~ Methods of debugging ~

i2b2 offers the following approaches for debugging installation errors.

1 - Webclient XML Log

The webclient XML log shows errors that occur in client/server messaging in a user-friendly interface and should often be the starting point for debugging.

1. Verify the debug log is turned on. Make sure **debug: true** appears in **i2b2_config_data.js** in the webclient installation (see Step 3 of the Web Client Install Guide for an example of the correct file).
2. Now, the webclient will have a Message Log button in the upper right corner of the screen.

The Message Log allows users to view debug messages if the webcient is in debug mode. This shows the XML messages that pass between the client and server. The Message Log can be opened via the link in the upper right corner of the screen, or via any of the buttons (which will open a filtered view showing only messages relevant to the view from which it was selected). Debug mode can be enabled by an administrator, by modifying the i2b2_config_data.js file.

---

2 - Server Logging

This is a good second step in debugging, as it shows the detailed errors and warnings that occur on the server. It is the most detailed method, though it also tends to be verbose.

---

3 - Webclient Javascript Log

If the error is happening in the client, enabling the Javascript log can be a helpful approach.

1. In the webclient directory, edit default.htm and comment out line 30, which reads: `<script type="text/javascript" src="js-ext/firebug/firebugx.js"></script>`
2. Reload the page, open the javascript console in your web browser and watch for errors. (This process varies by browser. Here is, for example, the method for viewing Javascript errors in Chrome.

---

4 - Other Approaches

1. **POSTing REST messages.** It is possible to POST requests directly to the server, which allows more detailed debugging of REST messages than the webclient offers. Postman is a useful, free tool for this. See https://www.google.com/url?q=https%3A%2F%2Flearning.postman.com%2Fdocs%2Fpostman%2Fsending-api-requests%2Frequests%2F&sa=D&sntz=1&usg=AFQjCNGe9rABMx809bQXVsuGbpG5zsFLeA
2. **Eclipse Server Logging.** It is possible to run the i2b2 server interactively in Eclipse, set breakpoints, and monitor the log output from within the application. In brief, the approach is:
   a. *Prerequisites*: a local installation of the i2b2 server, and an install of Eclipse with the i2b2 source code checked out (from Git) in a project.
   b. Install the JBOSS tools in Eclipse https://tools.jboss.org/downloads/jbosstools/2019-03/4.11.0.Final.html
   c. Click the server icon in the upper right corner, and set up Wildfly 14 (or whatever version you are using).
   d. *Optional*: to support breakpoints, rebuild i2b2 with debugging turned on. In each cell's build.xml, search for the compile target and set debug to true (e.g., `<javac destdir="${classes}" optimize="${javac.opt}" debug="true"> )`. Then rebuild and redeploy each cell using the ant plugin for Eclipse.