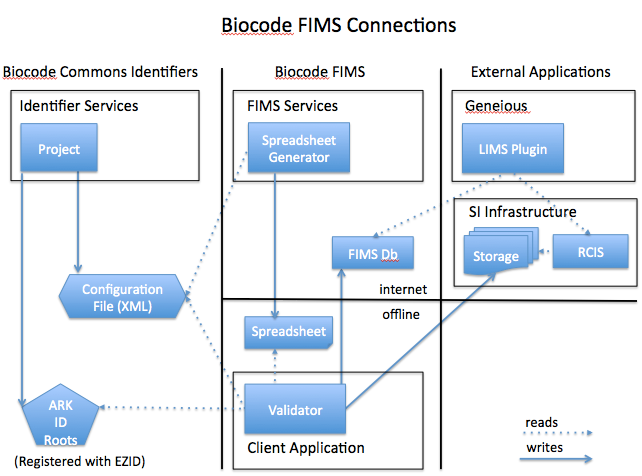
**Biocode-FIMS scoping meetings for SI-NMNH deployment**

**2/7/2014**

Tom, Jamie, Mike, Beth, John (along with others throughout the week leading up to this)

**How the system works:**

The biocode-FIMS is controlled using a configuration file, written in XML, which tracks the following information: 1) mappings between column names and standards-based term URIs, 2) validation rules, and 3) defining a path for normalizing the flattened data-sheet. The following diagram shows the various connections in the Biocode FIMS system:



**Actions**

1. (John, with help from Mike) Build the spreadsheet generator that reads the XML Configuration File and the user selects a series of options.
   1. Create the ability to specify “preferred templates” with pre-checked boxes for botany, iz, etc…
   2. First iteration will be stateless… that is, user customizations (other than the canned templates) will not be saved. This can be added later.
   3. Built using the spreadsheet template generator that Mike Trizna has built as a foundation.
2. (John) Embed a configuration file pointer in downloaded spreadsheets (an ARK identifier that resolves to the configuration file itself).
3. (John) Build a Java Client Application for data loading
4. Cache XML configuration file, and update it whenever there is an internet connection.
5. (Jamie/Tom) Specify new rules, come up with EMU mappings & mappings to standards along with definitions.
6. (John) Create a Tissue object HASH identifier to key spreadsheet off of (NOT a user-assigned identifier but built off relevant fields)
7. (John) Handle application updates… when there is a new version available prompt user to get it.