Excerpt from http://open-services.net/bin/view/Main/OslcCoreSpecification?sortcol=table;up=#Query_Capabilities to show full context of the proposed changes.

Query Capabilities

An OSLC Service may provide one or more Query Capabilities to enable query of resources. A Query Capability provides a base URI for forming Query Resource URIs and MAY provide Resource Shapes that describe the property values that may be expected in the resources that are queryable via the query capability. Thus, Query Capabilities provide a way to discover the resources managed by an OSLC Service.

In a Query Capability, the base URI, as defined by the oslc:queryBase property, is itself a resource managed by the service and it acts as the starting subject resource for the queries based on it. Since the list may contain hundreds of thousands of members, queries are used to filter the list for members that satisfy certain conditions, e.g. the bugs that have high priority and were created this week.

Conceptual Model

To perform a query an OSLC client first creates a URI by starting with a Query Capability's base URI as a base and adding a URI Query String to express the query criteria. The OSLC client then uses HTTP GET to request a Query Resource representation of the query results. The Query Resource representation will contain property values about the query and a collection of resources that match the query criteria.

HTTP GET Queries

To perform an HTTP GET query, an OSLC client starts with the base URI as defined by the oslc:queryBase property of a Query Capability, and appends to it query parameters in a syntax supported by the service. The resulting URI is the query URI. The OSLC client sends an HTTP GET request to the query URI, optionally specifying the preferred content media type for the query response in the HTTP Accept header. OSLC Services MUST support query responses in RDF/XML format (media type application/rdf+xml) and MAY support other formats. OSLC Services SHOULD support the Query Syntax defined in this specification, but MAY support other syntaxes.

Although HTTP does not limit a URI's length, web browser and server implementations do impose limits. HTTP status code 414 (Request-URI Too Long) exists to inform clients when the request URI is longer than the server can handle, and OSLC Services are expected to return 414 per HTTP. OSLC Clients receiving a 414 response to a request using a query URI have the option to re-form the request as a HTTP POST Query, as described below.

Comment [JA1]: See http://tools.ietf.org/html/rfc2616#section-3.2.1 paragraph 2

HTTP POST Queries

OSLC Services MAY support queries via HTTP POST, in order to support requests that would otherwise result in 414 status codes, as described in [HTTP GET Queries]. To perform an HTTP POST query, an OSLC client uses the query's base URI as defined [HTTP GET Queries] as the query URI, and form-encodes the query parameters in the HTTP request body instead of appending them to the query's base URI. The OSLC client sends an HTTP POST request to the query URI, optionally specifying the preferred content media type for the query response in the HTTP Accept header. The HTTP Content-Type header of the request MUST be application/x-www-form-urlencoded, and the message body MUST contain the form-encoded URL query parameters. Aside from the HTTP method, request Content-Type header, and the location of the query parameters, all requirements for HTTP GET queries apply to HTTP POST queries.

If the same URI is used for both query and creation capabilities, then a HTTP POST request with Content-Type application/x-www-form-urlencoded MUST be interpreted by OSLC Services as a query, and that request MUST NOT create a resource.

Caching effects: HTTP POSTs are not cacheable by default in HTTP. OSLC clients are strongly RECOMMENDED to query using HTTP GET first, and fall back to POST only if the GET fails with a 414 status code.

Query Syntax

A query URI can be formed by adding a query string to the end of the Query Capability's base URI (or by sending the query string in the request body when using HTTP POST). The syntax used to express the query criteria in that string is specified by each OSLC domain specification.

The OSLC Core Spec Query Specification document defines a standard set of OSLC query parameters that other OSLC domain specifications MAY use to query resources.

Query Specification errors

If there is an error in the specification of the query, whether the query is specified by key=value pairs in the HTTP GET URL or key=value pairs in the body of an HTTP POST, then the provider MUST respond with an error. The error response should be an HTTP 400 Bad Request error and an explanation of the error in the OSLC Error Response format (see Error Responses below).

Delegated User Interface Dialogs