

BLUEHILL® UNIVERSAL API

Application Programming Interface



Bluehill Universal's Application Programming Interface (API) allows Bluehill system administrators to program their own custom interfaces for various applications and devices to control pre- and post-test actions. In the simplest terms, the API allows you to automate some components of the testing sequence, and create your own interface for running certain parts of the system (exporting, importing, starting, etc.). The Bluehill API is included with the standard purchase of Bluehill Universal, and is accessible with simple Visual Studio .NET based programming.

Importing Data into Bluehill before a Test Starts

In some cases, specimen dimensions or other identifying characteristics are recorded before testing begins. If these values are saved in a spreadsheet or existing database, they can be transferred directly into Bluehill Universal using the API. For example, dimensions for a batch of specimens can be measured and imported beforehand, eliminating dimensional measurement prior to each individual test.

Interface with External Devices or Software Applications

Leverage external tools to perform some tasks, such as opening specific test methods, transferring pre-determined information (as above), or triggering the start of a test. For example, a barcode reader could be used to open the correct method for a particular type of sample.

Exporting Results into an Existing Database (e.g. SQL, LIMS, Access)

Bluehill Universal can create customizable files with relevant information. For example, a portion of a results table, a single numerical result, or a vertical appending *.txt file. The API can be used to interface directly with an existing database, eliminating the need to manually input the test results.

Who can Use the API?

The API can be used by any programmer who is familiar with the following languages: VB, NET, C# or any programming language that's capable of connecting with Windows Connection Foundation (WCF). Additionally, users can interface with Labview via C#.