Market Requirements Document

Feature Name: Objectivity/30 Proof of Concept

Version: 0.1	Date Submitted: 3/25/15	Completed By: Brian Clark
Version: 0.2	Date Submitted: 4/13/15	Completed By: Brian Clark

Description of the Problem

Project ‘Purple’ is bringing Objectivity products into the “Big Data” space leveraging a number of existing open source technologies, some of which we are familiar with but most are new to us. To help validate the approaches we are taking it will be necessary to incrementally build one or more proof of concepts (POC) where we can validate and/or try out alternatives. The POC should include elements of what we know, what can show off Objectivity’s products strengths, and new features and capabilities as they are introduced.

This MRD will set out requirements for any proof of concepts.

Background

The traditional formal development model in Objectivity is to specify market requirements in documents known as MRD (Market Requirement Document). Once these are agreed more detailed requirements may be specified in a Software Requirements Document. Then engineering produce a detailed Function Requirements Specification document.

While this formal model works well for most engineering activities we need something more agile during the investigative and exploratory stages of the ‘purple’ project (MVP). It is expected that building any POC will lead to formal requirements for the ‘purple’ product GA version and the corresponding functional specifications so that engineering can build the GA product.

Description of the Requested Feature

Not all of these requested features will be available in the first release, but should be added incrementally as new features become available. It is highly likely that some new features will be prototyped before they become product. As they become product they should be replaced in the POC.

1. The first POC will be based on the current work with CGG for the PNEC demonstration; primarily the HDFS support work. This could be extended by exposing the built-in workflow (to be investigated with CGG). This is very specific to CGG and the Oil & Gas Industry, so we will work in parallel on a second POC with more general applicability.

   The following requirements are for the second POC:
2. The POC should support a ‘realistic’ workflow use case. The high-level use case is ingest, process, output. The workflow components can be Java main classes.
3. The POC should use a model we are familiar with, e.g. Link Hunter. The POC will be based off OFJ implementation of the Link Hunter (persons, phone and calls subset), not InfiniteGraph.
4. The POC should demonstrate strengths of Objectivity products e.g. relationship analytics (how are these two things connected in any possible way?); scalability; distribution; cloud deployment.
5. The POC should demonstrate integration with a workflow engine (Apache Oozie or other (Knime)??)
6. The POC should demonstrate integration with HDFS.
7. The POC should demonstrate integration with a resource scheduler/monitor (YARN, or Mesos).
8. The POC should include use of Objectivity/RDD capability.
9. The POC should include use of the Objectivity/REST server.
10. The POC should include use of the meta data repository.
11. The POC should include elements of the Objectivity/Administration Console as they become available.

Part of an existing feature or does it require another feature, if so, which one?

- New POC dependent on components of ‘purple’.

How is this problem being solved now, and why isn't that acceptable?

- New problem being addressed and resulting solution.

What languages must support this capability?

- Initially Java.

Which platforms must be supported?

- POC should run on Linux, Windows and Mac OS X.

Do any competitors already have this feature?

- Major implementations of Hadoop (MapR, Cloudera, and Hortonworks) support running in this type of environment.

Customers who require this feature

- Some existing customers e.g. CGG, new customers in the 'big data' space.

Revenue at risk, or which could be won

- Could lead to more early adopters.

When is this required?

- Increments through 'Purple' MVP end of June.

Additional Notes

1. **Implementation notes:**
    a. CGG demonstration
    b. Need to prioritize what features are needed for each increment.

2. **Related Material**
   We will also need:
   Field Training.
   Quality Assurance.

3. **Software requirements**
    a. Objectivity/HDFS integration [http://objyshare.objy.com:8080/display/PM/MRD_Objectivity_HDFS_Integration_V0.1]
    b. Objectivity/RDD [http://objyshare.objy.com:8080/display/PM/Spark+Connector+MRD]
    c. Objectivity/REST Server [http://objyshare.objy.com:8080/display/PM/MRD_Objectivity_REST_Server_V0.5]
    d. Objectivity/Administration Console [http://objyshare.objy.com:8080/display/PM/MRD_Objectivity_Administration_Console_V0.1]
    e. Apache Spark [https://spark.apache.org]
    f. Apache Oozie [http://oozie.apache.org]
    g. YARN [http://hadoop.apache.org/docs/stable/hadoop-yarn/hadoop-yarn-site/YARN.html]
    h. Cloudera Quickstart [http://www.cloudera.com/content/cloudera/en/downloads/quickstart_vms.html]
    i. Hortonworks Sandbox [http://hortonworks.com/products/hortonworks-sandbox/] [http://docs.hortonworks.com/HDPDocuments/HDP2/HDP-2.1.5/bk_dataintegration/content/ch_using-oozie.html]
    j. MapR Sandbox [http://doc.mapr.com/display/MapR3/MapR+Sandbox+for+Hadoop#MapRSandboxforHadoop-InstallingonVMwarePlayer/VMwareFusion]

4. **Hardware Requirements**
    a. Hadoop cluster