

Debugging in the Admin Console

The Debugging section of the admin console at **Debugging > Debugging** controls debugging options as well as the execution of Java scripts.

Compare Knowledgebases

The Compare option is used to compare the structural differences between two selected KBs.

Any two KBs located on the same server may be compared by defining one of them as the source KB, and the other one as the target KB. The comparison will produce an Excel report listing all of the differences, as a downloadable link. Differences are organized by the type of item, i.e. Tables, Fields, Choices, Rules, Actions, Searches, Groups, Teams, Email Templates, Data Conversion, and Views. These items are displayed on separate tabs in the Excel file. If either the target or the source KB is broken and cannot be compared correctly, the system will report this damage so that the admin can contact Agiloft Support for assistance.

Start/Stop Debugging

To start debugging, click the Start Debug Output button. Debugging is automatically turned off after twenty minutes. Clicking the button again before twenty minutes have passed resets the timer.

Standard server logs contain many standard output messages like Info, Warning, or Errors. When debugging starts, the server outputs yet another message prepended with Debug. In most cases, the standard server outputs are enough to identify issues that arise. Debugging is useful only if standard server outputs do not give the cause of the problem.

Click "Stop Debugging" to manually end the debugging session.

In certain cases, attention must be brought to Agiloft regarding problems discovered through debugging. Click Send Debug Logs to send a copy of the logs to Agiloft for inspection.

Bean Shell

The Bean Shell button brings up a separate window to run Java scripts for testing and debugging purposes. From this window, the admin gains full control over the Java Virtual Machine (JVM) that runs the Agiloft system.

BeanShell debugging allows runtime manipulations of parameters and instrumentation. For example, if the admin knows a specific class name, the class may be instantiated, and all of its methods may be invoked from this window.

Debugging and testing in this mode is strongly discouraged unless the admin has significant prior experience with Java/BeanShell and debugging. Misuse may result in a system crash or data corruption.

JavaScript Debugging

This option controls browser debugging. To start, click Start Debugging.

Unlike the Debugging option above, JavaScript debugging produces pop-ups in the browser indicating which JS code is currently running. JavaScript debugging sees its main use in GUI development and testing. Use of this button is rarely if ever needed outside of development.

JavaScript debugging must be manually stopped by re-clicking the Start/Stop Debugging button.

Slow GUI Operation Detection

Agiloft automatically tracks the length of time each GUI operation takes to complete. If a GUI operation takes longer than the specified limit to complete, the system creates an entry in the server logs. This allows admins and developers to determine the frequency and context of slow GUI operations.

To set a new limit, enter the number of seconds in the box and click Set Limit. The limit is applied immediately.

Show Current Activities

This shows activities that are running on the Agiloft system at the moment the Show activities button is clicked. The screen does not show any processes that may have already completed prior to the button click.

For small operations, the table of activities is usually empty at any instant in time. To ensure a non-empty output in small operations for testing purposes, importing multiple records or scheduling a mass edit/delete should suffice.