20141107 Outline

Tuesday, October 7, 2014　　9:20 AM

<Name> Documentation

1. What is <Name>?

1. Overview of Application:
2. Purpose, high level summary of the main problem we solve (for perspective of: open source developers). Developed for open source use... Scala and Json...fully customizable, etc.

3. Capabilities
   a. Deployment Options - Kafka and file right now
   b. Fault Tolerance - Exactly-once   (using Zookkeeper (manages node availability))
   c. <add concepts in capabilities)
   d. POSS: Data Management, Model Management, Insight Management

4. Currently in Use Examples

2. Getting Started

1. New User Guide
2. *c*
3. *Open Source License (?)*
4. Tutorials

3. Developer Reference

1. Overview and Concepts
   a. Summary
      i. <Name> achieves near real-time processing of transactional data stored in the Hadoop Distributed File System. It does this by: <tbd>
2. Architecture
   a. Overview (and top level Diagram)
   b. Architecture
      i. Inputs and Outputs: Messages and Alerts
         1) Diagrams
      ii. Description of Engine Processing Flow
         1) Diagram
      iii. Rules as Models
         1) Diagrams
      iv. Why we get the speed/results we do
   c. Benefits and Highlights
      i. Key Technologies
      ii. Cluster/Node Solutions
      iii. Performance and Resource Utilization
      iv. Scalability, Availability, Optimization
3. API Documentation
   a. Installation Guide
      i. Download the <Name> installation file here
      ii. Installation steps
      iii. *Access steps - Keys and accounts*
   b. Data Structures
   c. Function Signatures

     i. Names
     ii. Function parameters names and types
        1) Return types
     iii. Error handling
     iv. Pre/post conditions
     v. Description of state changes post-execution
        1) tbd: compilers…

d. Objects
     i. Relationship of type to other types (inheritance: super-types, sub-types, implemented interfaces or traits), composite structures, delegating entities or any mixed-in set of functionality
     ii. Public parts of an instance of a class:
        1) Constants
        2) Name and type of properties
        3) Method signatures
        4) *Class-specific operators? (is overloading supported)*
        5) *Static nature?*
     iii. Constraints on objects derived from this class
     iv. Nested structures
     v. Object exception handling
     vi. Concurrency Handling

e. Metadata API (if different from the Objects section above)
     i. PMML Extensions
        1) PMML
     ii. Object model diagram
     iii. Elements
        1) Relationship to other Types
        2) Public Parts
        3) Constraints
        4) Nested Structures
        5) Exception Handling
        6) Concurrency Handling
     iv. Models
        1) *<sic>*
     v. Messages
        1) *<sic>*
     vi. Containers
        1) *<sic>*
     vii. Concepts
        1) *<sic>*
     viii. Fields
        1) *<sic>*

f. Communication protocols

g. Auditing
     i. Creating an output message that collects all input, outputs and intermediate fields from engine which can then be persisted to create an audit log.

h. *Unstable parts?*

i. *Deprecated parts?*

j. *Frameworks?:*
     i. *GUI*
     ii. *Network*
     File System

k. *How to Deploy Client Libraries*

l. Public Data Sets by Industry
     i. Finance
     ii. Healthcare
     iii. Diverse

m. Glossary

n. Troubleshooting
     i.

4. *SDK Tools / Documentation*
   1. *Tool 1*
      a. *Install and User guide for Tool 1*
   2. *Tool 2*
      a. *Install and User guide for Tool 2*

Release Notes

- *2014 1215 Release Notes*
- *20141201 Release Notes*
- *20141115 Release Notes*

Learning

Step by Step Guide - Create your First <Name> Project
Customization Guide

Licensing Options *(this will likely be somewhere else on our github page, but can keep links to it here)*

Free Version
Enterprise Version
Services