

**Title** FCS 2.0 work plan  
**Version** 1.1  
**Author(s)** Leif-Jöran Olsson (CLARIN ERIC > Språkbanken)  
**Date** 2015-09-09  
**Status** Draft  
**Distribution** public  


---

**ID** CE-2015-xxxx



## 1 Planning

Month	1 - Septem- ber	2	3	4	5 - Januar y	6	7	8
Developme nt plan	X							
Specificatio n FCS-QL 2.0	X	X						
Implement ation FCS- QL 2.0	X	X						
Aggregator integration preview				X				
Client library preview				X				
Conversion API preview					X			
Advanced DataView	X	X	X	X	X	X		
Stable release								X
FCS taskforce meeting		X	X	X	X	X	X	X
FCS taskforce videoconf	X	X	X	X	X	X	X	X

## 2 Work

The Federated Content Search (FCS) task consists of two main parts that are being developed: the FCS-QL *grammar specification v2.0* and the implementation of the *libraries and their integrations* into applications (endpoints) and the FCS Aggregator.

Most of the work described in this work plan is related to the FCS-QL specification v2.0 and the supporting libraries and the conversion API. Some work is also invested into the advanced DataView for the transport of query results from the clients to the Aggregator. Since there is a strong decoupling, changes on any component can be carried out

without affecting other components. This requires testing both on integration level and also backed and ensured by an extensive test suite, on unit level.

## 2.1 FCS-QL specification 2.0

To advance from the legacy and 1.0 versions of the QL/API we have been gathering use cases and come up with a draft version in a community fashion driven by actual use cases.

the FCS-QL *grammar specification v2.0*, will in short be more expressive and more CQP-like. By being more expressive it will hopefully be used for more than word or phrase searches. By making it more CQP-like more service users will recognize and hopefully use it in more advanced ways. It also makes the conversion work by the endpoint developers/admins easier since many uses CQP internally. Both points contribute to the possibility to cover more use cases from the end users we want to reach.

Extensive tests on unit and integration level is developed to make sure we cover the use cases brought up in the community involvement process. Test on these levels are essential to back and ensure the functionality once the QL grammar v2.0 is agreed upon.

The QL grammar v2.0 is being implemented in antlr4 with a primary java API. This is currently based on the draft version which means the implementation will be finalized shortly after the QL specifications is agreed on by end of month 2015-10 (Milestone 1) and the implementation of the *libraries and their integrations* into applications (endpoints) and the FCS Aggregator will follow shortly.

## 2.2 Aggregator integration

The legacy and v1.0 APIs, though marked as deprecated, will work in parallel with version 2.0 so that endpoints can make the migration at their earliest convenience based on local conditions. The integration testing of the library v2.0 with the Aggregator will start month 2015-12.

## 2.3 Client library

The client java library for the endpoints will be available for testing with the Aggregator by month 2015-12.

## 2.4 Conversion API

Eventhough the FCS-QL v2.0 is more CQP-like there will still be need for conversions to the local query languages. This will be supported on the API level with rule based conversions.

## 2.5 Releases

The release cycle steps for each of the components will basically be the same, e. g. preview, release candidate (RC), and stable release. The FCS endpoint community will be heavily involved in recurring video conferences (every 2-3 weeks) and other feedback processes to advance in the release cycle. Multiple release candidates might be needed if issues are found. When any release candidate is found to be stable and complete, it will be marked stable and deployed for production latest by end of month 2016-04.