INIT1SD2.pm7 - CQL 2.0 Enhancements

Contents of this Page

Get status of query process

Use Case Number	INIT1SD2.pm7.1
Brief Description	When a long-running query is submitted to a number of different federated services, it is desirable to be able to provide users with a status update on the progress of the query, including which services have completed, how much data has been returned, and (if possible) how much work is left.
Actor(s) for this particular use case	Cancer Researcher
Pre-condition **The state of the system before the user interacts with it**	A number of services are available upon which a federated query can be performed.
Post condition **The state of the system after the user interacts with it**	The user is updated with the status of the query.
Steps to take **The step-by-step description of how users will interact with the system to achieve a specific business goal or function**	1. The user constructs a federated query and submits it to multiple services 2. The user periodically requests status updates on the progress of the query from the federated query processor 3. The user receives information on which services have completed processing the query, how much data has been processed, and/or how much work/time is required to complete the query
Alternate Flow **Things which would prevent the normal flow of the use case**	None.
Priority **The priority of implementing the use case: High, Medium or Low**	Medium.
Associated Links **The brief user stories, each describing the user interacts with the system for the one function only of the use case. There would potentially be a number of user stories that make up the use case.**	- INIT1SD2-CQL 2.0 Enhancements - CQL 2.0 Enhancements - CQL 2 Use Cases
Fit criterion/Acceptance Criterion **How would actor describe the acceptable usage scenarios for the software or service that meets the actor's requirement?**	Minimally, the user should be informed on which services have completed execution of the query, which are still working on the query, and how much time has passed for each service.

Get intermediate query results

Use Case Number	INIT1SD2.pm7.2
Brief Description	After receiving a status update that a federated query has completed on some services, the user may want to fetch and view the results from the completed sub-queries.
Actor(s) for this particular use case	Cancer Researcher
Pre-condition **The state of the system before the user interacts with it**	A federated query has been submitted to multiple services and a subset of them have completed.
Post condition **The state of the system after the user interacts with it**	The user receives intermediate data from the completed sub-queries.

Steps to take The step-by-step description of how users will interact with the system to achieve a specific business goal or function	1. The Cancer Researcher finds from a status update that a subset of queries have completed from his federated query 2. The Cancer Researcher selects the services that he'd like to fetch the intermediate results from 3. The Cancer Researcher fetches those results and views them while the other sub-queries are being completed by the other systems
Alternate Flow Things which would prevent the normal flow of the use case	None.
Priority The priority of implementing the use case: High, Medium or Low	Medium.
Associated Links The brief user stories, each describing the user interacts with the system for the one function only of the use case. There would potentially be a number of user stories that make up the use case.	• INIT1SD2-CQL 2.0 Enhancements • CQL 2.0 Enhancements • CQL 2 Use Cases
Fit criterion/Acceptance Criterion How would actor describe the acceptable usage scenarios for the software or service that meets the actor's requirement?	The Cancer Researcher can view intermediate results in order to determine whether to continue the processing of the federated query.

Cancel portions of a query

Use Case Number	INIT1SD2.pm7.3
Brief Description	After reviewing intermediate results or after a federated query takes too long to complete at some number of services, the Cancer Researcher may choose to cancel the processing of the query at any number of the services.
Actor(s) for this particular use case	Cancer Researcher
Pre-condition The state of the system before the user interacts with it	A federated query is running at a number of services.
Post condition The state of the system after the user interacts with it	The selected services have canceled processing of the federated query, and data is available at all services that have completed
Steps to take The step-by-step description of how users will interact with the system to achieve a specific business goal or function	1. The Cancer Researcher views the list of services that have completed and are processing the federated query, as well as the time spent processing the query. 2. The Cancer Researcher selects those services for which he wants to cancel the query 3. The services cease to process the query
Alternate Flow Things which would prevent the normal flow of the use case	The Cancer Researcher sets a time limit to set on the query, and the status update returns whether the time limit has been met and the query has been canceled (see use case below).
Priority The priority of implementing the use case: High, Medium or Low	Medium.
Associated Links The brief user stories, each describing the user interacts with the system for the one function only of the use case. There would potentially be a number of user stories that make up the use case.	• INIT1SD2-CQL 2.0 Enhancements • CQL 2.0 Enhancements • CQL 2 Use Cases
Fit criterion/Acceptance Criterion How would actor describe the acceptable usage scenarios for the software or service that meets the actor's requirement?	Query processing is completed totally, but the status of canceled remains for some period of time (as determined by the service or the query itself).

Interrogate a service for additional data information

Use Case Number	INIT1SD2.pm7.4
Brief Description	Before issuing a federated query, the user may want to determine which services to query based upon criteria such as how many data objects are stored within the service, etc.
Actor(s) for this particular use case	Cancer Researcher

Pre-condition The state of the system before the user interacts with it	A number of services are available upon which a federated query can be performed.
Post condition The state of the system after the user interacts with it	The user has extra information on the data stored by each service referenced by the federated query.
Steps to take The step-by-step description of how users will interact with the system to achieve a specific business goal or function	1. The Cancer Researcher constructs a federated query 2. The Cancer Researcher views a list of the services available to issue the query upon 3. The Cancer Research interrogates each service to determine whether to include it, gathering information about the number of data items, available data, etc.
Alternate Flow Things which would prevent the normal flow of the use case	None.
Priority The priority of implementing the use case: High, Medium or Low	Low.
Associated Links The brief user stories, each describing the user interacts with the system for the one function only of the use case. There would potentially be a number of user stories that make up the use case.	- INIT1SD2-CQL 2.0 Enhancements - CQL 2.0 Enhancements - CQL 2 Use Cases
Fit criterion/Acceptance Criterion How would actor describe the acceptable usage scenarios for the software or service that meets the actor's requirement?	It is necessary that at least the number of data objects is returned, but it would also be desirable to know which data elements from the information model have associated data.

Specify complexity of federated joins

Use Case Number	INIT1SD2.pm7.4
Brief Description	Before issuing a query, a user may want to specify constraints on it that limits its complexity in order to make sure that it completes in satisfactory time.
Actor(s) for this particular use case	Cancer Researcher
Pre-condition The state of the system before the user interacts with it	A number of services are available upon which a federated query can be performed.
Post condition The state of the system after the user interacts with it	The federated query is augmented with complexity limits
Steps to take The step-by-step description of how users will interact with the system to achieve a specific business goal or function	1. The Cancer Researcher constructs a federated query and selects the services to run it against 2. The Cancer Researcher determines the factors that should be limited when running the query, such as time limit, a limit to the number of data items returned, and whether cross-organization joins should be performed (e.g. to limit the joins between services within an organization rather than across organizations)
Alternate Flow Things which would prevent the normal flow of the use case	The Cancer Researcher views the default complexity limit factors (from the query tool and from the services themselves) and modifies them if desired.
Priority The priority of implementing the use case: High, Medium or Low	Medium
Associated Links The brief user stories, each describing the user interacts with the system for the one function only of the use case. There would potentially be a number of user stories that make up the use case.	- INIT1SD2-CQL 2.0 Enhancements - CQL 2.0 Enhancements - CQL 2 Use Cases
Fit criterion/Acceptance Criterion How would actor describe the acceptable usage scenarios for the software or service that meets the actor's requirement?	Minimally, a user must be able to determine that joins across different services remain within an organization rather than go across organizations (because there would not be common identifiers).

Customize complexity of supported queries

Use Case Number	INIT1SD2.pm7.5

Brief Description	Some queries can be intractable or impractical to complete. When the level of complexity of these types of queries is well understood, it is important that the Software Engineer is able to configure the query processor to handle them appropriately. This includes the amount of data that can be returned, the length of time that can be spent processing a query, and the number of joins that are allowed.
Actor(s) for this particular use case	Software Engineer
Pre-condition **The state of the system before the user interacts with it**	A service has been developed that supports queries.
Post condition **The state of the system after the user interacts with it**	The service is configured to handle complex queries appropriately.
Steps to take **The step-by-step description of how users will interact with the system to achieve a specific business goal or function**	1. The Software Engineer configures the handling of different aspects of query complexity, including the amount of data that can be returned, the length of time that can be spent processing a query, and the number of joins that are allowed 2. The Software Engineer deploys the service
Alternate Flow **Things which would prevent the normal flow of the use case**	The Software Engineer changes the configuration of a deployed service.
Priority **The priority of implementing the use case: High, Medium or Low**	Low
Associated Links **The brief user stories, each describing the user interacts with the system for the one function only of the use case. There would potentially be a number of user stories that make up the use case.**	• INIT1SD2-CQL 2.0 Enhancements • CQL 2.0 Enhancements • CQL 2 Use Cases
Fit criterion/Acceptance Criterion **How would actor describe the acceptable usage scenarios for the software or service that meets the actor's requirement?**	Minimally, the number of joins and amount of data returned should be supported.