. It is unreasonable to expect that every CC will be accessible at all times to every other Implementer and CC, but if a CC is consistently inaccessible to other Implementers and/or their CCs, the Implementer must work with that CC to resolve its connectivity or suspend its status as a CC.

If an Implementer or CC is persistently inaccessible, but does not voluntarily suspend its status as an Implementer or CC, and another Implementer believes that productive efforts are not being made to resolve the connectivity, a dispute may be filed under the Carequality Dispute Resolution Process.

In instances where connectivity previously existed, but is no longer functioning properly, Implementers are required to take steps to reestablish connectivity. A joint troubleshooting session must be scheduled to develop a mutually agreed upon plan to resolve the issue(s). For example, a plan will contain a deadline to complete any development required to reestablish data exchange. Failure to comply with the mutually agreed upon development plan may result in the escalation of the matter to the formal Dispute Resolution Process.

In order to facilitate communications between Implementers prior to a formal dispute, Carequality staff may be contacted to mediate inter-Implementer discussions. This intervention is also appropriate to use when a test Query Initiator finds that a Query Responder's Non-Production gateway is not operational and their staff is not responsive.

The testing and connectivity validation approach outlined in Sections 6.2.2 through 6.2.5 relies on Implementers serving as testing and validation partners for other Implementers. All Implementers who play, or support CCs who play, the Query Initiator and/or Query Responder roles have an obligation to serve as testing and validation partners at the request of other Implementers or Carequality on behalf of other Implementers. Implementers are strongly encouraged to coordinate with one another to distribute the effort of serving as testing and validation partners among the community of Implementers.

6.2.1. Assertion of Compliance

By declaring the intent for itself and/or its CCs to play a role or roles in this Use Case, and beginning the Technical Testing process outlined in Sections 6.2.2 and 6.2.3, an Implementer asserts that the system or systems used to play the declared role or roles are compliant with the technical specifications for the declared role or roles, as outlined in Sections 7.0 and 8.0 of this Guide.

Implementers are encouraged to take advantage of testing opportunities such as tools provided by the National Institute of Standards and Technology (NIST), testing platforms maintained by private organizations, and Integrating the Healthcare Enterprise (IHE) Connectathon events.

6.2.2. Transaction Testing Program

When available, this process will be described as part of a transaction testing program.

6.2.3. Non-Production Partner Test

Prior to implementing production connectivity via the transactions specified for this Use Case, each Implementer will complete a series of non-production tests. As of the publication of this version XX.YY (fill in correct version) of this Guide, Carequality intends to facilitate an ecosystem of interconnected, non-production instances of Implementer and, where applicable, CC gateways ("Test Ecosystem"). The Test Ecosystem will include a Carequality Directory instance with which Implementers and CCs can interact to obtain and update test gateway information. Test Ecosystem gateways and transactions will be secured using valid, non-production Carequality certificates. The Test Ecosystem, once available, will greatly facilitate non-production testing for Carequality Implementers and, where applicable, CCs.

Until such time that the Test Ecosystem is available, however, Implementers are responsible for conducting partner tests with three other Implementers whose connectivity relies on software provided by a different technology vendor or provider (each of which is a "Test Partner").

The non-production partner tests consist of successful execution of each transaction required for the role(s) declared by the Implementer as being played either directly by that Implementer, or by its CCs.

Prior to the availability of a test ecosystem, the success of the test will be at the discretion of the Test Partner, but Test Partners MUST NOT report success unless each transaction has been completed and

data returned to the other party in that transaction. Specifically, matching patients must be found, at least one document must be available, and one or more documents must be retrieved. Data should be coordinated among the test partners such that patient matching is successful.

When the Test Ecosystem is available, Implementers are required to achieve 75% transaction success in the Query Initiator Role with 50% of the Live Implementers participating in the Test Ecosystem, rounding up to the nearest integer when necessary. Specifically, matching patients must be found, at least one document must be available, and one or more documents must be retrieved. For example, if there are 20 Live Implementer gateways in the Test Ecosystem, an onboarding Implementer playing the Query Initiator role must test with 10 of them and achieve success with no fewer than 8 of them.

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. To the extent that the Test Ecosystem includes gateways for CCs, Implementers are encouraged to test separately with them. For example, suppose Acme Connectivity Services is an Implementer providing Carequality connectivity to the customers of two EHR vendors, Precision Health and Ultimate Health. Acme Connectivity Services would be strongly encouraged to maintain at least one CC in the Test Ecosystem that reasonably represents the production behavior of Precision Health clients, and another CC that reasonably represents the production behavior of Ultimate Health clients. Onboarding Implementers SHOULD complete a successful test with each such CC, to the extent reasonably possible.

Implementers testing in the Query Responder role, or who support CCs in the Query Responder role, must receive Patient Discovery transactions from at least 50% of Live Implementers, and must respond successfully with a "No Matching Patient Found" response for at least 75% of these transactions. Such a response is "successful" if it is received and processed without error by the querying system.

Upon completion of the necessary test transactions for either the Query Initiator or Query Responder role, the onboarding Implementer must provide to Carequality a list of the partner Implementers involved and the outcome of the query, namely, (1) "No Matching Patient Found", (2) an error, or (3) no response. Carequality may corroborate the reported results with some or all of the other Implementers with whom testing occurred.

Implementers who have received a Stage Read/Write API key are required to add an entry(ies) into the Non-Production Test Ecosystem Directory that is available for testing at any time. Implementers are required to maintain entries in the directory that reasonably represent their production environment. Implementers MAY add additional entries to the Stage directory beyond the minimal list if they so choose.

Entries in the Test Ecosystem Directory, and the gateway(s) behind them, should behave substantially similarly to the production environment without access to live data. Specifically, test partners should expect that the same query that yielded a successful test in the Initiator role with entries in the Test Directory would yield a non-errored (see. 6.2.5) query response in production. These test entries should ONLY return information consisting of synthetic patient data including demographics and retrievable

files. No production clinical data should be available via gateways published in the Test directory. Implementers MUST ensure that their test gateway is active and ready to reply to test queries. Failure to repair these test entries in a timely manner (30 days after the problem has been reported) may result in punitive actions from Carequality which may include, but are not limited to, a denial of access to Carequality Directory read/write privileges. At this time, Carequality will not proactively test that the test directory entries are functioning, but downtime reported by test partners will begin the 30 day repair requirement.

Implementers who play the Query Initiator role for a non-treatment purpose must declare that fact to their Test Partners. Test Partners MUST NOT report success for the test unless they are able to successfully parse and recognize the specific non-treatment purpose for the query asserted by the Implementer being tested.

It is anticipated that many systems will automatically assign a particular Permitted Purpose when performing a query, based on the workflow from which the query originates. Therefore, an Implementer or its designated CCs may claim any Permitted Purpose within the transactions used for the connectivity test, including Treatment, as long as: (i) the patient record used in the transaction is a dummy record deliberately constructed so that it is reasonably expected not to match legitimate patient records; and (ii) the Implementer or CC is acting in good faith to perform a test as required by this Implementation Guide and is not knowingly attempting to access data for a real patient. A dispute may not be filed under the Carequality Dispute Resolution Process if it is based solely on the fact that test transactions performed under this validation process do not actually conform to their stated Permitted Purpose.

Upon completion of the tests, the Implementer must provide to Carequality a list of the partner Implementers involved and the outcome of the query, namely, (1) “No Matching Patient Found”, (2) an error, or (3) no response. Carequality may corroborate the reported results with some or all of the other Implementers with whom connectivity testing occurred. Implementers who themselves do not play a role in this Use Case may serve as Test Partners for other Implementers, either by designating a CC to perform the transactions or by using an internal environment as long as that environment has the same code base that will be delivered to the Implementer’s CCs. In such cases, the Implementer serving as the Test Partner will itself inform Carequality of the test’s successful completion, even if a CC performs the transactions on the Test Partner’s behalf.

6.2.4. Production Connectivity Validation – Pre-Live

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, applicants are not considered to have achieved Live Implementer status 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. Implementers who themselves do

not play a role in this Use Case must designate at least three CCs to individually perform the connectivity validation. In such a case, the designated CCs will each perform every step below that is described as required of the Implementer. The Implementer, however, will compile the results from all CCs and submit a single report to Carequality.

Implementers in the Query Responder role, or who support CCs in the Query Responder role, must receive Patient Discovery transactions from at least four other Implementers, and must respond successfully with a No Matching Patient Found" response for at least 75% of these transactions. Such a response is "successful" if it is received and processed without error by the querying system. If fewer than four other Implementers exist, the Patient Discovery transaction must be received from all other Implementers, with all other Implementers receiving a successful response as defined above.

Upon completion of this test, the Implementer must provide to Carequality a list of the other Implementers involved and the outcome of the query, namely, (1) "No Matching Patient Found", (2) an error, or (3) no response. Carequality may corroborate the reported results with some or all of the other Implementers with whom connectivity testing occurred.

If more than eight other Implementers exist, the connectivity test must be performed with at least half of the other Implementers, rounding up when there are odd numbers of Implementers. Connectivity must still be successful with 75% of the other Implementers, again rounding up if 75% is not an integer. For example, if there are nine other Implementers, an Implementer must perform the connectivity test with at least five of them. If the test is performed with five other Implementers, at least four must be successful.

It is anticipated that many systems will automatically assign a particular Permitted Purpose when performing a query, based on the workflow from which the query originates. Therefore, an Implementer or its designated CCs may claim any Permitted Purpose within the transactions used for the connectivity test, including Treatment, as long as: (i) the patient record used in the transaction is a dummy record deliberately constructed so that it is reasonably expected not to match legitimate patient records; and (ii) the Implementer or CC is acting in good faith to perform a test as required by this Implementation Guide and is not knowingly attempting to access data for a real patient. A dispute may not be filed under the Carequality Dispute Resolution Process if it is based solely on the fact that test transactions performed under this validation process do not actually conform to their stated Permitted Purpose.

While general experience shows that receiving the "No Matching Patient Found" response for a dummy patient is a reasonable method for establishing that connectivity will likely be successful between two parties, it does not guarantee that there is not a configuration issue related to the other required transactions. Therefore, all Implementers in the Query Responder and Query Initiator roles must complete testing with a Production Validation Partner. An Implementer must coordinate data with its Production Validation Partner such that connectivity can be confirmed for all required transactions for that Implementer's role or roles.

The Production Validation Partner may be the same as the Test Partner, and, like the Test Partner, must be an Implementer whose connectivity relies on software provided by a different technology vendor or provider. The CCs performing the validation steps on behalf of Implementers who themselves do not play a role may use the same Production Validation Partner as each other, or may choose different Production Validation Partners. Query Initiators must demonstrate that they are able to retrieve data successfully from the Production Validation Partner, while Query Responders must demonstrate that the Production Validation Partner is able to retrieve data successfully from them. Implementers are strongly encouraged to perform the validation with their Production Validation Partner using data from an actual shared patient can be used to perform validation as long as any appropriate access policy requirements are met, but if it is not possible to do so under policy constraints, coordinated dummy patient data can be used.

Upon completion of the validation to the Production Validation Partner's satisfaction, the Production Validation Partner will independently inform Carequality that the Implementer's production partner validation was successfully completed.

6.2.5. Production Connectivity Validation – Ongoing

Initial testing at first live use does not guarantee ongoing connectivity as systems and networks evolve over time. In order to demonstrate ongoing connectivity, Implementers acting in the Query Initiator role must submit monthly statistics to Carequality listing the number of non-errored XCPD query responses received from each other Responding Gateway, grouped by Implementer. For example, a "no matching patient found" response is considered a non-errored outbound XCPD query. Non-errored responses establish ongoing connectivity by demonstrating the Responding Gateway's ability to receive, parse, and respond to the Query Initiator's query. Implementers may use an automated or manual query of a test patient(s) to demonstrate ongoing connectivity (no matching patient found) when real exchange is unavailable.

Implementers shall provide a monthly report to Carequality of the number of documents retrieved from each Responding Gateway, grouped by Implementer. For any Responding Gateway where an Implementer in the Query Initiator Role has retrieved no documents over any consecutive three-month period, Carequality may require a test to validate the ability of the two Implementers to exchange with each other. Implementers shall work together to resolve any connectivity issues revealed through testing. An Implementer's persistent inability to resolve connectivity issues over a three-month period may result in suspension at Carequality's discretion.

All live Implementers in the Query Initiator Role shall also provide to Carequality response time statistics for each of the patient query (XCPD), document query (XCA Query), and document retrieve transactions (XCA Retrieve) grouped by Implementer. Implementers shall convey the following monthly statistics to Carequality by reporting: Median XCPD, Median XCA Query, and Median XCA Retrieve times. Implementers are required to report on a monthly basis, each Implementer to whom the Implementer made at least one request of a given query type in the relevant month. Implementers who support fewer than five Initiating Gateways shall provide such response time statistics for each of their Initiating Gateways, separately. Implementers who support more than five Initiating Gateways may select any five

Initiating Gateways to report, as long as the Implementer reasonably expects each of the five Initiating Gateways chosen to be representative of its community as a whole. The same five Initiating Gateways should not be used for reporting every month.

To satisfy these requirements, Implementers may enable functionality that will automatically query other live Implementers in order to prove ongoing connectivity.

Carequality requires that all Implementers transmit these statistics to the Carequality staff on a monthly basis. Failure to report these statics in a timely manner (30 days after the month in question has ended) may result in punitive actions from Carequality which may include, but are not limited to, a denial of access to Carequality Directory read/write privileges. Implementers MUST automate these reporting functions in instances where their connection to Carequality is structured around a singular Carequality gateway within 30 days of the effective date of this Implementation Guide. In instances where Implementers utilize multiple gateways, Carequality recommends that reports from each gateway should be automated to then be compiled and reported to Carequality staff.

7.0 Query-Based Document Exchange Use Case

7.1. Background

This use case describes the actors, transactions, and requirements to enable the exchange of health information between and among networks for simple query. The use case focuses on desired functionality, i.e. the user goals and how system actors meet them, highlighting the information that flows and the variations allowed by the existing specifications. Non-functional considerations such as security are minimized here for readability and covered in section 8.4.

The use case is written to enumerate all flows (both alternate and error) that are possible, given the underlying transactions. The decisions regarding which flows are considered in and out of scope for Carequality, and required/optional for roles/actors, are made in section 8.0, Technical Requirements and Guidance.

7.2. Use Case: Query Systems For Patient Information (XCPD/XCA)

In this use case, a user (acting through an Initiating Gateway) queries Responding Gateways for patient clinical information, using the IHE XCPD and XCA profiles.

7.2.1. Actors

1. Initiating Gateway (multiplicity of 1)
2. Responding Gateway (multiplicity of 1..*).
3. Participant Gateway Directory, i.e. phonebook (e.g. FHIR, UDDI or other) (multiplicity of 0..*)
4. Record Locator Service (multiplicity of 0..*)

7.2.2. Assumptions

1. The Initiating Gateway and Responding Gateway agree on transport level details (specified elsewhere in this document) that allow for the following:

a. Secure messaging over TLS.
b. The ability of the Initiating Gateway (and the Responding Gateway, in the case of deferred responses) to send information in each message that identifies security and permission details about the request such as: who is requesting, what their role is, and what their purpose is.
c. The ability of the Responding Gateway (and the Initiating Gateway, in the case of deferred responses) to choose if/how to allow the transaction to proceed based on this information and its own business rules.

7.2.3. Pre-conditions

1. The Initiating Gateway knows the patient's demographics.
2. (Nominal flow only) The Initiating Gateway has the desired service endpoint(s), and optionally the HCIDs, for some number of Responding Gateways that may be queried for patient information.

7.2.4. Use Case Steps – "Nominal Flow"

Each of the following steps may be repeated for each Responding Gateway of interest.

1. This use case begins when the Initiating Gateway sends an IHE Cross Gateway Patient Discovery [ITI-55] request to a Responding Gateway to attempt to match a patient by demographics. The request includes patient demographics (e.g. name, gender, date of birth) as known by the Initiating Gateway. See IHE ITI TF-1: 27 XCPD Integration Profile and IHE ITI TF-2b: 3.55.
2. The Responding Gateway compares the demographics to its known patients, applying its own algorithm to determine matches, and returns an IHE Cross Gateway Patient Discovery [ITI-55] response to the Initiating Gateway. The response contains a single patient match, including demographics and patient ID as known by the Responding Gateway. Each match (i.e. RegistrationEvent) includes the code NotHealthDataLocator to indicate that the corresponding community does not maintain externally available location information about this patient. See IHE ITI TF-2b: 3.55.4.2.2.5 Specifying support as a Health Data Locator.
3. The Initiating Gateway sends an IHE Cross Gateway Query [ITI-38] "FindDocuments" request to the Responding Gateway to query for document entries for this patient. "FindDocuments" refers to the fact that the ITI-38 request has multiple flavors, known as stored queries, such as FindFolders and GetAssociations. FindDocuments is the most basic query. The query includes a number of parameters, which restrict the set from all document entries available for the patient. The minimum required parameters for FindDocuments are the patient ID at the Responding Gateway and the status of the document entries to return, typically urn:oasis:names:tc:ebxml-regrep:StatusType:Approved. Approved in this context means the document is available for patient care. In addition, the Initiating Gateway specifies a returnType parameter value of LeafClass, which means to return full metadata contents. See IHE ITI TF-1: 18 Cross-Community Access (XCA) Integration Profile, IHE ITI TF-2b: 3.38, and IHE ITI TF-2a: 3.18.
4. The Responding Gateway filters its known documents by the query parameters passed in and returns an ITI-38 response containing a number of document entries, otherwise known as document metadata. In the document entry is a tuple of IDs (Home Community ID, Repository

ID, and Document unique ID) that enable an Initiating Gateway to later retrieve the actual document. See IHE ITI TF-3: 4.2.1.1 DocumentEntry.

5. The Initiating Gateway sends an IHE Cross Gateway Retrieve [ITI-39] request to the Responding Gateway to retrieve desired documents (a user may actively choose specific documents to retrieve based on the metadata). The request includes the document/repository/community IDs at the Responding Gateway. See IHE ITI TF-1: 18 Cross-Community Access (XCA) Integration Profile, IHE ITI TF-2b: 3.39, and IHE ITI TF-2b: 3.43.
6. The Responding Gateway retrieves the requested documents from its repositories and returns an ITI-39 response containing the documents and their related IDs.

7.2.5. Post-conditions

1. The Initiating Gateway has correlated its local patient ID and demographics to the patient ID and demographics as known by each Responding Gateway that returned a patient match that was confirmed by the Initiating Gateway. Left unspecified is whether the Initiating Gateway has persisted this correlation for later use beyond the completed workflow.
2. The Initiating Gateway has obtained the desired document entries as known by each Responding Gateway.
3. The Initiating Gateway has obtained the desired documents from each Responding Gateway.

7.2.6. Alternate Flows

1. Find Service Endpoint by HCID
    a. Prior to step 1, 3, or 5, the Initiating Gateway has the HCID of the community it wishes to query, but does not have the web services endpoint.
    b. The Initiating Gateway queries a Participant Gateway Directory for the endpoint of the desired service, passing the HCID.
        i. Note that there may be multiple ways to perform this query: pull everything about a HCID; first get business info then pull endpoints via separate queries, etc. Details of the querying are not specified.
    c. The Participant Gateway Directory returns the requested service endpoint for the Responding Gateway.
    d. The use case continues.
2. Find Service Endpoint by search parameters
    a. Prior to step 1, 3, or 5, the Initiating Gateway knows some information about the location at which the patient has been seen, but does not have the HCID of the community it wishes to query, nor the web services endpoint.
    b. The Initiating Gateway queries a Participant Gateway Directory for the endpoint of the desired service, passing search parameters such as: name and location of the healthcare facility, geographic area, provider specialty, provider name, use cases or profiles supported.
        i. Note that this is distinct from an RLS use case in that it uses “top-down” searching for patient data locations based on what is known by the Initiating Gateway, not “bottom-up” searching based on patient data locations explicitly known by an RLS service.

        ii. Note that there may be multiple ways to perform this query: pull everything about a HCID; first get business info then pull endpoints via separate queries, etc. Details of the querying are not specified.
    c. The Participant Gateway Directory returns the requested HCID and service endpoint for the Responding Gateway.
    d. The use case continues.
3. Find Service Endpoint by external directory
    a. In any of the "Find Service Endpoint" alternate flows, rather than communicating with a web services based Participant Gateway Directory, the Initiating Gateway utilizes an external directory (e.g. a web-based, human-readable directory) to obtain equivalent information.
    b. The use case continues.
4. Find Service Endpoint – multiple Responding Gateways found
    a. In any of the "Find Service Endpoint" alternate flows, the Participant Gateway Directory returns multiple Responding Gateways.
    b. The Initiating Gateway may attempt to further filter the Responding Gateways, for example, by presenting the responses to the patient, or may simply use all Responding Gateways found for the Query use case.
    c. The use case continues.
5. Use of directory to obtain information other than Responding Gateway endpoints
    a. In any of the "Find Service Endpoint" alternate flows, the Initiating Gateway queries a Participant Gateway Directory or external directory for information other than Responding Gateway endpoints, for example: use cases or profiles supported, internal organizations, levels of assurance.
    b. The use case continues.
6. Demographic Query and Feed mode
    a. In step 1, the ITI-55 request includes at least one patient ID as known by the Initiating Gateway, as well as an indication of which Assigning Authority ID to use in the event there is more than one patient ID. See IHE ITI TF-1: 27 XCPD Integration Profile and IHE ITI TF-2b: 3.55.4.1.2.4 Values used by Responding Gateway for a reverse Cross Gateway Query. The use case continues.
    b. Post-Condition (additional): The Responding Gateway may have persisted the correlation between its local patient ID and demographics and the patient ID and demographics as known by the Initiating Gateway. This allows the Responding Gateway, if paired with an Initiating Gateway, to execute this use case in reverse and skip steps 1 and 2.
    c. Note: in this case, both gateways have both sets of patient IDs and demographics, but they may have slightly different patient matching algorithms, so it is possible for one gateway to consider this a match and the other not to. See error flow "Initiating Gateway vetoes correlation".
7. Known third party patient identifier
    a. Background: The nominal use of the patient ID [Assigning Authority ID + unique ID] is as an opaque identifier from the perspective of the Initiating Gateway.
    b. In step 2 (or in alternate flow "Demographic Query and Feed mode"), the AAID is from a third party known to both Gateways, and the patient identifier is known or knowable to both Gateways through other means. Use of these third party identifiers can greatly increase the degree of confidence of a patient match. The use case continues.

8. Ambiguous match may be resolved with more demographics
    a. In step 2, the Responding Gateway cannot make a conclusive match, but may be able to if the Initiating Gateway provides additional demographics. The Responding Gateway returns a special error code indicating which specific demographics would help resolve the ambiguity. The Initiating Gateway chooses to execute one of the following subflows:
        i. Subflow 1: The Initiating Gateway repeats step 1, passing the additional demographics. The use case continues.
        ii. Subflow 2: The Initiating Gateway declines to pass additional demographics, perhaps due to privacy concerns. The flow ends for this Responding Gateway and the use case continues.
    b. See IHE ITI TF-2b: 3.55.4.2.2.6 Special handling for more attributes requested, and 3.55.4.2.3 Expected Actions, Case 3.
9. Multiple matches returned within a given HCID
    a. In step 2, the Responding Gateway returns multiple patient matches (i.e. multiple RegistrationEvents) with the same HCID. See IHE ITI TF-2b: 3.55.4.2.3 Expected Actions, Case 2, and 3.55.4.2.2.4 Specifying homeCommunityId in Response. This implies the patient matched multiple records at the Responding system, each of which pertains to a distinct patient. The Initiating Gateway chooses to execute one of the following subflows.
        i. Subflow 1: The Initiating Gateway attempts to resolve the patient match by comparing the demographics returned to its own. If it can resolve to one record, it continues to step 3. If not, the flow ends for this Responding Gateway and the use case continues.
        ii. Subflow 2: If policy permits, the Initiating Gateway continues with step 3 for each patient ID, and once all documents have been retrieved, attempts to disambiguate based on document content.
        iii. Subflow 3: The Initiating Gateway abandons the attempt to match the patient. The flow ends for this Responding Gateway and the use case continues.
10. Multiple matches returned with different HCIDs
    a. In step 2, the Responding Gateway returns multiple patient matches (i.e. multiple RegistrationEvents) with different HCIDs. This implies the patient was successfully matched, but has data under multiple patient records (e.g. at different facilities). See IHE ITI TF-2b: 3.55.4.2.2.4 Specifying homeCommunityId in Response.
    b. The Initiating Gateway resolves the HCIDs to endpoints, executing the “Find Service Endpoint” alternate flows if needed, and will use these endpoints later in step 3.
    c. The use case continues with step 3 for each patient ID.
11. Asynchronous patient discovery
    a. In step 1, the Initiating Gateway sends the Cross Gateway Patient Discovery request asynchronously. The request includes the endpoint to send the response to. The request returns immediately.
    b. In step 2, the Responding Gateway sends the Cross Gateway Patient Discovery response asynchronously.
    c. The use case continues.
12. Deferred patient discovery
    a. In step 1, the Initiating Gateway sends the Cross Gateway Patient Discovery request using the deferred mechanism.
    b. The Responding Gateway stores the request for later processing and returns an acknowledgement message immediately.

c. The Responding Gateway resolves the Initiating Gateway's HCID to the deferred response endpoint, executing a "Find Service Endpoint" alternate flow if needed.
d. In step 2, the Responding Gateway sends the Cross Gateway Patient Discovery response using the deferred mechanism. The response uses WS-Addressing RelatesTo and the XCPD QueryId to link back to the original request at both the transport and application layers respectively.
e. The Initiating Gateway returns an acknowledgement message.
f. The use case continues.

13. Health data locators returned
    a. In step 2, within one or more RegistrationEvents, the Responding Gateway returns the code SupportsHealthDataLocator. This indicates that the community identified by the Home Community ID in that RegistrationEvent is a Health Data Locator for this patient (aka a Record Locator Service).
    b. For each community identified as a Health Data Locator for this patient, the Initiating Gateway may execute the following subflow:
        i. The Initiating Gateway resolves the HCID to an endpoint, executing a "Find Service Endpoint" alternate flow if needed.
        ii. The Initiating Gateway sends an IHE Patient Location Query [ITI-56] request to the Responding Gateway to find communities where this patient may have healthcare data. The request includes the patient identifier as known by the Responding Gateway. See IHE ITI TF-1: 27 XCPD Integration Profile and IHE ITI TF-2b: 3.56 (some content is currently found in the XCPD Health Data Locator and Revoke Option supplement).
        iii. The Responding Gateway returns an ITI-56 response to the Initiating Gateway. The response contains some number of patient identifiers, each with a corresponding HCID.
        iv. The Initiating Gateway resolves the HCIDs to endpoints, executing the "Find Service Endpoint" alternate flows if needed, and will use these endpoints later in step 3.
        v. If the Initiating Gateway had previously obtained a list of potential communities to look for data for this patient through executing the "Find Service Endpoint" alternate flows, the requesting user or system may choose to reduce that list based on these results.
        vi. The use case continues with step 3 for each patient ID.
    c. The use case continues.

14. Asynchronous patient location query
    a. In step b.ii of alternate flow "Health data locators returned", the Initiating Gateway sends the Patient Location Query request asynchronously. The request includes the endpoint to send the response to. The request returns immediately.
    b. In step b.iii, the Responding Gateway sends the Patient Location Query response asynchronously.
    c. The use case continues.

15. Chunked document query
    a. Prior to step 3, the Initiating Gateway expects a large number of document entries.

b. In step 3, the Initiating Gateway passes a returnType value of ObjectRef, which means to return references to registry objects instead of the metadata-containing objects themselves. See IHE ITI TF-2a: 3.18.4.1.2.3.1 Parameter returnType.
c. In step 4, the Responding Gateway returns a list of matching object references.
d. The Initiating Gateway sends an IHE Cross Gateway Query [ITI-38] request to the Responding Gateway with a stored query that takes object references, for example, GetDocuments. See IHE ITI TF-2a: 3.18.4.1.2.3.7 Parameters for Required Queries for other queries.
e. The Responding Gateway returns an ITI-38 response containing a number of registry objects.
f. The Initiating Gateway continues to send similar requests until all desired registry objects have been retrieved.
g. The use case continues at step 5.

16. Advanced document queries
    a. In step 3, the Initiating Gateway queries for patient clinical information using one of the other XCA/XDS.b stored queries, which allow traversal of the relational XDS.b model of clinical information about a patient. See IHE ITI TF-3: section 4 Metadata used in Document Sharing profiles (section titled “Cross-Transaction Specifications” in earlier versions of the IHE ITI TF), and IHE ITI TF-2a: 3.18.4.1.2.3.7 Parameters for Required Queries.
        i. FindSubmissionSets – Find submission sets by filter parameters.
        ii. FindFolders – Find folders by filter parameters.
        iii. GetAll – Find document entries, submission sets, folders and associated document entries by filter parameters.
        iv. GetDocuments – Get document entries by reference.
        v. GetFolders – Get folders by reference.
        vi. GetAssociations – Get associations by associated object reference.
        vii. GetDocumentsAndAssociations – Get document entries and associations by reference.
        viii. GetSubmissionSets – Get submission sets by reference.
        ix. GetSubmissionSetAndContents – Get a submission set by reference, including all contained document entries, folders and associations.
        x. GetFolderAndContents – Get a folder by reference, including all contained document entries and associations.
        xi. GetFoldersForDocument – Get folders by document entry reference
        xii. GetRelatedDocuments – Get document entries by related document entry reference
    b. In step 4, the Responding Gateway returns an ITI-38 response containing the appropriate registry objects and/or object references.
    c. The use case continues.
17. Query for deprecated documents
    a. In step 3, the Initiating Gateway queries for a document status of urn:oasis:names:tc:ebxml-regrep:StatusType:Deprecated, which means to return historical document entries that have been superseded or are otherwise not considered valid for current clinical use.
    b. In step 4, the Responding Gateway returns a set of deprecated documents.
    c. The use case continues.
18. Document entries returned with different HCIDs

a. In step 4, the Responding Gateway returns document entries with different HCIDs than that of the Responding Gateway itself. This is not currently permitted by the XCA profile, but the Initiating Gateway may choose to be flexible and handle it.
b. The Initiating Gateway chooses to execute one of the following subflows.
    i. Subflow 1: The Initiating Gateway considers this an error. The flow ends for this Responding Gateway and the use case continues.
    ii. Subflow 2: The Initiating Gateway continues to use the same endpoint(s) for the Responding Gateway. The use case continues, and the Responding Gateway successfully handles and routes subsequent messages containing these different HCIDs.
    iii. Subflow 3: The Initiating Gateway resolves the HCIDs to endpoints, executing the "Find Service Endpoint" alternate flows if needed. The use case continues.

19. Query returns partial success
    a. In step 4, the Responding Gateway returns some but not all available document entries, along with the status urn:ihe:iti:2007:ResponseStatusType:PartialSuccess, and some number of RegistryError elements.
    b. The Initiating Gateway chooses to execute one of the following subflows.
        i. Subflow 1: The Initiating Gateway determines that it still wants these documents, so it continues to step 5 with the received document entries.
        ii. Subflow 2: The Initiating Gateway determines that it does not want to retrieve these documents. The flow ends for this Responding Gateway and the use case continues.
20. Asynchronous document query
    a. In step 3, the Initiating Gateway sends the Cross Gateway Query request asynchronously. The request includes the endpoint to send the response to. The request returns immediately.
    b. In step 4, the Responding Gateway sends the Cross Gateway Query response asynchronously.
21. On-demand documents, initial query/retrieve
    a. Additional precondition: The Initiating Gateway and Responding Gateway support the On-Demand Documents option. See IHE ITI TF On-Demand Documents supplement, Vol 2b, 3.43.4.2.2 Message Semantics.
    b. In step 3, the Initiating Gateway requests On-Demand document entries be included in the response via the $XDSDocumentEntryType query parameter.
    c. In step 4, the Responding Gateway returns On-Demand document entries.
    d. In step 5, the Initiating Gateway retrieves documents passing in On-Demand document entries, and may also pass stable document entries.
    e. In step 6, for each On-Demand document entry, the Responding Gateway returns a document based on the latest information available for that patient and document entry. In addition to the document content, the Responding Gateway will return NewDocumentUniqueId. If the Responding Gateway returns NewRepositoryUniqueId, this indicates that the Responding Gateway supports the Persistence of Retrieved Documents Option, meaning it has persisted a stable document that is a snapshot in time and may be retrieved at a later time using these identifiers – see alternate flow "On-demand documents, retrieve persisted document after change in underlying data".
    f. The use case continues.
22. On-demand documents, retrieve after change in underlying data

a. Additional precondition: the Initiating Gateway has previously retrieved an on-demand document entry, and since that time, the underlying patient data has been updated.
b. In step 5, the Initiating Gateway retrieves documents passing in On-Demand document entries, and may also pass stable document entries.
c. In step 6, for each On-Demand document entry, the Responding Gateway returns a new document containing the most recent snapshot of information for that patient. In addition to the document content, the Responding Gateway will return NewDocumentUniqueId. If the Responding Gateway returns NewRepositoryUniqueId, this indicates that the Responding Gateway supports the Persistence of Retrieved Documents Option, meaning it has persisted a stable document that is a snapshot in time and may be retrieved at a later time using these identifiers – see alternate flow "On-demand documents, retrieve persisted document after change in underlying data".
d. The use case continues.

23. On-demand documents, retrieve persisted document after change in underlying data
    a. Additional preconditions:
        i. The Responding Gateway supports the Persistence of Retrieved Documents Option.
        ii. The Initiating Gateway has previously retrieved an on-demand document entry and saved the returned NewDocumentUniqueId and NewRepositoryUniqueId.
        iii. Since the initial retrieve, the underlying patient data has changed.
    b. In step 5, the Initiating Gateway retrieves the persisted stable document passing in the saved NewDocumentUniqueId and NewRepositoryUniqueId, and may also pass On-Demand document entries.
    c. In step 6, the Responding Gateway returns the previously persisted stable document, which matches what was previously retrieved exactly.
    d. The use case continues.
24. Initiating Gateway begins with cached patient correlation
    a. Additional precondition: the Initiating Gateway has previously cached the correlation between its local patient identifier and the remote patient identifier at the Responding Gateway. This may have been obtained in one of the following ways:
        i. The Initiating Gateway has completed step 2 of a previous instance of the use case.
        ii. The Initiating Gateway has completed alternate flow "Demographic Query and Feed mode" of a previous instance of the use case as a Responding Gateway.
        iii. The Initiating Gateway has obtained the remote patient identifier through out-of-band means.
    b. The use case begins at step 3.
25. Retrieve returns partial success
    a. In step 6, the Responding Gateway returns some but not all requested documents, along with the status urn:ihe:iti:2007:ResponseStatusType:PartialSuccess, and some number of RegistryError elements.
    b. The use case continues.
26. Asynchronous document retrieve
    a. In step 5, the Initiating Gateway sends the Cross Gateway Retrieve request asynchronously. The request includes the endpoint to send the response to. The request returns immediately.
    b. In step 6, the Responding Gateway sends the Cross Gateway Retrieve response asynchronously.

c. The use case continues.

27. Initiating Gateway begins with cached document entry
    a. Additional precondition: the Initiating Gateway has previously cached a document entry identifier at the Responding Gateway. This may have been obtained in one of the following ways:
        i. The Initiating Gateway has completed step 4 of a previous instance of the use case.
        ii. The Initiating Gateway has obtained the remote document entry identifier through out-of-band means.
    b. The use case begins at step 5.
28. Initiating Gateway asserts Access Consent Policy
    a. In step 1, 3, or 5, the Initiating Gateway has included in its security information an identifier (by URI) of one or more Access Consent Policies (ACPs) applicable to this request.
        i. An ACP is a policy that the asserting entity has previously agreed to with other entities.
        ii. An asserted ACP may have an associated instance (IACP, e.g. a signed Patient Permission form) specific to this patient available for retrieval by the Responding Gateway using Carequality mechanisms.
    b. In step 2, 4, or 6 respectively, the Responding Gateway may incorporate these ACPs into its access decision. Additionally, the Responding Gateway may attempt to obtain IACP documents (see Alternate flows "Responding Gateway retrieves Patient Permission document…") and may incorporate these IACPs into its access decision. If an ACP URI is not supported, it shall be ignored.
    c. The use case continues.
29. Initiating Gateway asserts Instance Access Consent Policy
    a. In step 1, 3, or 5, the Initiating Gateway has included in its security information a document identifier (by URI) of one or more Instance Access Consent Policies (IACPs) applicable to this request.
        i. An IACP is a patient-specific access policy instance document, which must be available for retrieval by the Responding Gateway using Carequality mechanisms.
    b. In step 2, 4, or 6 respectively, the Responding Gateway may obtain these IACP documents (see Alternate flows "Responding Gateway retrieves Patient Permission document…") and may incorporate these IACPs into its access decision.
    c. The use case continues.
30. Responding Gateway retrieves Patient Permission document during Cross Gateway Patient Discovery transaction
    a. In step 1, alternate flow "Initiating Gateway asserts Access Consent Policy" and/or "Initiating Gateway asserts Instance Access Consent Policy" is taken.
    b. The Responding Gateway sends an IHE Cross Gateway Query [ITI-38] "FindDocuments" (by policy) or "GetDocuments" (by document) request to the Initiating Gateway to query for the document entry(ies) for the Patient Permission form(s).
    c. The Initiating Gateway returns an ITI-38 response containing document entry(ies) for the Patient Permission form(s).
    d. The Responding Gateway sends an IHE Cross Gateway Retrieve [ITI-39] request to the Initiating Gateway to retrieve the document(s).

e. The Initiating Gateway retrieves the requested document(s) from its repositories and returns an ITI-39 response containing the document(s).
f. The Responding Gateway completes its access determination and grants the Initiating Gateway access for this transaction.
g. The use case resumes at step 2.

31. Responding Gateway retrieves Patient Permission document during Cross Gateway Query transaction
    a. In step 3, alternate flow "Initiating Gateway asserts Access Consent Policy" and/or "Initiating Gateway asserts Instance Access Consent Policy" is taken.
    b. The Responding Gateway sends an IHE Cross Gateway Query [ITI-38] "FindDocuments" (by policy) or "GetDocuments" (by document) request to the Initiating Gateway to query for the document entry(ies) for the Patient Permission form(s).
    c. The Initiating Gateway returns an ITI-38 response containing document entry(ies) for the Patient Permission form(s).
    d. The Responding Gateway sends an IHE Cross Gateway Retrieve [ITI-39] request to the Initiating Gateway to retrieve the document(s).
    e. The Initiating Gateway retrieves the requested document(s) from its repositories and returns an ITI-39 response containing the document(s).
    f. The Responding Gateway completes its access determination and grants the Initiating Gateway access for this transaction.
    g. The use case resumes at step 4.
32. Responding Gateway retrieves Patient Permission document during Cross Gateway Retrieve transaction
    a. In step 5, alternate flow "Initiating Gateway asserts Access Consent Policy" and/or "Initiating Gateway asserts Instance Access Consent Policy" is taken.
    b. The Responding Gateway sends an IHE Cross Gateway Query [ITI-38] "FindDocuments" (by policy) or "GetDocuments" (by document) request to the Initiating Gateway to query for the document entry(ies) for the Patient Permission form(s).
    c. The Initiating Gateway returns an ITI-38 response containing document entry(ies) for the Patient Permission form(s).
    d. The Responding Gateway sends an IHE Cross Gateway Retrieve [ITI-39] request to the Initiating Gateway to retrieve the document(s).
    e. The Initiating Gateway retrieves the requested document(s) from its repositories and returns an ITI-39 response containing the document(s).
    f. The Responding Gateway completes its access determination and grants the Initiating Gateway access for this transaction.
    g. The use case resumes at step 6.

7.2.7. Error Flows

1. Error in SOAP request
    a. In step 2, 4, or 6, the Responding Gateway detects a problem with the SOAP request. This could be due to a number of reasons, such as:
        i. Missing required elements (e.g. timestamp)
        ii. Expired timestamp
        iii. Invalid XML signature
        iv. Untrusted, expired, or revoked certificate used to create XML signature
    b. The Responding Gateway executes one of the following subflows:

i. Subflow 1: The Responding Gateway returns a standard SOAP fault, for example: wsse:FailedAuthentication defined in SOAP Message Security 1.1.
ii. Subflow 2: The Responding Gateway returns a response with no results, for example, no match for XCPD. This case is where the Responding Gateway wishes to "hide the error" to avoid phishing attempts.

c. The Responding Gateway takes appropriate action to log the error.
d. The flow ends for this Responding Gateway and the use case continues.

2. Error in SOAP response
    a. Following step 2, 4, or 6, the Initiating Gateway detects a problem with the SOAP response. This could be due to a number of reasons, such as:
        i. Missing required elements (e.g. timestamp)
        ii. Expired timestamp
        iii. Invalid/missing signature confirmation
    b. The Initiating Gateway takes appropriate action to log the error.
    c. The flow ends for this Responding Gateway and the use case continues.
3. Access denied
    a. In step 2, 4, or 6, the Responding Gateway makes a determination that this request is to be denied due to some business rule/policy, for example, patient permission.
    b. The Responding Gateway returns a regular (i.e. no SOAP fault) response with no results, executing one of the following subflows:
        i. Subflow 1: In the SOAP header, the Responding Gateway returns the Carequality SOAP header block AccessDenial (see Transaction Detail Requirements). In the SOAP body, the Responding Gateway returns the transaction-specific error code of AnswerNotAvailable for XCPD or XDSRegistryError for XCA.
        ii. Subflow 2: The Responding Gateway returns a regular application response for no results found, for example, the flow "No patient match" for XCPD. This allows the Responding Gateway to "hide the error" to avoid release of sensitive information.
    c. The flow ends for this Responding Gateway and the use case continues.
4. Access partially denied
    a. In step 2, 4, or 6, the Responding Gateway makes a determination that part of this request is to be denied due to some business rule/policy, for example, patient permission, while part of the request can be granted.
    b. The Responding Gateway returns a regular (i.e. no SOAP fault) response with partial results, for example, some documents but not others, executing one of the following subflows:
        i. Subflow 1: In the SOAP header, the Responding Gateway returns the Carequality SOAP header block AccessDenial (see Transaction Detail Requirements). In addition,
            1. For XCA Query, the Responding Gateway populates the SOAP body as in the flow "Query returns partial success", using the error code XDSRegistryError.
            2. For XCA Retrieve, the Responding Gateway populates the SOAP body as in the flow "Retrieve returns partial success", using the error code XDSRegistryError.
        ii. Subflow 2: The Responding Gateway returns an application response where the partial results found appear to be the only results found, for example, step 2 of

the Nominal Flow for XCPD. This allows the Responding Gateway to “hide the error” to avoid release of sensitive information.

c. The use case continues.

5. Additional permission needed
    a. In Subflow 1 of Error Flow “Access denied” or Alternate Flow “Access partially denied”, the Responding Gateway includes in the AccessDenial header block details about the specific requirements to be met in order to gain access.
    b. The use case continues. The Initiating Gateway may retry the request after meeting the access requirements.
6. Responding Gateway not found
    a. In all of the available “Find Service Endpoint” alternate flows, no Responding Gateway can be found in any directory.
    b. The use case ends.
7. No patient match
    a. In step 2, the Responding Gateway is unable to make a conclusive match. This could be due to no matching patients, or due to an inability to disambiguate multiple potential matches. The Responding Gateway returns no RegistrationEvents (presence of RegistrationEvent elements in the response message indicate matches).
    b. The flow ends for this Responding Gateway and the use case continues.
8. Initiating Gateway vetoes correlation
    a. Following step 2, even though the Responding Gateway returned a positive match, the Initiating Gateway compares the returned demographics to its own and decides that the patient does not match.
    b. The flow ends for this Responding Gateway and the use case continues.
9. XCPD: Responding Gateway returns AnswerNotAvailable
    a. In step 2, the Responding Gateway determines that the answer is not available, and returns the code AnswerNotAvailable. This implies human intervention may be needed.
    b. The flow ends for this Responding Gateway and the use case continues.
10. XCPD: Responding Gateway cannot process Cross Gateway Patient Discovery for internal reasons
    a. In step 2, the Responding Gateway cannot process the patient discovery for some reason specific to the responding side. The Responding Gateway returns one of the following error codes:
        i. InternalError: an internal error or inconsistency
        ii. ResponderBusy: not able to process the request because it is currently overloaded
11. Patient location query returns no patient locations
    a. In step b.iii of alternate flow “Health data locators returned”, the Responding Gateway returns no locations.
    b. The alternate flow continues at step b for any other communities identified.
12. Responding Gateway is not a health data locator for this patient
    a. In step b.iii of alternate flow “Health data locators returned”, the Responding Gateway returns a Sender SOAP fault indicating it is “Not a Health Data Locator for the specified patient identifier”. See IHE ITI TF-2b Table 3.56-1: SOAP Faults (currently found in the XCPD Health Data Locator and Revoke Option supplement).
    b. The alternate flow continues at step b for any other communities identified.
13. Responding Gateway cannot process patient location query for internal reasons

a. In step b.iii of alternate flow “Health data locators returned”, the Responding Gateway cannot process the document query for some reason specific to the responding side. The Responding Gateway returns a Receiver SOAP fault. See IHE ITI TF-2b Table 3.56-1: SOAP Faults (currently found in the XCPD Health Data Locator and Revoke Option supplement).
b. The alternate flow continues at step b for any other communities identified.

14. Patient correlation becomes invalid
    a. Background: patient demographics may change over time, and in addition, patient records may be merged or linked. This means the quality of a patient correlation may degrade, and gateways may wish to force re-correlation. This is especially important when correlations are cached as in alternate flow “Initiating Gateway begins with cached patient correlation”. See IHE ITI TF-2: 3.55.4.2.3.1 Caching (Informative) and IHE ITI TF-3: Table 4.2.4.1-2: Error Codes.
    b. One of the following triggering subflows occurs:
        i. Subflow 1: In step 4, the Responding Gateway returns an ITI-38 response with error code XDSUnknownPatientId indicating the patient ID has become invalid and needs to be re-correlated.
        ii. Subflow 2: In step 2 of the Nominal Flow, the Responding Gateway includes a CorrelationTimeToLive SOAP header containing a duration in the response. The duration expires.
        iii. Subflow 3: In alternate flow “Demographic Query and Feed mode”, the Initiating Gateway includes a CorrelationTimeToLive SOAP header containing a duration in the request. The duration expires. At this point the Responding Gateway begins this alternate flow in the role of Initiating Gateway, and vice versa.
        iv. Subflow 4: At any time, an Initiating Gateway sends an IHE Revoke [ITI-55] request to a Responding Gateway to inform it that a patient correlation is no longer valid. At this point the Responding Gateway begins this alternate flow in the role of Initiating Gateway, and vice versa.
    d. The Initiating Gateway may choose to re-correlate the patient. If so, the use case begins at step 1.
15. No document entries found
    a. In step 4, the Responding Gateway cannot find any document entries for the patient that match the query parameters. It returns the status urn:oasis:names:tc:ebxml-regrep:ResponseStatusType:Success, and an empty RegistryObjectList.
    b. The flow ends for this Responding Gateway and the use case continues.
16. Query has bad inputs
    a. In step 4, the Responding Gateway detects problems with the inputs, for example: an invalid stored query ID is passed in. The Responding Gateway returns one or more RegistryError elements and status of either urn:ihe:iti:2007:ResponseStatusType:PartialSuccess or urn:oasis:names:tc:ebxml-regrep:ErrorSeverityType:Error. The error codes used in this flow are:
        i. XDSMissingHomeCommunityId
        ii. XDSStoredQueryMissingParam
        iii. XDSStoredQueryParamNumber
        iv. XDSUnknownCommunity
        v. XDSUnknownPatientId

vi. XDSUnknownStoredQuery

b. The flow ends for this Responding Gateway and the use case continues.

17. Responding Gateway cannot process document query for internal reasons
    a. In step 4, the Responding Gateway cannot process the document query for some reason specific to the responding side. The Responding Gateway returns one or more RegistryError elements and status of either urn:ihe:iti:2007:ResponseStatusType:PartialSuccess or urn:oasis:names:tc:ebxml-regrep:ErrorSeverityType:Error. The error codes used in this flow are:
        i. XDSRegistryBusy
        ii. XDSRegistryError
        iii. XDSRegistryOutOfResources
        iv. XDSTooManyResults
    b. The flow ends for this Responding Gateway and the use case continues.
18. Retrieve has bad inputs
    a. In step 6, the Responding Gateway detects problems with the inputs, for example: an invalid document ID is passed in. The Responding Gateway returns one or more RegistryError elements and status of either urn:ihe:iti:2007:ResponseStatusType:PartialSuccess or urn:oasis:names:tc:ebxml-regrep:ErrorSeverityType:Error. The error codes used in this flow are:
        i. XDSDocumentUniqueIdError
        ii. XDSMissingHomeCommunityId
        iii. XDSUnknownCommunity
        iv. XDSUnknownRepositoryId
    b. The flow ends for this Responding Gateway and the use case continues.
19. Responding Gateway cannot process document retrieve for internal reasons
    a. In step 6, the Responding Gateway cannot process the document retrieve for some reason specific to the responding side. The Responding Gateway returns one or more RegistryError elements and status of either urn:ihe:iti:2007:ResponseStatusType:PartialSuccess or urn:oasis:names:tc:ebxml-regrep:ErrorSeverityType:Error. The error codes used in this flow are:
        i. XDSRepositoryBusy
        ii. XDSRepositoryError
        iii. XDSRepositoryOutOfResources
    b. The flow ends for this Responding Gateway and the use case continues.

8.0 Technical Requirements and Guidance

8.1. Roles

Carequality introduces the concept of "roles", which are high-level aggregations of actors and behavior. See Section 2 of this Guide for additional information.

8.1.1. Query Initiator

Informative: Directory services are not in scope for the current version, but will be added in the future.

CONF-001: Each Query Initiator MUST provide an XCPD Initiating Gateway actor and support required transactions as described in this Technical Requirements and Guidance section.

CONF-002: Each Query Initiator MUST provide an XCA Initiating Gateway actor and support required transactions as described in this Technical Requirements and Guidance section.

8.1.2. Query Responder

CONF-003: Each Query Responder MUST provide an XCPD Responding Gateway actor and support required transactions as described in this Technical Requirements and Guidance section.

CONF-004: Each Query Responder MUST provide an XCA Responding Gateway actor and support required transactions as described in this Technical Requirements and Guidance section.

8.1.3. Record Locator Service

CONF-005: An XCPD Responding Gateway actor that supports the Health Data Locator option is considered a Carequality Record Locator Service and MUST adhere to the requirements in this Technical Requirements and Guidance section.

8.2. Overall Query Workflow

These requirements address multiple transactions and other cross-cutting concerns in the Query workflow.

8.2.1. Use Case Flow Requirements

This table shows the required flows from the Query use case for the Initiating (I) and Responding (R) Gateways.

Flow	I/R	Requirements
Nominal Flow	R	Required
Nominal Flow (Steps 1 and 2)	I	Choice: MUST support at least one of: Nominal Flow or Demographic Query and Feed mode
Nominal Flow (Steps 3-6)	I	Required
Multiple matches returned with different HCIDs	R	Optional

Multiple matches returned with different HCIDs	I	Required
Document entries returned with different HCIDs	R	Not currently permitted.
Document entries returned with different HCIDs	I	Required. The Initiating Gateway MUST implement at least one of the subflows to handle this case.
Patient correlation becomes invalid	R	Required. Responding Gateways MUST have the ability to detect that a patient correlation has become invalid, and report that via the error code XDSUnknownPatientId. Responding Gateways MAY additionally support Revoke and/or CorrelationTimeToLive.
Patient correlation becomes invalid	I	Required. Initiating Gateways MUST have the ability to process or ignore the error code XDSUnknownPatientId, and SHOULD re-correlate. Initiating Gateways MAY additionally support Revoke and/or CorrelationTimeToLive.
Initiating Gateway asserts Access Consent Policy	R	Required. Responding Gateway MUST ignore the reference without error if it is unsupported.
Initiating Gateway asserts Access Consent Policy	I	Optional
Initiating Gateway asserts Instance Access Consent Policy	R	Required. Responding Gateway MUST ignore the reference without error if it is unsupported.
Initiating Gateway asserts Instance Access Consent Policy	I	Optional
Responding Gateway retrieves Patient Permission document during Cross Gateway Patient Discovery transaction	I/R	Optional for Initiating Gateway, required if flow “Initiating Gateway asserts Instance Access Consent Policy” is supported. Optional for Responding Gateway: MAY query, retrieve and parse the Patient Permission document(s), and MAY incorporate the results into their access control decision.
Responding Gateway retrieves Patient Permission document during Cross Gateway Query transaction	I/R	Optional for Initiating Gateway, required if flow “Initiating Gateway asserts Instance Access Consent Policy” is supported. Optional for Responding Gateway: MAY query, retrieve and parse the Patient Permission document(s), and MAY incorporate the results into their access control decision.
Responding Gateway retrieves Patient Permission	I/R	Optional for Initiating Gateway, required if flow “Initiating Gateway asserts Instance Access Consent Policy” is supported.

document during Cross Gateway Retrieve transaction		Optional for Responding Gateway: MAY query, retrieve and parse the Patient Permission document(s), and MAY incorporate the results into their access control decision.

8.2.2. XCPD/XCA Gateway Requirements

Informative: Carequality will attempt, through guidance and constraints, to maintain forward and backward compatibility, but this will be subject to overriding concerns by participants.

CONF-006: All requirements pertaining to the IHE ITI Technical Framework, unless otherwise specified, refer to Revision 7.0 (2010), including:

- IHE IT Infrastructure Technical Framework Supplement – Cross-Community Patient Discovery (XCPD) Trial Implementation, Rev. 2.1 – 2010-08-10.
- IHE IT Infrastructure Technical Framework Supplement – Cross-Community Access (XCA) Trial Implementation, Rev. 2.1 – 2010-08-10.
- IHE IT Infrastructure Technical Framework Supplement – On-Demand Documents Trial Implementation, Rev. 1.1 – 2010-08-10.
- IHE IT Infrastructure Technical Framework Supplement – Patient Identifier Cross-Reference HL7 V3 (PIXV3) and Patient Demographic Query HL7 V3 (PDQV3), Rev. 2.1 – 2010-08-10.
    - Note: This supplement is needed for Volume 2b, Appendix O, used by XCPD.
- The following IHE ITI Change Proposals MUST be implemented unless otherwise specified below:
    - CP 459: Editorial – Fixes XDS.b retrieve example.
    - CP 460: Editorial – Fixes XDS.b SourcePatientId example.
    - CP 510: Normative: For non-HL7 transactions (e.g. XCA) requires receivers to ignore SOAP action HTTP header in favor of WS-Addressing Action. For IHE this is a normative change, but not for Carequality, as this is already required by WS-I Basic Security Profile 1.1 which is required by NHIN Messaging Platform 3.0.
    - CP 518: Normative: Fixes a handful of XCPD errors, mostly in the response and for cases where there is no patient match. Carequality participants SHOULD implement this CP, and SHOULD be tolerant of systems which have not.
    - CP 521: Editorial – Fixes XDS.b ExternalIdentifier example.
    - CP 531: Normative: Modifies XCA so that it will pass through query request/response parameters when it is grouped with XDS.b actors. This allows for XDS.b to evolve without having to modify XCA every time.
    - CP 534: Normative - Fixes wrappers for XCPD request. This is critical for correct functioning.
    - CP 535: Normative – Fixes detectedIssueEvent in XCPD response so that it doesn't conflict with underlying HL7V3 specification.
    - CP 544: Normative – Fixes incorrect object type for On-Demand Document Entries.
    - CP 546: Editorial – Fixes typo reference to ITI-16.
    - CP 547: Normative – Allows On-Demand Document Source that supports the Persistence of Retrieved Documents Option to optionally replace and deprecate

persisted documents. Since CONF-063 already requires queries to include a deprecated status, Carequality systems will not have an impact from adopting this CP.

- CP 557: Normative - Fixes other errors in XCPD request stemming from problems fixed by CP 534.
- CP 558: Editorial – Fixes lower/upper case typos in XCA retrieve examples.
- CP 572: Editorial - Fixes typo in XCPD example of homeCommunityId.
- CP 577: Normative – Restricts XDS.b document entry attribute SourcePatientID to a single value. Provides alternate way to return multiple patient ids within sourcePatientInfo.
- CP 578: Normative – More clearly calls out the need for patient ID translation when an XCA Initiating Gateway supports the XDS Affinity Domain Option. It needs to translate the patient ID to one known at the Responding Gateway. While normative, this is really just clarifying behavior that should have been inferred.
- CP 583: Normative – In auditing requirements, fixes incorrect references to nonexistent sections.
- CP 593: Editorial – fixes reference to HL7 CDA R1.

Informative: After choosing to reference the 2010 IHE ITI Technical Framework, Carequality performed an analysis of the Change Proposals incorporated into the 2011 Technical Framework, with a goal of choosing to adopt the highest value and most critical set. The heuristics applied were:

- Limit focus to CPs that apply to Carequality participants: ignore CPs in unused profiles, such as PIX/PDQ, and which affect unused features, such as querying by submission set.
- Adopt editorial CPs (i.e. non-normative), for example, corrections to examples.
- Adopt breaking CPs (i.e. normative changes) only if judged critical, after impact analysis with pilot participants.

Participants should be aware that Carequality wishes to continue to reduce interoperability issues, and that the current degree of tolerance for nonconformance (i.e. CPs that SHOULD rather than MUST be implemented) may be sunsetted in the future, subject to Steering Committee approval.

CONF-083: All requirements pertaining to the IHE PCC Technical Framework, unless otherwise specified, refer to Revision 11.0 Final Text 2016-11-11.

8.2.3. XCPD/XCA Federation

CONF-007: If a Query Initiator receives a Cross Gateway Patient Discovery (ITI-55) response with a match containing an HCID different from the Responding Gateway's community, and wishes to make a subsequent Patient Location Query (ITI-56) or Cross Gateway Query (ITI-38) using that match, it MUST resolve the HCID to a web services endpoint.

CONF-008: If a Query Initiator receives a Patient Location Query (ITI-56) response with a patient location with an HCID different from the Responding Gateway's community, and wishes to make a subsequent Cross Gateway Query (ITI-38) using that match, it MUST resolve the HCID to a web services endpoint.

Informative: See Section 8.3, Directory Services, for more information on this topic.

8.2.4. Flow: Patient correlation becomes invalid

CONF-013: An XCPD Initiating Gateway MAY support the Revoke option.

CONF-014: An XCPD Responding Gateway MAY support the Revoke option.

CONF-015: An XCPD Initiating Gateway that includes the CorrelationTimeToLive SOAP header in XCPD requests MUST NOT send a mustUnderstand value of “true” or “1”.

CONF-016: An XCPD Responding Gateway MAY support the CorrelationTimeToLive SOAP header in XCPD requests.

CONF-017: An XCPD Responding Gateway that includes the CorrelationTimeToLive SOAP header in XCPD responses MUST NOT send a mustUnderstand value of “true” or “1”.

CONF-018: An XCPD Initiating Gateway MAY support the CorrelationTimeToLive SOAP header in XCPD responses.

Informative: The XCPD profile, in sections 3.55.4.1.2 and 3.55.4.2.2, suggests not caching correlations unless CorrelationTimeToLive is sent. Carequality adopts the non-normative position that allowing optimistic caching, combined with requiring systems to detect patient identity issues and return XDSUnknownPatientId, is more deterministic and preferable.

8.2.5. Asserting Policies and Policy Instances

Within this document, the terms “Access Consent Policy” (ACP) and “Instance Access Consent Policy” (IACP) are used to indicate policies and policy instances (i.e. acknowledgements), which may influence access control decisions. See section **7.2.6, Alternate Flows:** “Initiating Gateway asserts Access Consent Policy” and “Initiating Gateway asserts Instance Access Consent Policy”.

This section defines a new formatting of ACP and IACP, which replaces the original formatting as defined in NHIN Authorization Framework 3.0, section 3.2.3 Authorization Decision Statement. This new formatting, combined with sections **8.2.6 Hosting and Retrieving Policy Instance Documents** and **8.4.5 Reporting Access Denials** supports full workflows for Access Consent Policy processing within the Query use case.

CONF-084: A Query Initiator asserting Access Consent Policy (ACP) or Instance Access Consent Policy (IACP) values MUST format these values according to the XUA Authz-Consent Option as defined in the IHE ITI Technical Framework, Rev. 13.0 Final Text 2016-09-09, Volume 2, section 3.40.4.1.2.2.

Informative: Unlike NHIN Authorization Framework, which uses the more complex SAML Authorization Decision Statement, the XUA Authz-Consent Option uses simple attribute statements. All statements within a SAML assertion are statements of fact by the Initiator, and can be relied upon by the Responder. Thus the more simple attribute statement is easier to produce and process while continuing to be asserted.

- Access Consent Policy – is referred to in IHE XUA as the “Patient Privacy Policy Identifier”
- Instance Access Consent Policy – is referred to in IHE XUA as the “Patient Privacy Policy Acknowledgement Document”
- Initiator – is referred to in IHE XUA as the “X-Service User”
- Responder – is referred to in IHE XUA as the “X-Service Provider”

CONF-085: A Query Initiator asserting Access Consent Policy (ACP) values MUST limit these values to the list specified in this Implementation Guide, section 4.4.1.

CONF-086 A Query Initiator asserting Instance Access Consent Policy (IACP) values MUST format these values as an OID-as-URN, as follows:

“urn:oid:” + <OID that is the document unique ID root>

Informative: in section 8.2.6, Patient Permission documents are constrained not to allow a document ID extension.

8.2.6. Hosting and Retrieving Policy Instance Documents

This section defines the requirements for Query Initiators to host policy instance documents (IACPs), typically Patient Permission forms in the Carequality context, and for Query Responders to retrieve them, using the references passed in SAML assertions. See section **7.2.6, Alternate Flows:** “Responding Gateway retrieves Patient Permission document…”. Note that in these flows, although the Query Responder acts as a Query Initiator temporarily, for clarity we do not change role names.

CONF-087: A Query Initiator that includes an IACP document unique id in a request MUST make the corresponding IACP Patient Permission document available for query and retrieval by the Query Responder.

CONF-088 A Query Initiator that includes an IACP document unique id in a request MAY also include the Patient Identifier Attribute in the SAML assertion. See NHIN Authorization Framework 3.0, section 3.2.2.7 (also defined in IHE XUA, section 3.40.4.1.2.2). Note: This is a relaxation of the requirement in 3.2.2.7, as the new mechanism to query by document id does not require a patient id.

CONF-089: A Query Initiator that includes an ACP policy id in a request MAY make a corresponding IACP Patient Permission document available for query and retrieval by the Query Responder. Note that some specific ACP policy ids defined by Carequality – see Section 4.4.1 in this Implementation Guide – must be accompanied by an IACP Patient Permission document in order for the Query Initiator to validly assert them.

CONF-090: A Query Initiator that includes an ACP policy id in a request and makes a corresponding IACP Patient Permission document available MUST also include the Patient Identifier Attribute in the SAML assertion. See NHIN Authorization Framework 3.0, section 3.2.2.7.

CONF-091: Unless otherwise specified by an additional agreement, in a context in which Carequality's non-discrimination rules permit additional agreements as a pre-requisite for honoring queries, a Query Initiator that makes IACP Patient Permission documents available MUST ensure each Patient Permission document and its metadata conforms to one of the following supported types:

- IHE Basic Patient Privacy Consents: Patient Privacy Consent Acknowledgment Document Specification With no Scanned Document Part (BPPC), as specified in the IHE ITI Technical Framework, Rev. 13.0 Final Text 2016-09-09, Volume 3, section 5.1.2.
- IHE Basic Patient Privacy Consents: Patient Privacy Consent Acknowledgment Document Specification With Scanned Document (BPPC-SD), as specified in the IHE ITI Technical Framework, Rev. 13.0 Final Text 2016-09-09, Volume 3, section 5.1.3.
    - Informative: This is used when there is a scanned document with a wet signature.

CONF-092: Unless otherwise specified by an additional agreement, in a context in which Carequality's non-discrimination rules permit additional agreements as a pre-requisite for honoring queries, a Query Initiator that makes IACP Patient Permission documents based on HL7 CDA available MUST not use an extension value in the CDA document id.

Informative: the above requirement is derived from the following:

- IHE XDS requires that the CDA document id be reflected in the document entry unique ID of the metadata.
- This guide defines a mechanism to query for an IACP Patient Permission document by document unique ID as passed in IHE XUA.
- In IHE XUA, the Patient Permission document id in a SAML assertion is formatted as an OID-as-URI, and there is no defined way to encode a document extension in a URI.

CONF-093: A Query Initiator that makes IACP consent documents based on HL7 CDA available MUST NOT include the caret "^" in the XDS document unique id when there is no extension value in the CDA document id.

Informative: Patient Permission documents based on CDA, such as IHE BPPC, pull in the IHE PCC Technical Framework Volume 2, section 4.1.1, which links the document id in the CDA with the document id in the metadata. However, the PCC requirement includes the carat, which is in conflict with the XDS metadata definition in IHE ITI Technical Framework Volume 3, Table 4.1-5 Document Metadata Attribute Definition.

Informative: Carequality adopts the value sets for XDS document metadata coded attributes defined in HITSP C80. See section 8.7.3. These also apply to Patient Permission documents.

CONF-094: A Query Initiator that makes IACP Patient Permission documents available SHOULD use the following XDS metadata values:

- confidentialityCode: N (Normal)
- healthcareFacilityTypeCode: 385432009 (SNOMED CT code for Not Applicable)

- practiceSettingCode: 385432009 (SNOMED CT code for Not Applicable)

CONF-095: A Query Initiator that makes IACP Patient Permission documents available MUST use the following XDS metadata values:

- eventCodeList: includes an entry with:
    - Code: The exact policy URI that is asserted in requests as the ACP value
    - Coding scheme: 2.16.840.1.113883.4.873, expressed as a raw OID (not URN)

Informative: The coding scheme is defined by HL7 specifically for URI identifiers. See https://www.hl7.org/fhir/identifier-registry.html and https://www.hl7.org/oid/index.cfm.

CONF-096: A Query Initiator that makes IACP Patient Permission documents based on IHE BPPC available MUST use the following values in the Patient Privacy Acknowledgement Service Event (see ITI TF 3: 5.1.2.2.6):

- <code> code attribute: The exact policy URI that is asserted in requests as the ACP value
- <code> codeSystem attribute: 2.16.840.1.113883.4.873, expressed as a raw OID (not URN)

CONF-097: A Query Initiator that makes IACP Patient Permission documents available MUST provide an XCA Responding Gateway actor and support required transactions as described in Technical Requirements and Guidance, section 8.

CONF-098: A Query Initiator that makes IACP Patient Permission documents available MUST support XCA Responding Gateway transactions between its own request and the responder's response to that request, in order to allow the Query Responder to incorporate these documents in its access decision.

CONF-099: A Query Responder that wishes to query and retrieve IACP Patient Permission documents MUST provide an XCA Initiating Gateway actor and support required transactions as described in Technical Requirements and Guidance, section 8.

CONF-100: A Query Responder that wishes to query and retrieve IACP Patient Permission documents MAY execute these XCA Initiating Gateway transactions before responding to the Query Initiator, in order to incorporate these documents in its access decision.

CONF-101: A Query Responder that wishes to query for IACP Patient Permission documents by IACP document id passed in a request MUST constrain the document query as follows:

- MAY use any stored query that takes document unique id, e.g. GetDocuments. However, note that only GetDocuments is required to be fully supported by the Query Initiator.
- In the slot $XDSDocumentEntryUniqueId, strip off the "urn:oid:" prefix from the IACP document ID received in the SAML assertion of the request.

CONF-102: A Query Responder that wishes to query for IACP Patient Permission documents by ACP policy id passed in a request MUST constrain the document query as follows:

- MAY use any stored query that takes event code list, e.g. FindDocuments. However, note that only FindDocuments is required to be fully supported by the Query Initiator.
- In the slot $XDSDocumentEntryPatientId, use the exact Patient Identifier Attribute received in the SAML assertion of the request.
- In the slot $XDSDocumentEntryEventCodeList, include a value with:
    - Code: The exact policy URI received in the SAML assertion of the request.
    - Coding scheme: 2.16.840.1.113883.4.873, expressed as a raw OID (not URN).

Informative: Query Initiators may host multiple Patient Permission documents for a given patient and policy. Query Responders may wish to apply further filters, such as format code, class code, service dates (which are typically tied to a document's effective dates) or status (because deprecated versions may exist). These filters are not constrained by this guide.

8.2.7. Other Requirements

Informative: The following requirement was prompted by a real system that wished to expose an XCPD gateway as essentially "only" an RLS.

CONF-019: A Query Responder that returns a patient ID in an XCPD response but does not have any clinical documents for that patient (whether it simply has no documents, or because it is acting as an RLS only), MUST return zero documents, not an XDSUnknownPatientID error code, in a response to an XCA Query for that patient ID.

Informative: There is a slight imbalance between the type of the patient ID returned in an XCPD response, which is of HL7V3 II type, and the type of the patient ID passed in a XCA Cross Gateway Query request, which is of HL7V2 CX type. The CX type as defined in HL7 2.5.1 suggests length restrictions on the assigning authority (227 chars) and ID Number (15 chars), which are not imposed on the corresponding HL7V3 II root and extension. After research, these lengths were not intended to be treated as maxima, so Initiating Gateways should be able to handle longer IDs.

CONF-020: A Query Initiator MUST be able to process without error HL7V3 II patient identifiers returned in an XCPD response whose Assigning Authority and/or ID Number are longer than 227 characters and 15 characters respectively, and use them in an XCA Cross Gateway Query request without truncating them.

8.3. Directory Services

8.3.1. Use Case Flow Requirements

This table shows the required flows from the Query use case for the Initiating Gateway (I) and Participant Gateway Directory (D).

Flow	I/D	Requirements
Nominal Flow	I	Required. Nominal flow assumes Initiating Gateway has already obtained endpoint(s) in some way.
Find Service Endpoint by HCID	I/D	Optional - this feature is not currently in scope and is not tested by Carequality.

Find Service Endpoint by search parameters	I/D	Optional - this feature is not currently in scope and is not tested by Carequality.
Find Service Endpoint by external directory	I	Optional
Find Service Endpoint – multiple Responding Gateways found	I	Required – Initiating Gateways MUST be able to support communicating with multiple gateways. Informative: This guide does not specify a processing model for communicating with multiple Responding Gateways, e.g. sequential or parallel, aggregation of results, human intervention, etc.
Use of directory to obtain information other than Responding Gateway endpoints	I	Optional
Responding Gateway not found	I/D	Optional - this feature is not currently in scope and is not tested by Carequality.

8.3.2. Detailed Requirements

Specific online directory services are not in scope for the current version, but will be added in the future. The current flows and requirements allow for much flexibility in how an Initiating Gateway might obtain endpoints.

CONF-021: An Initiating Gateway MUST have some way of knowing or discovering the service endpoints for a Responding Gateway.

CONF-022: An Initiating Gateway MUST have some way of resolving a HCID to the desired service endpoints for a Responding Gateway.

8.4. Security and Transport

8.4.1. Use Case Flow Requirements

This table shows the required flows from the Query use case for the Initiating (I) and Responding (R) Gateways.

Flow	I/R	Requirements
Nominal Flow	I/R	Required. Nominal flow assumes all security aspects function successfully.
Error in SOAP request	I/R	Required
Error in SOAP response	I	Required
Access denied	R	Optional. A Responding Gateway may choose not to implement access control, allowing access for all valid requests.
Access denied	I	Required, but note that this is simply a regular response potentially including an extra SOAP header block that may be ignored.

Access partially denied	R	Optional. A Responding Gateway may choose not to implement access control, allowing access for all valid requests.
Access partially denied	I	Required, but note that this is simply a regular response potentially including an extra SOAP header block that may be ignored. Initiators may choose to treat partial denials as full denials.
Additional permission needed	R	Optional
Additional permission needed	I	Required. The Initiating Gateway MAY choose to act on this information.

8.4.2. Referenced Specifications

CONF-023: An XCPD Initiating Gateway MUST implement the requirements in NHIN Messaging Platform 3.0 and NHIN Authorization Framework 3.0 (maintained by eHealth Exchange) except as constrained by this document.

CONF-024: An XCPD Responding Gateway MUST implement the requirements in NHIN Messaging Platform 3.0 and NHIN Authorization Framework 3.0 (maintained by eHealth Exchange) except as constrained by this document.

CONF-025: An XCA Initiating Gateway MUST implement the requirements in NHIN Messaging Platform 3.0 and NHIN Authorization Framework 3.0 (maintained by eHealth Exchange) except as constrained by this document.

CONF-026: An XCA Responding Gateway MUST implement the requirements in NHIN Messaging Platform 3.0 and NHIN Authorization Framework 3.0 (maintained by eHealth Exchange) except as constrained by this document.

8.4.3. Technical Trust

CONF-027: Carequality participants MUST follow the requirements listed in the separate document: Carequality Technical Trust Policy.

8.4.4. Digital Signatures

CONF-028: When Gateways include digital signatures in messages, the following instances of ds:KeyInfo:

- wsse:Security/saml:Assertion/ds:Signature/ds:KeyInfo – allows for validating the assertion signature
- wsse:Security/saml:Assertion/saml:Subject/saml:SubjectConfirmation/saml:SubjectConfirmationData/ds:KeyInfo – allows for validating the timestamp signature
- ds:Signature/ds:KeyInfo of any additional digital signatures

are limited to the following flavors of specifying KeyInfo such that the signature can be validated:

- ds:KeyInfo/ds:KeyValue/ds:RSAKeyValue
- ds:KeyInfo/ds:X509Data, and the included certificate must contain an RSA public key

Informative: This does not include the ds:KeyInfo instance in the timestamp signature: wsse:Security/ds:Signature/ds:KeyInfo/wsse:SecurityTokenReference, which uses Holder-of-Key to indirectly reference the SAML assertion SubjectConfirmation that contains the ultimate KeyInfo.

Informative: These flavors of KeyInfo are in common use and are known to be interoperable; they allow a receiving system to validate a signature without a priori knowledge or out-of-band exchange of the sender's public key, since the public key is included in the signature itself.

8.4.5. Reporting Access Denials

Informative: The use of SOAP faults for reporting access denials as specified in version 1.0 of this guide, including the Carequality UserNotAuthorized SOAP fault specified in requirement **CONF-029**, has been deprecated.

The following sections define the XML constructs that support reporting access denials and additional permission requirements. See section **7.2.7 Error Flows**: "Access denied", "Access partially denied" and "Additional permission needed".

8.4.5.1. Schema Header and Namespace Declarations

The following schema fragment defines the XML namespaces and other header information for the Carequality permission schema:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
           xmlns:tns="urn:carequality"
           targetNamespace="urn:carequality"
           blockDefault="#all"
           elementFormDefault="qualified"
           finalDefault=""
           attributeFormDefault="unqualified">

  <xs:import namespace="http://www.w3.org/XML/1998/namespace"
  schemaLocation="http://www.w3.org/2001/xml.xsd"/>
```

8.4.5.2. Element <AccessDenial>

The AccessDenial element is used as a SOAP header block in SOAP responses to indicate an access denial. It is similar in design and intent to the SOAP 1.2 Fault structure, except that being a header block, it is used in combination with a SOAP Body. This supports both full and partial denials (i.e. containing a subset of available results) using the same structure.

Responding Gateways are not required to report access denial errors; instead they may return an empty or partial response in order to prevent inadvertent disclosure of patient information, such as the presence of a record.

CONF-103: Query Responders MAY report full or partial access denial errors.

CONF-104: Query Responders, when reporting full or partial access denial errors, MUST use the Carequality defined AccessDenial SOAP header block.

CONF-105: When returning an AccessDenial SOAP header block, Query Responders MUST ensure that it conforms to the Carequality permission schema, included as an appendix to this document.

CONF-106: When returning an AccessDenial SOAP header block, Query Responders MUST NOT send a SOAP mustUnderstand value of “true” or “1”.

CONF-107: When returning an AccessDenial SOAP header block, Query Responders MUST send an isPartialDenial value of “true” or “1” if the denial is partial, that is, if the body of the response includes partial results, and “false” or “0” otherwise.

The following schema fragment defines the AccessDenial element and its AccessDenialType complex type:

```
<xs:element name="AccessDenial" type="tns:AccessDenialType"/>
<xs:complexType name="AccessDenialType">
  <xs:sequence>
    <xs:element name="Reason" type="tns:ReasonType"/>
    <xs:element name="Code" type="tns:DenialCodesOpenEnumType"
minOccurs="0"/>
    <xs:element name="Detail" type="tns:DetailType" minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="isPartialDenial" type="xs:boolean" use="required"/>
</xs:complexType>
```

8.4.5.3. Element <Reason>

The Reason element is intended to provide a human-readable explanation of the denial.

CONF-108: When returning an AccessDenial SOAP header block, Query Responders MUST include a Reason value with an explanation of the denial, and the xml:lang attribute valued according to XML 1.0 Section 2.12, Language Identification.

CONF-109: When returning an AccessDenial SOAP header block, Query Responders MAY use the following suggested values for the Reason element:

- For full denials: “The requester is not permitted to access this information.”
- For partial denials: “There is more information available for this request, but further permission would be needed.”

The following schema fragment defines the ReasonType complex type:

```
<xs:complexType name="ReasonType">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute ref="xml:lang" use="required"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

8.4.5.4. Elements <Code> and <Detail>

The Code and Detail elements are extensible mechanisms for returning processable information about the access denial. These mirror the pattern used by SOAP 1.2 faults, where codes imply the structure of the detail. A specification or agreement would be the typical way in which specific codes and their corresponding detail structures are defined.

CONF-110: When returning an AccessDenial SOAP header block, Query Responders MAY send a Code, which MUST be namespace-qualified. This specification defines a single code, AuthorizingPoliciesNeeded, but any namespace-qualified name MAY be used.

CONF-111: When returning an AccessDenial SOAP header block, Query Responders MAY send a Detail element containing processable information using a custom structure. If a Detail element is included, the Code element MUST also be included, and this code MUST unambiguously imply the structure of data in the Detail element. The mapping of codes to detail structures MAY be specified externally to this guide.

The following schema fragment defines the types of the Code and Detail elements:

```
<xs:simpleType name="DenialCodesOpenEnumType">
  <xs:union memberTypes="tns:DenialCodesType xs:QName"/>
</xs:simpleType>

<xs:simpleType name="DenialCodesType">
  <xs:restriction base="xs:QName">
    <xs:enumeration value="tns:AuthorizingPoliciesNeeded"/>
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="DetailType">
  <xs:sequence>
    <xs:any namespace="##any" processContents="lax" minOccurs="0"
maxOccurs="unbounded"  />
  </xs:sequence>
  <xs:anyAttribute namespace="##other" processContents="lax" />
</xs:complexType>
```

8.4.5.5. Element <QualifyingPolicies>

The QualifyingPolicies element is used within the Detail element when the Code AuthorizingPoliciesNeeded is used. This structure allows Query Responders to identify the specific combinations of policies that must be asserted in a subsequent request (using ACP values in the SAML assertion) to gain access to the patient information requested. As stated in section 4.4.2, Requirements for Query Responders, the policy expectations returned must be effective at the time they were returned, but may change by the time the Query Initiator is able to retry.

The expression of policies starts with a top-level expression of either AnyPolicy (i.e. inclusive OR) or AllPolicies (i.e. AND). Within this top-level expression, a group of policies and/or other Boolean operators may be nested to form a complex Boolean expression of policies needed. For example:

```
<cq:AccessDenial xmlns:S=http://www.w3.org/2003/05/soap-envelope
```

```
S:mustUnderstand="false" isPartialDenial="true">
  <cq:Reason xml:lang="en-US">There is more information available for this
request, but further permission would be needed. Either of the following
combinations of access consent policies may be asserted:
  1.2.3.4 AND 1.2.3.5
  OR
  1.2.3.6.
  </cq:Reason>
  <cq:Code>cq:AuthorizingPoliciesNeeded</cq:Code>
  <cq:Detail>
    <cq:QualifyingPolicies>
      <cq:AnyPolicy> <!-- i.e. Inclusive OR -->
        <cq:AllPolicies> <!-- i.e. AND -->
          <cq:AccessConsentPolicy uri="urn:oid:1.2.3.4"/>
          <cq:AccessConsentPolicy uri="urn:oid:1.2.3.5"/>
        </cq:AllPolicies>
        <cq:AllPolicies>
          <cq:AccessConsentPolicy uri="urn:oid:1.2.3.6"/>
        </cq:AllPolicies>
      </cq:AnyPolicy>
    </cq:QualifyingPolicies>
  </cq:Detail>
</cq:AccessDenial>
```

CONF-112: When returning an AccessDenial SOAP header block, Query Responders MAY include a QualifyingPolicies element under the Detail element. If this is included, the Code element MUST be valued as AuthorizingPoliciesNeeded.

The following schema fragment defines the QualifyingPolicies element and complex type:

```
<xs:element name="QualifyingPolicies" type="tns:QualifyingPoliciesType"/>
<xs:complexType name="QualifyingPoliciesType">
  <xs:choice>
    <!-- Choose the top-level expression -->
    <xs:element ref="tns:AnyPolicy"/>
    <xs:element ref="tns:AllPolicies"/>
  </xs:choice>
</xs:complexType>
```

8.4.5.6. Element <AnyPolicy>

The AnyPolicy element specifies an “inclusive OR” Boolean expression, such that if the Query Initiator asserts any combination of its children in a subsequent request, the Query Responder’s access policies will be satisfied. These children may be policy elements and/or AllPolicies elements.

For example, an AnyPolicy element with children of policy A and policy B implies that a subsequent request can satisfy the requirements with either policy A or policy B or both.

Complex Boolean expressions can be created by using the nested AllPolicies (i.e. AND) element. This allows for an outer OR expression with inner AND expressions, for example: (A AND B) OR (C AND D) OR E.

CONF-113: When returning an AnyPolicy element, Query Responders MUST include as its children policies and/or AllPolicies elements where any combination of these immediate children will satisfy the Query Responder's access policy requirements if asserted by the Query Initiator in a subsequent request.

The following schema fragment defines the AnyPolicy element and complex type:

```
<xs:element name="AnyPolicy" type="tns:AnyPolicyType"/>
<xs:complexType name="AnyPolicyType">
  <xs:choice minOccurs="1" maxOccurs="unbounded">
    <xs:element ref="tns:AccessConsentPolicy"/>
    <xs:element ref="tns:AllPolicies"/> <!-- Nested AND -->
  </xs:choice>
</xs:complexType>
```

8.4.5.7. Element <AllPolicies>

The AllPolicies element specifies an "AND" Boolean expression, such that if the Query Initiator asserts all of its children in a subsequent request, the Query Responder's access policies will be satisfied. These children may be policy elements and/or AnyPolicy elements.

For example, an AllPolicies element with children of policy A and policy B implies that a subsequent request can satisfy the requirements by asserting both policy A and policy B.

Complex Boolean expressions can be created by using the nested AnyPolicy (i.e. inclusive OR) element. This allows for an outer AND expression with inner OR expressions, for example: (A OR B) AND (C OR D) AND E.

CONF-114: When returning an AllPolicies element, Query Responders MUST include as its children policies and/or AnyPolicy elements where all of these immediate children must be asserted by the Query Initiator in a subsequent request in order to satisfy the Query Responder's access policy requirements.

The following schema fragment defines the AllPolicies element and complex type:

```
<xs:element name="AllPolicies" type="tns:AllPoliciesType"/>
<xs:complexType name="AllPoliciesType">
  <xs:choice minOccurs="1" maxOccurs="unbounded">
    <xs:element ref="tns:AccessConsentPolicy"/>
    <xs:element ref="tns:AnyPolicy"/> <!-- Nested Inclusive OR -->
  </xs:choice>
</xs:complexType>
```

8.4.5.8. Element <AccessConsentPolicy>

The AccessConsentPolicy element is used to specify the actual policies that need to be asserted in the SAML assertion of subsequent requests to gain access to patient information. See section **8.2.5 Asserting Policies and Policy Instances**.

CONF-115: When returning an AccessConsentPolicy element, Query Responders MUST specify in the "uri" attribute a unique Access Consent Policy Identifier in URI format, e.g. "urn:oid:1.2.3.4", of an Access Consent Policy that can be asserted in a subsequent request.

The following schema fragment defines the AccessConsentPolicy element:

```
<xs:element name="AccessConsentPolicy" type="tns:ConsentPolicyType"/>
<xs:complexType name="ConsentPolicyType">
  <xs:attribute name="uri" type="xs:anyURI" use="required"/>
</xs:complexType>
```

8.5. Patient Discovery

8.5.1. Use Case Flow Requirements

This table shows the required flows from the Query use case for the Initiating (I) and Responding (R) Gateways.

Flow	I/R	Requirements
Nominal Flow (Steps 1 and 2)	R	Required
Nominal Flow (Steps 1 and 2)	I	Choice: MUST support at least one of: Nominal Flow or Demographic Query and Feed mode.
Demographic Query and Feed mode	R	Required. Responding Gateways MAY use the patient ID passed in to persist a correlation.
Demographic Query and Feed mode	I	Choice: MUST support at least one of: Nominal Flow or Demographic Query and Feed mode.
Known third party patient identifier	R	Optional. Responding Gateways MAY return known third party patient identifiers in responses. Responding Gateways MAY base matches on known third party patient identifiers received in requests.
Known third party patient identifier	I	Optional. Initiating Gateways MAY send known third party patient identifiers in requests. Initiating Gateways MAY base matches on known third party patient identifiers received in responses.
Ambiguous match may be resolved with more demographics	R	Optional
Ambiguous match may be resolved with more demographics	I	Required. If received in a response, Initiating Gateways MAY treat the same as no patient match found.
Multiple matches returned within a given HCID	R	Optional

Multiple matches returned within a given HCID	I	Required
Asynchronous patient discovery	R	Optional
Asynchronous patient discovery	I	Optional
Deferred patient discovery	I/R	Optional, but this feature is not used currently by Carequality, nor will it be tested.
No patient match	I/R	Required
Initiating Gateway vetoes correlation	R	N/A. If the IG vetoes, the RG is unaware of it.
Initiating Gateway vetoes correlation	I	Optional
XCPD: Responding Gateway returns AnswerNotAvailable	R	Optional
XCPD: Responding Gateway returns AnswerNotAvailable	I	Required
XCPD: Responding Gateway cannot process Cross Gateway Patient Discovery for internal reasons	R	Optional
XCPD: Responding Gateway cannot process Cross Gateway Patient Discovery for internal reasons	I	Required

8.5.2. Detailed Requirements

CONF-030: An XCPD Initiating Gateway MUST implement the appropriate requirements in IHE ITI TF-2b: 3.55.

CONF-031: An XCPD Responding Gateway MUST implement the appropriate requirements in IHE ITI TF-2b: 3.55.

CONF-032: An XCPD Initiating Gateway MUST support Synchronous Web Services Exchange, and if also using the Asynchronous Web Services Exchange option MUST send to the appropriate Endpoint Type in the Carequality Directory.

CONF-033: An XCPD Responding Gateway MUST use Synchronous Web Services Exchange, and MAY additionally use the Asynchronous Web Services Exchange option. The appropriate Endpoint Type MUST be published for each in the Carequality Directory.

CONF-034: An XCPD Initiating Gateway MAY support the Deferred Response option. However, Carequality is not currently using this, so it will not be tested.

CONF-035: An XCPD Initiating Gateway MUST NOT require a Responding Gateway to support the Deferred Response option as a precondition to interoperate.

CONF-036: An XCPD Responding Gateway MAY support the Deferred Response option. However, Carequality is not currently using this, so it will not be tested.

Informative: In the Transmission Wrapper of the ITI-55 Cross Gateway Patient Discovery request and response and Revoke messages, the fields sender/device/id and receiver/device/id, while required, are not defined by XCPD. They are defined by the HL7 transmission infrastructure, which is not entirely utilized by Carequality. In other production exchanges, gateways have been known to make assumptions about these values, which has led to interoperability problems, so we are clarifying that outside a higher level agreement, these values are unconstrained.

Carequality is aware of some systems that do make use of this infrastructure to perform more sophisticated routing - for example, a Responding Gateway will expect a certain value in receiver/device/id. Currently this can only be coordinated through individual partner agreement, but in the future, Carequality may attempt to provide further guidance and constraints on these fields.

CONF-009: In the Transmission Wrapper of the ITI-55 Cross Gateway Patient Discovery request and Revoke messages, an XCPD Initiating Gateway MAY send any conformant value for the fields sender/device/id and receiver/device/id, unless constrained through a higher level agreement.

CONF-010: In the Transmission Wrapper of the ITI-55 Cross Gateway Patient Discovery request and Revoke messages, an XCPD Responding Gateway SHOULD NOT make any assumptions about the values of the fields sender/device/id and receiver/device/id, unless constrained through a higher level agreement.

CONF-011: In the Transmission Wrapper of the ITI-55 Cross Gateway Patient Discovery response message, an XCPD Responding Gateway MAY send any conformant value for the fields sender/device/id and receiver/device/id, unless constrained through a higher level agreement.

CONF-012: In the Transmission Wrapper of the ITI-55 Cross Gateway Patient Discovery response message, an XCPD Initiating Gateway SHOULD NOT make any assumptions about the values of the fields sender/device/id and receiver/device/id, unless constrained through a higher level agreement.

CONF-037: An XCPD Initiating Gateway MUST send, in the ITI-55 Cross Gateway Patient Discovery request, all demographic parameters that are available and can be sent and are not constrained by local policy.

See IHE ITI TF-2b: 3.55.4.1.2.2 Message Information Model of the Patient Registry Query by Demographics Message.

CONF-038: An XCPD Responding Gateway MUST send, in each RegistrationEvent in the ITI-55 Cross Gateway Patient Discovery response, all demographic parameters that are available and can be sent and are not constrained by local policy.

See IHE ITI TF-2b: 3.55.4.2.2.2 Message Information Model of the Patient Registry Find Candidates Response Message.

CONF-039: An XCPD Initiating Gateway SHOULD include the "use" attribute for the patientTelecom/value element in the ITI-55 Cross Gateway Patient Discovery request.

CONF-040: An XCPD Responding Gateway SHOULD include the "use" attribute for the telecom element in the ITI-55 Cross Gateway Patient Discovery response.

CONF-041: An XCPD Initiating Gateway that receives multiple matches with the same HCID and a different AAID in an XCPD response SHOULD allow a user to manually review the matches before proceeding. They may represent either multiple people who could not be resolved to a single match (IHE interpretation) or multiple sources of documents for the same person (eHealth Exchange interpretation).

Informative: The XCPD request parameters MatchAlgorithm and MinimumDegreeMatch do not have deterministic meaning defined by the XCPD profile. Responding Gateways may make known if/how they will interpret these parameters in light of their specific matching algorithms, but how this is communicated is out of scope of this guide. If an XCPD Initiating Gateway sends request parameters MatchAlgorithm and MinimumDegreeMatch without knowing their interpretation by the Responding Gateway, they should not expect consistent results.

CONF-042: An XCPD Responding Gateway that receives an unexpected or unsupported value of the request parameter MatchAlgorithm MUST process the message as if that value were not present.

Informative: This Implementation Guide defines a single value for MatchAlgorithm that equates to the semantics used by the eHealth Exchange. The value is optional to be provided, and should be supported.

CONF-043: An XCPD Initiating Gateway MAY provide the request parameter MatchAlgorithm with a value of "urn:carequality:OneMatchPerAAID".

CONF-044: If an XCPD Responding Gateway supports the request parameter MatchAlgorithm with a value of "urn:carequality:OneMatchPerAAID", if it receives this value in an XCPD request:

- It MUST restrict matches to one per AAID. This implies consolidating multiple sources of data for a given patient within a given AAID to a single record.
- If it returns multiple matches per HCID (each with a different AAID), these MUST be multiple sources of data for the same person, not multiple patients who must be disambiguated by the Initiating Gateway. Informative: this is different semantics than the underlying IHE XCPD requirements, which state that these MUST be multiple patients who must be disambiguated by the Initiating Gateway.

CONF-045: An XCPD Responding Gateway SHOULD support the request parameter MatchAlgorithm with a value of “urn:carequality:OneMatchPerAAID”.

CONF-046: An XCPD Initiating Gateway that provides the request parameter MatchAlgorithm with a value of “urn:carequality:OneMatchPerAAID” SHOULD be able to process responses from Responding Gateways that do not support this value, e.g. 1. presenting the multiple matches to the user for disambiguation, or 2. presenting no matches and documenting the possibility of false negatives.

Informative: This includes multiple matches per AAID, as well as multiple matches per HCID that represent multiple patients that must be disambiguated.

8.6. Record Locator Services

8.6.1. Use Case Flow Requirements

This table shows the required flows from the Query use case for the Initiating (I) and Responding (R) Gateways.

Flow	I/R	Requirements
Health data locators returned	R	Optional
Health data locators returned	I	Required. Initiating Gateways MUST be able to process responses that indicate Health Data Locators, and MAY make use of them with ITI-56 transactions.
Asynchronous patient location query	R	Optional
Asynchronous patient location query	I	Optional
Patient location query returns no patient locations	I/R	Required
Responding Gateway is not a health data locator for this patient	I/R	Required
Responding Gateway cannot process patient location query for internal reasons	R	Optional
Responding Gateway cannot process patient location query for internal reasons	I	Required

8.6.2. Detailed Requirements

Informative: A Record Locator Service is an optional value-added service provided by an XCPD Responding Gateway. It adds value by potentially limiting the scope of communities a requester needs to contact in order to find information about a patient.

Scope of the RLS: A given RLS covers some number of communities, and it is important that the requesting user understands this scope, and does not assume that the RLS is asserting knowledge about the presence or absence of patient data in communities outside of that scope.

Quality of the RLS: It is important to note that the RLS interface and behavior requirements do not specify how the service keeps track of patient data, nor do they guarantee the accuracy or completeness of results. For example, a community could be returned as a possible location that has no clinical documents for this patient, or a community could be left out of the results that does have clinical documents for this patient. The former is less of a problem, as it will be discovered when attempting to query for documents, but the latter situation can hide useful clinical data, which might have been found using a broader search. Individual record locator services can differentiate by explaining and demonstrating how they ensure accurate results.

CONF-047: An XCPD Initiating Gateway MAY support the Health Data Locator option.

CONF-048: An XCPD Responding Gateway MAY support the Health Data Locator option.

CONF-049: An XCPD Initiating Gateway exercising ITI-56 MUST implement the appropriate requirements in IHE ITI TF-2b: 3.56.

CONF-050: An XCPD Responding Gateway exercising ITI-56 MUST implement the appropriate requirements in IHE ITI TF-2b: 3.56.

8.7. Document Query and Retrieve

8.7.1. Use Case Flow Requirements

This table shows the required flows from the Query use case for the Initiating (I) and Responding (R) Gateways.

Flow	I/R	Requirements
Nominal Flow (Steps 3 and 4)	I/R	Required
Chunked document query	R	Required
Chunked document query	I	Optional
Advanced document queries	I/R	See Detailed Requirements.
Query for deprecated documents	R	Required

Query for deprecated documents	I	Optional
Query returns partial success	R	Conditional. If Responding Gateway can encounter partial success, it MUST communicate it. See Detailed Requirements.
Query returns partial success	I	Required. See Detailed Requirements.
Asynchronous document query	R	Optional
Asynchronous document query	I	Optional
On-demand documents, initial query/retrieve	R	Conditional. MUST support if supports the On-Demand Documents option.
On-demand documents, initial query/retrieve	I	Required
On-demand documents, retrieve after change in underlying data	R	Conditional. MUST support if supports the On-Demand Documents option.
On-demand documents, retrieve after change in underlying data	I	Required
On-demand documents, retrieve persisted document after change in underlying data	R	Conditional. MUST support if supports the On-Demand Documents option (which requires the Persistence of Retrieved Documents Option).
On-demand documents, retrieve persisted document after change in underlying data	I	Optional. Initiating Gateway MAY choose to retrieve persisted documents.
Initiating Gateway begins with cached patient correlation	R	Required
Initiating Gateway begins with cached patient correlation	I	Optional. Initiating Gateway MAY cache correlations.
Retrieve returns partial success	I/R	Conditional. See Detailed Requirements.

Asynchronous document retrieve	R	Optional
Asynchronous document retrieve	I	Optional
Initiating Gateway begins with cached document entry	R	Required
Initiating Gateway begins with cached document entry	I	Optional. Initiating Gateway MAY cache document entries.
No document entries found	I/R	Required
Query has bad inputs	I/R	Required. Responding Gateway MUST detect these conditions and Initiating Gateway MUST be able to process these error codes. See Detailed Requirements.
Responding Gateway cannot process document query for internal reasons	R	Optional
Responding Gateway cannot process document query for internal reasons	I	Required
Retrieve has bad inputs	I/R	Required
Responding Gateway cannot process document retrieve for internal reasons	R	Optional
Responding Gateway cannot process document retrieve for internal reasons	I	Required

8.7.2. XCA Gateway Requirements

CONF-051: An XCA Initiating Gateway MUST implement the appropriate requirements in IHE ITI TF Vol2b: 3.38 and 3.39.

CONF-052: An XCA Responding Gateway MUST implement the requirements in IHE ITI TF Vol2b: 3.38 and 3.39.

CONF-053: An XCA Responding Gateway MAY satisfy ITI-38 and ITI-39 transactions through either a single endpoint or one endpoint for each.

8.7.3. Document Metadata Vocabulary

CONF-054: Carequality adopts the value sets for document metadata elements defined in HITSP C80, version 2.0.1, according to the table below:

Document Metadata	HITSP C80 reference	scheme OID
classCode	HITSP C80, version 2.0.1, table 2-144	2.16.840.1.113883.6.1
confidentialityCode	HITSP C80, version 2.0.1, table 2-150.	2.16.840.1.113883.5.25
eventCodeList	Very specific to the type of document and not expected to be constrained externally.	
formatCode	HITSP C80, version 2.0.1, table 2-152, not including concept code urn:nhin:names:acp:XACML	1.3.6.1.4.1.19376.1.2.3
healthcareFacilityTypeCode	HITSP C80, version 2.0.1, table 2-146	2.16.840.1.113883.6.96
practiceSettingCode	HITSP C80, version 2.0.1, table 2-149 which is a list of members of the value set in table 2-148	2.16.840.1.113883.6.96
typeCode	HITSP C80, version 2.0.1, table 2-144 - same list of values as used for classCode	2.16.840.1.113883.6.1

Informative: Carequality is adopting these value sets in the absence of any other governing body for nationwide value sets.

Informative: An XCA Initiating Gateway SHOULD make no assumptions that XCA Responding Gateways use the HITSP C80 vocabulary. If useful clinical data is not received while querying, filtering by coded values, consider not filtering by coded values.

CONF-055: An XCA Responding Gateway SHOULD use the vocabulary defined in HITSP C80, version 2.0.1 as well as the schemes identified in the above table, for document metadata elements.

8.7.4. XCA Profile Options

CONF-056: An XCA Initiating Gateway MAY support the XDS Affinity Domain option. However, Carequality will neither make use of nor test this option.

CONF-057: An XCA Initiating Gateway MUST support Synchronous Web Services Exchange, and if also using the Asynchronous Web Services Exchange option MUST send to the appropriate Endpoint Type in the Carequality Directory.

CONF-058: An XCA Responding Gateway MUST use Synchronous Web Services Exchange, and MAY additionally use the Asynchronous Web Services Exchange option. The appropriate Endpoint Type MUST be published for each in the Carequality Directory.On-Demand Documents

CONF-059: An XCA Initiating Gateway MUST support the On-Demand Documents option.

CONF-060: An XCA Responding Gateway MAY support the On-Demand Documents option.

CONF-061: An XCA Responding Gateway that supports the On-Demand Documents option MUST support the Persistence of Retrieved Documents option.

Informative: Because there is no in-band way for Initiating Gateways to know if they are interacting with Stable or On-Demand systems, the following guidance ensures the Initiating Gateway will not miss available clinical data.

CONF-062: An XCA Initiating Gateway MUST request both On-Demand and Stable document entries, unless it is exercising a use case that requires targeted query of only On-Demand or Stable.

Informative: Some XCA Responding Gateways that support the On-Demand Documents option and the Persistence of Retrieved Documents Option deprecate all persisted stable documents as soon as they are generated. Others use the replacement mechanism to replace and deprecate all but the most recently retrieved stable document. Initiating Gateways should be aware of these behaviors. The conformance statement below prevents the Initiating Gateway from false negatives in the query response, but still allows it to selectively retrieve only the approved entry if it wishes.

CONF-063: An XCA Initiating Gateway wishing to retrieve a persisted stable document from an On-Demand document entry MUST include the document status of urn:oasis:names:tc:ebxml-regrep:StatusType:Deprecated in the query.

Informative: An XCA Initiating Gateway retrieving the same On-Demand document entry multiple times can compare the NewDocumentUniqueId to the one obtained with the previous retrieve. If they are the same, then the data has not changed. If they are different, then the data may have changed. See ITI TF Vol2b 3.43.4.2.2 Message Semantics.

CONF-064: An XCA Responding Gateway SHOULD NOT return the optional elements NewRepositoryUniqueId and NewDocumentUniqueId for stable documents in an ITI-39 response.

CONF-065: An XCA Responding Gateway that does not support the Persistence of Retrieved Documents Option SHOULD NOT return the optional element NewRepositoryUniqueId for on-demand documents in an ITI-39 response, as it does not have any defined meaning.

8.7.5. Supported Queries

CONF-066: An XCA Initiating Gateway MUST support the FindDocuments stored query.

Informative: The concepts of submission sets, folders and associations are not used by Carequality. Therefore, if an XCA Initiating Gateway sends the following stored queries it may receive no results:

FindSubmissionSets, FindFolders, GetAll, GetFolders, GetAssociations, GetDocumentsAndAssociations, GetSubmissionSets, GetSubmissionSetAndContents, GetFolderAndContents, GetFoldersForDocument, GetRelatedDocuments.

CONF-067: An XCA Responding Gateway MUST support all stored queries in IHE ITI TF Vol2b: Table 3.38.4.1.2.3-1.

CONF-068: An XCA Responding Gateway MAY return zero elements for non-supported concepts as specified in IHE ITI TF Vol2b: Table 3.38.4.1.2.3-1.

Informative: FindDocumentsByReferenceId is a relatively new stored query that is included in the XDS.b profile via a named option. It is not listed as an option in XCA, and further, XCA includes all XDS.b queries by reference. Carequality does not intend to use this query at this time.

CONF-069: An XCA Initiating Gateway SHOULD NOT send the FindDocumentsByReferenceId stored query.

CONF-070: An XCA Responding Gateway, if it receives a FindDocumentsByReferenceId stored query, MAY do any of the following: support it, return zero elements, or return the XDSUnknownStoredQuery error.

8.7.6. Query Behavior

CONF-071: An XCA Responding Gateway MUST compare coded value query parameters by the combination of code and scheme.

CONF-072: An XCA Responding Gateway MUST compare date query parameters to the corresponding metadata as specified in IHE ITI TF Vol2a: 3.18.4.1.2.3.3 Date/Time Coding.

8.7.7. Error Handling

Informative: The requirements below for conveying errors to end users may be met via logs.

CONF-073: An XCA Initiating Gateway SHOULD, in the case of a Failure result in an ITI-38 response, convey to an end user that no documents are currently available as queried, and convey the reasons for the problem(s) via the RegistryError elements returned.

CONF-074: An XCA Initiating Gateway SHOULD, in the case of a PartialSuccess result in an ITI-38 response, convey to an end user that some but not all documents are currently available as queried, and convey the reasons for the problem(s) via the RegistryError elements returned.

CONF-075: An XCA Initiating Gateway SHOULD, in the case of a Failure result in an ITI-39 response, convey to an end user that no documents were retrieved, and convey the reasons for the problems via the RegistryError elements returned.

CONF-076: An XCA Initiating Gateway SHOULD, in the case of a PartialSuccess result in an ITI-39 response, convey to an end user which documents were retrieved and which were not, and convey the reasons for the problems via the RegistryError elements returned.

CONF-077: An XCA Responding Gateway MUST detect the error conditions for the following ITI-38 error codes (see IHE ITI TF Vol3, section 4) and return those errors:

- XDSMissingHomeCommunityId (Informative: already required by IHE ITI TF-2b: 3.38.4.1.3)
- XDSStoredQueryMissingParam
- XDSStoredQueryParamNumber (Informative: already required by IHE ITI TF-2a: 3.18.4.1.3)
- XDSUnknownCommunity (Informative: already required by IHE ITI TF-2b: 3.38.4.1.3)
- XDSUnknownPatientId or return successful response with no elements (Informative: already required by IHE ITI TF-2b: 3.38.4.1.2.2)
- XDSUnknownStoredQuery (Informative: already required by IHE ITI TF-2a: 3.18.4.1.3)

CONF-078: An XCA Responding Gateway MAY detect the error conditions for the following ITI-38 error codes (see IHE ITI TF Vol3, section 4) and return those errors:

- XDSRegistryBusy
- XDSRegistryError
- XDSRegistryOutOfResources
- XDSTooManyResults

Informative**:** The existing requirements around ITI-38 error reporting are summarized here:

- IHE ITI TF-2b: 3.38.4.1.3 Expected Actions, requires Vol 2a: 3.18.4.1.3 Expected Actions.
- IHE ITI TF-2a: 3.18.4.1.3 Expected Actions, references IHE ITI TF-3: 4.2.4 Error Reporting.
- IHE ITI TF-3: 4.2.4 Error Reporting, describes how to format an error. Specifically, “location” is optional and contains “module name and line number or stack trace if appropriate.”
- IHE ITI TF-2b: 3.38.4.1.3 Expected Actions, states “every RegistryError element returned in the response shall have the location attribute set to the homeCommunityId of the Responding Gateway”. This requirement overrides the one in ITI TF-3: 4.2.4.

CONF-079: An XCA Responding Gateway, in the case of a combination of success and failure in an ITI-38 or ITI-39 transaction, MUST return a PartialSuccess result, if permitted by policy.

Informative: This is a restriction over the base requirement in 3.38.4.1.3 Expected Actions. Examples: when it is only able to provide some but not all documents available, or when it cannot assert whether all documents can be located, e.g., in the case of downtime of components of the network(s) that the Responding Gateway represents.

Informative: The policy allowance above is intended to permit hiding the fact that documents could not be returned for access policy reasons.

Informative: There is a gap in the requirements for ITI-39 error reporting. IHE ITI TF-2b: 3.39.4.1.3 Expected Actions, requires Vol 2b: 3.43.4.1.3 Expected Actions. However, this section pertains to the Initiating Gateway only. There is no reference to Vol 2b: 3.43.4.2.3, which requires the responding side

to report errors and which references IHE ITI TF-3: 4.2.4 Error Reporting. This gap is being addressed via a CP. In the meantime, the following error reporting requirements are added.

CONF-080: An XCA Responding Gateway MUST detect the error conditions for the following ITI-39 error codes (see IHE ITI TF Vol3, section 4) and return those errors:

- XDSDocumentUniqueIdError
- XDSMissingHomeCommunityId
- XDSUnknownCommunity
- XDSUnknownRepositoryId

CONF-081: An XCA Responding Gateway MAY detect the error conditions for the following ITI-39 error codes (see IHE ITI TF Vol3, section 4) and return those errors:

- XDSRepositoryBusy
- XDSRepositoryError
- XDSRepositoryOutOfResources

Informative: There is a conflict in the requirements for ITI-39 error reporting. IHE ITI TF-2b: 3.39.4.1.3 Expected Actions, states “Every RegistryError element returned in the response shall have the location attribute set to the homeCommunityId of the Responding Gateway”. However, IHE ITI TF-2b: 3.43.5 Protocol Requirements states “location contains the DocumentUniqueId of the document requested”. This conflict is being addressed via a CP. In the meantime, the following error reporting requirement allows for any reasonable interpretation.

CONF-082: An XCA Responding Gateway MUST, when returning RegistryErrors in an ITI-39 response, provide in the location attribute: the homeCommunityId of the Responding Gateway, the DocumentUniqueId of the document requested, or both.

8.7.8. Identifying Documents from Facilities Covered by 42 CFR Part 2

In this section, Carequality defines an optional capability to support the release and subsequent protection of sensitive information under 42 CFR Part 2. XCA Initiating Gateways indicate support for this option by asserting the policy identifier urn:oid: 2.16.840.1.113883.3.7204.1.1.1.1.14, specified in section 4.4.1, Access Policy Assertions. XCA Responding Gateways may then return sensitive information under 42 CFR Part 2 with special markings in the metadata and content regarding its handling.

This capability is derived from the IHE/HL7 Data Segmentation for Privacy (DS4P) capability, but does not represent a full implementation of DS4P.

CONF-117: An XCA Initiating Gateway supporting the 42 CFR Part 2 option MUST implement the following requirements in the IHE ITI Technical Framework, Rev. 13.0 Final Text 2016-09-09, except as constrained by this document:

- Volume 3, section 4.2.3.2.5: DocumentEntry.confidentialityCode

CONF-118: An XCA Responding Gateway supporting the 42 CFR Part 2 option MUST implement the following requirements in the IHE ITI Technical Framework, Rev. 13.0 Final Text 2016-09-09, except as constrained by this document:

- Volume 3, section 4.2.3.2.5: DocumentEntry.confidentialityCode

Informative: The more recent version of the ITI Technical Framework is necessary for confidentialityCode, as it explains how to use multiple instances of the attribute to indicate additional security and privacy tags.

CONF-119: An XCA Initiating Gateway MAY indicate support for the 42 CFR Part 2 option by asserting the policy identifier TBD, as specified in section 4.4.1, Access Policy Assertions.

CONF-120: An XCA Initiating Gateway that supports the 42 CFR Part 2 option MUST be able to parse and interpret all special markings as constrained by this section.

CONF-121: An XCA Responding Gateway MAY support the 42 CFR Part 2 option.

CONF-122: An XCA Responding Gateway that supports the 42 CFR Part 2 option, and receives in a request the policy identifier TBD, as specified in section 4.4.1, Access Policy Assertions, MAY return patient information available under 42 CFR Part 2, and for this information, MUST include special markings as constrained by this section.

8.7.8.1. Document content

CONF-124: An XCA Responding Gateway supporting this option and returning a document under 42 CFR Part 2 MUST use a document-level confidentialityCode of codeSystem="2.16.840.1.113883.5.25", code="R" in the CDA document.

CONF-125: An XCA Responding Gateway supporting this option and returning a document under 42 CFR Part 2 SHOULD include visual indicators and/or text in the narrative portion of the document notifying the reader about the protected nature of the content.

8.7.8.2. DocumentEntry.confidentialityCode

This option uses the HL7 Healthcare Privacy and Security Classification System (HCS). See IHE ITI TF Volume 3, section 4.2.3.2.5: DocumentEntry.confidentialityCode.

CONF-126: An XCA Responding Gateway supporting this option and returning a document entry under 42 CFR Part 2 MUST include a DocumentEntry.confidentialityCode instance with code="R" (Restricted) from the HL7 code system V:Confidentiality (@codeSystem=”2.16.840.1.113883.5.25”) to indicate the Confidentiality coding of the content.

CONF-127: An XCA Responding Gateway supporting this option and returning a document entry under 42 CFR Part 2 MUST include a DocumentEntry.confidentialityCode instance with code “ETH” (Substance abuse information sensitivity) from the HL7 code system V:InformationSensitivityPolicy

(@codeSystem="2.16.840.1.113883.1.11.20428"), to indicate the Sensitivity coding of the content. See https://www.hl7.org/fhir/v3/InformationSensitivityPolicy/vs.html.

CONF-128: An XCA Responding Gateway supporting this option and returning a document entry under 42 CFR Part 2 MUST include a DocumentEntry.confidentialityCode instance with code "42CFRPart2" from the HL7 code system ActCode (@codeSystem="2.16.840.1.113883.5.4"), to indicate the specific policy that applies to the content. See http://build.fhir.org/v3/ActCode/cs.html.

CONF-129: An XCA Responding Gateway supporting this option and returning a document entry under 42 CFR Part 2 MUST include the following DocumentEntry.confidentialityCode instances from the HL7 code system ObligationPolicyCode (@codeSystem="2.16.840.1.113883.1.11.20445"), to indicate the Obligation Handling Caveats of the content: one instance with code="PERSISTLABEL", and one instance with code="PRIVMARK". See https://www.hl7.org/fhir/v3/ObligationPolicy/vs.html.

CONF-130: An XCA Initiating Gateway that supports the 42 CFR Part 2 option MUST, if it persists a 42 CFR Part 2 document and its metadata, follow the handling obligations for "PERSISTLABEL", and "PRIVMARK" as specified in https://www.hl7.org/fhir/v3/ObligationPolicy/vs.html.

CONF-131: An XCA Responding Gateway supporting this option and returning a document entry under 42 CFR Part 2 MUST include DocumentEntry.confidentialityCode instances from the HL7 code system PurposeOfUse (@codeSystem="2.16.840.1.113883.1.11.20448"), to indicate the PurposeOfUse Handling Caveats of the content. Note: this value set can be found at: https://www.hl7.org/fhir/v3/PurposeOfUse/vs.html. The Purpose Of Use specified in the request from the Query Initiator MUST be represented in the returned list, using the following mapping between NHIN (i.e. eHealth Exchange) and HL7 V3 value sets:

HL7 V3 Purpose of Use	NHIN PurposeOfUse code
TREAT	TREATMENT
HPAYMT	PAYMENT
HOPERAT	OPERATIONS
PUBHLTH	PUBLICHEALTH
PATRQT	REQUEST
COVERAGE	COVERAGE

CONF-132: An XCA Initiating Gateway that supports the 42 CFR Part 2 option MUST, if it persists a 42 CFR Part 2 document and its metadata, restrict access to the document according to the Purposes Of Use specified in the document metadata.

9.0 Issues and Questions

The following issues and questions were considered and researched during the writing of this Implementation Guide. All issues have been resolved. The issue descriptions below are provided for background only; they do not constitute any additional normative requirements.

9.1. Open Issues and Questions

9.2. Resolved Issues and Questions

QUERY-001: Are the semantics of the use case alternate flow "Multiple matches returned within a given HCID" accurate? The use case, in accordance with guidance received from IHE ITI, states that each record represents a distinct patient, which must be disambiguated. However, eHealth Exchange imposes an additional constraint that of matches within a given HCID, those with different AAIDs represent multiple sources of data for the same person, not different people. It doesn't appear both of these interpretations can be true. Please also confirm the semantics for alternate flow "Multiple matches returned with different HCIDs".

We received the semantics for ITI through an email conversation with an ITI subject matter expert. We have posed this question both to the eHealth Exchange (http://exchange-specifications.wikispaces.com/share/view/72162420) and to the ITI Technical Committee (https://groups.google.com/forum/?hl=en#!topic/ititech/U0ZjjCv9fhU) for clarification.

In the interim, this Implementation Guide adopts the stated semantics.

QUERY-002: The XCA profile does not currently allow a Responding Gateway to return HCIDs other than the one it is associated with. We confirmed that there are existing production systems that count on this interpretation, and some that can successfully parse the non-conformant response. We analyzed this in detail and asked for clarification with the ITI Technical Committee: https://groups.google.com/forum/?hl=en#!topic/ititech/LWQywiHXANA. They would like to relax this requirement via a new CP. The Carequality Query WG discussed this and decided to keep to the current interpretation for now but to allow for graceful handling of the error.

QUERY-003: Initially, this Implementation Guide considered adopting the most recent revision of the IHE ITI TF (2014) along with all CPs in effect as of the 2015 NA Connectathon. However, in light of the great number of existing systems that have been implemented against the eHEX 2011 specification (which leverages the 2010 ITI revision), Carequality has instead opted to base this Implementation Guide on the 2010 (7.0) revision of the IHE ITI TF.

In addition, Carequality carefully considered the various approaches to versioning and governance. The resulting policy will be defined outside this Implementation Guide, and will cover issues such as:

- The ability of Carequality to maintain multiple versions of this Implementation Guide, each tied to potentially different versions of underlying specifications
- The ability of Carequality participants to advertise the version(s) they support for each endpoint in a directory
- The ability of a given endpoint to optionally support multiple versions
- Governance around how Carequality participants will conform to this Implementation Guide and/or different revisions

QUERY-004: Related to QUERY-003, QUERY-004 considered the set of CPs to adopt along with the ITI. The CPs were chosen to maximize interoperability, focusing on error fixes.

QUERY-005: There is a typo in section 3.55.4.2.2 of XCPD – it reads: “The Responding Gateway may specify a duration value in the SOAP Header element of the request". This should say "response". We have posted a question to the ITI Technical Committee and created a CP: https://groups.google.com/forum/?hl=en#!topic/ititech/9n2_ACZfp6I.

The CP is in progress. For the purposes of this Implementation Guide, the requirement shall read: “The Responding Gateway may specify a duration value in the SOAP Header element of the response”.

QUERY-006: What mechanism(s) will Carequality adopt for technical trust between systems?

Carequality is addressing this in a separate Technical Trust policy document.

QUERY-007: The new SOAP header block AccessDenial defined by this Implementation Guide, as well as the new MatchAlgorithm described in issue QUERY-018, both use the namespace: “urn:carequality”. This URN has not been registered with IANA, as it is intended for temporary use only. The long-term plan is that Carequality will write and submit a CP to IHE ITI to add the AccessDenial SOAP header block within the IHE namespace, and deprecate the use of this namespace.

QUERY-008: The XCPD request parameters MatchAlgorithm and MinimumDegreeMatch appear to be “hooks” for higher-level profiles/agreements to define, i.e. they do not have deterministic meaning defined by the XCPD profile. How should Gateways use these parameters to achieve maximum interoperability? Should they always omit them unless there is a higher-level profile defining how they are to be used?

Carequality is addressing patient matching requirements in a separate supplement, and will consider these questions then. For now, we have added draft text to omit them unless mutually understood, and have defined a single new algorithm.

QUERY-009: There is a slight imbalance between the type of the patient ID returned in an XCPD response, which is of HL7V3 II type, and the type of the patient ID passed in a XCA Cross Gateway Query request, which is of HL7V2 CX type. The CX type as defined in HL7 2.5.1 suggests length restrictions on the assigning authority (227 chars) and ID Number (15 chars), which are not imposed on the corresponding HL7V3 II root and extension.

This may cause interoperability problems with XCA Responding Gateways unable to process query requests, and/or XCA Initiating Gateways failing to send query requests, and is under active discussion with the IHE Technical Committee: https://groups.google.com/forum/?hl=en#!topic/ititech/12pmjUnMCu4.

Added informative background and conformance statement to ensure compatibility, and will propose a CP to ITI to clarify.

QUERY-010: It has been suggested that Carequality needs to incorporate lessons learned from eHEX and other exchanges, and enumerate the document content formats (or a common subset) that will be supported, as well as to map each content type to allowable XDS metadata values, initially taken from HITSP C80.

Although considered out of scope for this initial version of the Implementation Guide, Carequality plans to pursue this effort long-term, either by leading or by supporting SDO initiatives such as IHE DAF, as prioritized by the Steering Committee and coordinated with the Query Workgroup.

QUERY-011: Suggest we just start with Approved docs and not worry about on-demand docs. Are some exchanges using on-demand docs to a great extent? Because of MU CCDA requirements, won't the preponderance of docs be "stable" as created by EHRs? The answer affects the importance of issues 2, 3, 4, 5.

Resolution: We allow On-demand as an option for Responding Gateways and we know of many that use it, so we have added guidance and requirements for Initiating Gateways to support it to ensure the greatest interoperability.

QUERY-012: There is no requirement for an XCA Responding Gateway to detect and return a XDSStoredQueryMissingParam error.

Resolution: Added a requirement as well as informative guidance about it and other errors.

QUERY-013: There is some confusion regarding the location attribute in an ITI-38 error. Specifically:

- IHE ITI TF-2b: 3.38.4.1.3 Expected Actions, requires Vol 2a: 3.18.4.1.3 Expected Actions.
- IHE ITI TF-2a: 3.18.4.1.3 Expected Actions, references IHE ITI TF-3: 4.2.4 Error Reporting.
- IHE ITI TF-3: 4.2.4 Error Reporting, describes how to format an error. Specifically, "location" is optional and contains "module name and line number or stack trace if appropriate." See http://exchange-specifications.wikispaces.com/share/view/51470662
- IHE ITI TF-2b: 3.38.4.1.3 Expected Actions, states "every RegistryError element returned in the response shall have the location attribute set to the homeCommunityId of the Responding Gateway".

Resolution: Since the requirement in IHE ITI TF-3: 4.2.4 is optional, the one in IHE ITI TF-2b: 3.38.4.1.3 can override it. Added informative text.

QUERY-014: There is a gap in the requirements for ITI-39 error reporting. IHE ITI TF-2b: 3.39.4.1.3 Expected Actions, requires Vol 2b: 3.43.4.1.3 Expected Actions. However, this section pertains to the Initiating Gateway only. There is no reference to Vol 2b: 3.43.4.2.3, which requires the responding side to report errors and which references IHE ITI TF-3: 4.2.4 Error Reporting.

In addition, there is a conflict in the requirements for ITI-39 error reporting. IHE ITI TF-2b: 3.39.4.1.3 Expected Actions, states "Every RegistryError element returned in the response shall have the location

attribute set to the homeCommunityId of the Responding Gateway". However, IHE ITI TF-2b: 3.43.5 Protocol Requirements states "location contains the DocumentUniqueId of the document requested".

Resolution: We have submitted a CP to cover both of these: see https://groups.google.com/forum/?hl=en#!topic/ititech/u95UnHtY6tE. In the meantime, added error reporting requirements for ITI-39 including a forgiving interpretation of the location attribute.

QUERY-015: When an XCA Initiating Gateway does not support on-demand but a Responding Gateway does, there is a potential for clinical information to be missed. The Initiating Gateway will query for stable document entries only. The Responding Gateway may not have stable versions of some/all documents.

Resolution: Required XCA Initiating Gateways to support on-demand for Carequality.

QUERY-016: Carequality is adopting the XCA profile, which does not have a shared set of coded values or MIME types in document metadata. Should Carequality adopt some standards in the interest of interoperability? This question is closely related to whether Carequality should do the same when it comes to document content.

Resolution: The group decided to adopt HITSP C80, as well as the schemes to use, taken from the eHEX FAQ: http://exchange-specifications.wikispaces.com/Query+for+Documents+Home#Query. In addition, added guidance on what to expect, as "adoption" is a SHOULD, not a MUST. See also QUERY-010.

QUERY-017: Carequality needs to define full operational details for security and transport requirements.

Resolution: Decided as a group to adopt the eHealth Exchange Messaging Platform and Authorization Framework specifications as a start, and then capture only the ways where Carequality chooses to deviate from them. This also took care of potential incompatibilities between eHEX and Carequality.

QUERY-018: eHealth Exchange restricts ITI-55 responses to one patient ID per AAID. Because of this, eHEX Initiating Gateways may not be able to process multiple matches from the same AAID. See question QUERY-001 as well regarding the semantics of these matches.

To address this, the Implementation Guide has defined a new value for MatchAlgorithm that equates to the eHEX semantics. See also issue QUERY-007 which discusses the namespace.

QUERY-019: eHealth Exchange does not support the XCPD ITI-55 ambiguous match return codes.

Resolution: These codes are optional to return. Allowed Initiating Gateways to optionally treat the same as no match.

QUERY-020: Networks and systems may have different requirements for which demographic parameters are required and which combinations of matching parameters result in a patient match.

Resolution: Added requirements for Initiating and Responding Gateways to send as many demographics as possible to maximize matching potential.

QUERY-022: eHealth Exchange does not "make use of" the CorrelationTimeToLive SOAP header. This means Responding Gateways are not expected to understand that header.

Resolution: Added requirement for Initiating Gateways to not use a mustUnderstand value of "true" or "1". Added requirement for Responding Gateways making support optional.

QUERY-023: Can we use the ACP identifier to assert a policy of "the form posted for my organization in the Carequality Directory must be signed by the patient", with the query initiator able to assert this policy has been satisfied without making the form available via an IACP identifier? It would still be up to the Responding Gateway (and its Policy Engine) to make the call whether the ACP identifier alone is sufficient to grant access. We have posed this question to the eHealth Exchange Spec Factory: http://exchange-specifications.wikispaces.com/share/view/79038131.

Resolution: Yes.

QUERY-024: Should the SOAP fault UserNotAuthorized, defined by Carequality in version 1.0 of this guide, be kept for use in the Access Denied error flow? Further, should the new mechanism to report access denial details have both a SOAP fault flavor for full denial and a SOAP header block flavor for partial denial? Finally, should other SOAP faults such as wsse:FailedAuthentication still be permitted to be used for access denial?

Resolution: The working group decided to simplify all variants of access denial to use a single, newly defined SOAP header block AccessDenied, which was necessary to support the partial denial case. All other mechanisms of reporting access denial were deprecated.

QUERY-025: The requirements for formatting the XDS document unique id for a Patient Permission document are in conflict. IHE PCC Technical Framework Volume 2, section 4.1.1, specifies an XPath expression that would include the caret "^" when there is no extension value in the CDA document id. IHE ITI Technical Framework Volume 3, Table 4.1-5 Document Metadata Attribute Definition, says not to include the caret if there is no extension. Which requirement is valid?

Resolution: After consulting with an IHE SME, we are adopting the ITI definition, which is now in final text, and we will submit a CP against the PCC Technical Framework. We also added a clarifying requirement.

QUERY-026: Should there be general policy constraining the ability of the Responding Gateway to inherently trust an instance access consent policy (IACP) asserted by the Initiating Gateway? For example:

1. No inherent trust; must retrieve and evaluate for each transaction
2. May inherently trust without ever retrieving
    - Assumes Initiating Gateway never asserts expired policy – we have added Carequality policy to enforce (must be valid for at least 24 hours)
    - Assumes policy is known externally vs. computed from document (e.g. XACML)
3. May trust based on prior retrieval

    - Example: retrieve/evaluate during Patient Discovery and decline to retrieve for immediately subsequent Query and Retrieve Documents that assert the same IACP
4. No general policy: this will be defined per specific policy identifier
5. This is an issue of local autonomy; remain silent

Resolution: This had been discussed by the policy group, who opted for the final option, to remain silent.

QUERY-027: This guide enumerates the allowed formats for Patient Permission documents. Is there a need for XDS-SD, i.e. scanned documents with XDS metadata bindings and an unstructured CDA but no Patient Permission info? Is there a need for "bare" PDFs, that is, PDFs that aren't wrapped in a CDA?

Resolution: Not at this time. XDS-SD is very close to BPPC-SD, and what BPPC-SD adds is very appropriate and valuable. Have not identified a need for bare PDF.

QUERY-028: Section 4.4.2, Requirements for Query Responders, discusses the 42 CFR Part 2 case, and seems to limit both the access decision and assumption of data protection to the level of HCID, even though there are finer-grained identifiers in the SAML assertion that identify the requester: Subject Organization ID and optional National Provider Identifier (NPI). Are responders allowed to utilize these finer-grained values? For example: Doctor A (who has an NPI) works in department B (which has an org ID) in organization C (which has a HCID). Could the responder limit dissemination to this doctor and still be conformant?

Resolution:

The preferred way to handle finer-grained disclosure makes use of the HCID and the Carequality directory to reflect organizational hierarchies. This way is preferred because this information is searchable and transparent.

In the original example, department B and organization C could each have their own HCID in the directory, and both could point to the same endpoint. Now the requester can make a request from department B and be assured it will be protected accordingly. Note that this still doesn't allow for protection down to an individual.

QUERY-029: (Question submitted from reviewer) Section 8.4.5 – We suggest that as part of a maintenance review to consider the purpose of having a Carequality specific fault/code in a cq: namespace. NHIN does not specify one, neither does SOAP, so perhaps we do not really need one either. We suggest that using section 3.4.6. on error reporting in the SAML Bindings 2.0 spec would be sufficient.

Resolution: We have deprecated the Carequality-defined fault that had been in section 8.4.5. However, we have replaced it with a new SOAP header block that does still have a Carequality-defined "error code". This is for two reasons:

- Because we designed the SOAP header block to follow the Code/Detail pattern of SOAP faults, and because we designed a new structure to convey policy needs, we needed a code that would indicate the structure within the Detail element.
- Existing SOAP fault codes such as wsse:FailedAuthentication were not sufficient for our needs. If we had decided to use the SOAP Fault structure for full access denials, that allows for a code and a subcode. The code would have to have been Sender, and we would have needed our new code as the subcode to indicate the structure within the Detail element.

Further, that SAML Bindings section pertains to SAML requests/responses. We do not use that pattern. Rather, we use the SOAP Message Security 1.1 and SAML Token Profile 1.1 pattern where the SAML assertion is within the WS-Security header.

Finally, use of this code is optional; Initiators may ignore it.

QUERY-030: This guide generally leverages the IHE ITI Technical Framework from 2010. Since that time, an error has been corrected in the BPPC profile, concerning the policy identifiers being acknowledged. In the 2010 version, the BPPC document does not contain the policy identifiers, while in the latest, it does, in /ClinicalDocument/documentationOf/serviceEvent[templateId/@root='1.3.6.1.4.1.193 76.1.5.3.1.2.6']/code/@code. Which version of BPPC is adopted?

Resolution: Carequality adopts the latest published version: Revision 13.0, September 9, 2016. In addition, because BPPC references the IHE PCC Technical Framework (and no other Carequality requirements do), we have referenced the 2016 revision of PCC as well.