UNI Ruby

A set of tools for quick and easy development of control modules
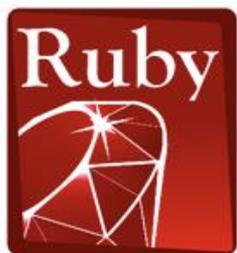
Control modules are pieces of software that create a bridge between the Clarity Chromatography Data System (CDS) and chromatographic hardware such as detectors, thermostats, pumps, etc. Sometimes, you might want to control a device that is rare or even custom-made. You may think that it should not take much effort to do it because what the device does is relatively simple. If you know a bit about programming, such as how to write macros in Excel, and you are willing to learn the Ruby programming language and also something about how Clarity interfaces control modules and chromatography hardware, then you can surely get it done. UNI Ruby provides code examples, mature projects, documentation, and a script editor, all integrated within Clarity CDS to help you develop your control module. Control modules can be used for both personal and public use (see Script distribution below). UNI Ruby's official website can be found at https://www.dataapex.com/product/uniruby.

What is UNI Ruby **NOT** intended for?

If you are thinking about:

- development of complex control modules (such as PDA detectors, MS detectors) or
- control modules with a rich user interface

then you should use **Clarity SDK**. See more at https://www.dataapex.com/product/sdk.

Documentation
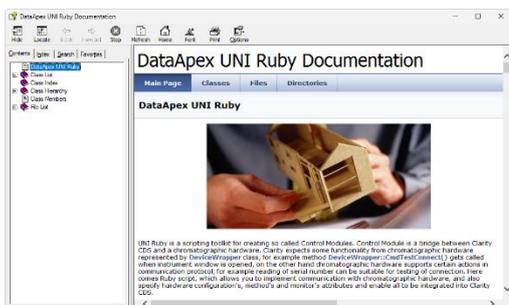
What does it take to write a Ruby control module? This is described on the first page when you double-click the *C:\Clarity\Bin\CswUNIVM.chm* file (assuming the default installation location). There, you will find how to specify items you want to show in the user interface, how to establish communication with your device, etc. You will find a lot of links to examples where a particular feature is implemented. Or you can select an example that is close to your problem from the *C:\Clarity\Bin\Utils\Uni_Drivers\Examples* folder, open it in Editor (see below), and view the context documentation. You can learn a lot from the production Ruby projects as well.

Editor

You can get Notepad++ installed on your computer along with the plug-in integrating the editor with documentation – just check the *Utils - Notepad*++ option during the installation of Clarity. If you open your Ruby file in Notepad++ and place the cursor on a word of interest, you can get help and documentation for it by pressing Ctrl+Shift+F1. Syntax check can be performed by pressing Ctrl+Shit+F7. All other features can be found in the menu *Plugins → UNI Ruby Plugin,* along with shortcuts and plugin configuration.
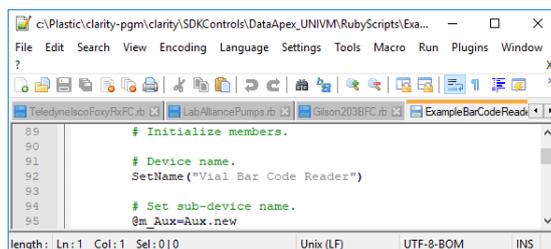
Clarity
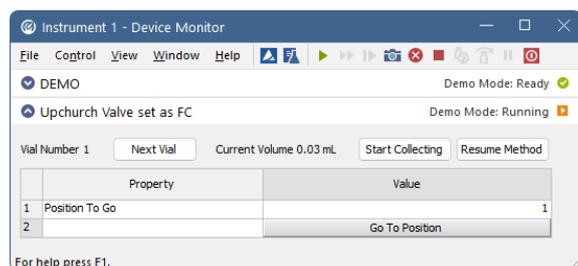
You probably want to see your script in action as soon as possible. For that, run Clarity CDS, go to *System Configuration*, click the *Add...* button, and select the *UNI Ruby* control module in the *Auxiliary* category of the *Available Control Modules* dialog. In the displayed configuration dialog, you can open your Ruby script file and bring it to life. If you leave your editor running and some runtime error occurs, the culprit line will be marked in your script in the editor.

❗ To use the script in Clarity, it is necessary to have the corresponding control module license (e.g., A24, A26, ...) according to the type of device used (device type is determined by UNI Ruby).

Script Distribution

UNI Ruby can be used to write personal (private) scripts and also official (public) scripts.

1. Personal scripts can be used without any approval by DataApex, and you already have all the tools and documents needed. However, in such cases, there will always be a warning in the Instrument window informing you that you are using an unauthorized driver.

2. If you want to distribute the script to multiple users or use it in a regulated environment, making the script official is recommended. An official script does not have the abovementioned limitation, as it has to pass internal testing and be approved by DataApex. To get more information, contact support@dataapex.com to get full access to specific approval documentation:

- D171 External UNI Ruby control module development checklist

- D115 Control module approval procedure

- D126 DataApex CTRL Module Specification

Support

UNI Ruby documentation is provided without any support. DataApex support services are not required for personal scripts. Public scripts require DataApex testing and inclusion into Clarity family products. For consulting/testing, the following services are available:

- p/n S001: SDK/UNI Ruby Support – annual fee, includes consultations, testing & incorporation of drivers/scripts

- p/n S028: UNI Ruby script testing – one-off testing of one UNI-Ruby script (if the script does not comply with the requirements outlined in D171, it will not be approved for public distribution).

Prices are charged according to the valid pricelist.
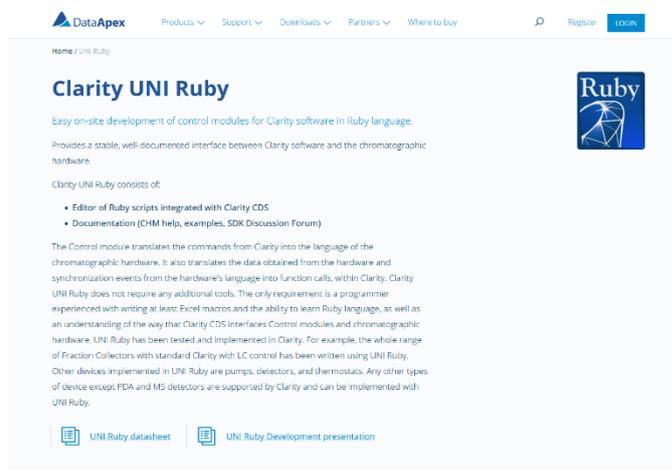
Step-by-Step manual: How to start developing scripts with UNI Ruby

1. Install Clarity CDS and in the custom installation select *Utils – Notepad++ with UNI Ruby plugin*.

2. Double click on *C:\Clarity\Bin\CswUNIVM.chm* and read the first (rather long) page.

3. Select an example or production Ruby script (*.rb) which is similar to your project.

4. Copy the selected script under a new name and open it in Notepad++.

5. Get an overview of what is done in the script by reading function names and comments included with them.

6. Get context help for the selected function by pressing Ctrl-Shift-F1.

7. Perform syntax check by pressing Ctrl-Shift-F7.

8. Run Clarity.

9. Go to *System Configuration*.

10. Add the *UNI Ruby* driver in the *Auxiliary* section.

11. Select your script (by pressing the "..." button).

12. You can see user interface items in the *DataApex UNI Setup* dialog.

13. Close the *DataApex UNI Setup* dialog.

14. Drag and drop the just added driver onto a Clarity Instrument and close *System Configuration*.

15. Open Instrument.

16. Open the *Monitor* window to see your user interface items (if there are any).

17. Open the *Method* dialog to see your user interface items (if there are any).

18. Modify your script to implement communication parameters and communication protocol.

19. Modify your script to implement driver logic and error handling.

Add script to your Clarity Installation:

20. Make sure the driver works in the DEMO mode too.

21. See the approval documents listed above to make sure that your script meets the approval requirements.

22. Contact DataApex to arrange the approval of a new script.