|  |
| --- |
|  |
| Technical Report: PySupDev  RGA System Implementation |
| ITER |

****

Table of Contents

[1 Report on the pyDevSup module 3](#_Toc125984683)

[2 Architecture 3](#_Toc125984684)

[3 Specific Issues 3](#_Toc125984685)

[3.1 Major design issue 3](#_Toc125984686)

[3.2 Python 2 -> 3 issues 3](#_Toc125984687)

[3.3 Support issues 4](#_Toc125984688)

[3.4 Windows build 4](#_Toc125984689)

[Document Information 5](#_Toc125984690)

[Document Control 6](#_Toc125984691)

1. Report on the pyDevSup module

The pDevSup module is a Python module used to manage the interface between Python and EPICS.

1. Architecture

The module consists of these code areas:

1. There are 5 example programs, altogether, that operate the IOC.

These launch the IOC from a command script (as is normal EPICS practice).

1. The IOC code softIocPy and pDevSup.

These are written in C and depend on EPICS base.

The C code makes calls into Python code, invoking the devsup.\_init Python function.

1. The Python module code devsup.

It loads the \_dbapi module and calls in to its C code.

The \_init function creates a temporary file containing the definitions of ‘Python Device’ which are needed by the IOC (above), and pushes the contents in to the \_dbapi module.

1. A shared library \_dbapi(.so or .pyd).

This is written in C and depends on EPICS base.

This dependency is wholly *separate* from the dependency of the IOC code on EPICS base.

The C code here in turn makes calls in to the Python code (3), in response to the ‘py’ command(s) that are present in the script (1).

The inter-operation of 2, 3 and 4 means that the architecture is extraordinarily complex and difficult to debug.

1. Specific Issues
   1. Major design issue

The above architecture means that the EPICS database is apparently loaded twice, once into the IOC code (2) and once into the \_dbapi code (4). If these copies are discrepant from each other, the application won’t work.

This appears to work on Linux, although we are at a loss to understand how.

It does not work on Windows, in that EPICS commands like ‘Python Device’ and ‘py’ that have been programmed in to (4) are unavailable for scripted use by (2).

Use of temporary files for EPICS database definition - as per (3) above – is a breach of ITER coding standards.

* 1. Python 2 -> 3 issues

It is clear that the module has not been thoroughly tested with Python 3.

Several 2->3 compatibility issues have been identified which needed to be corrected, before the code would run.

In particular, the examples were launched with the script command:

#!../../bin/linux-x86/softIocPy2.7

Which is easily changed to:

#!../../bin/linux-x86\_64/softIocPy3.10

This indicates the examples have not been operated since 32-bit Linux and Python 2.7.

N.B. Python 2.7 was released in July 2010, and support ended in Jan 2020.

* 1. Support issues

There’s no distribution of the module through pip or wheel.

The 1.2 tagged library could build on Linux, but the head failed. This can be resolved by reverting a line in the code.

It is also noted that there haven't been updates or pull requests to the GitHub module in the last 2 or 3 years.

These points tend to give the impression that this is not a mainstream or well-supported EPICS IOC implementation library.

* 1. Windows build

The module has not previously been used on Windows, and some build changes are required to make it run on Windows.

We are aware that ITER’s requirement is for use on Linux, and not Windows.

However, ITER may wish to make the RGA software available as open source and available to both Windows and Linux users. For this reason, it would be valuable to ensure that the software successfully builds and runs on both Linux and Windows operating systems.

Document Information

**Correspondence address:** Capgemini, 26 The Quadrant, Abingdon Science Park, Abingdon, Oxfordshire, OX14 3YS, UK

About Capgemini

Capgemini is a global leader in partnering with companies to transform and manage their business by harnessing the power of technology. The Group is guided everyday by its purpose of unleashing human energy through technology for an inclusive and sustainable future. It is a responsible and diverse organization of over 325,000 team members more than 50 countries. With its strong 55-year heritage and deep industry expertise, Capgemini is trusted by its clients to address the entire breadth of their business needs, from strategy and design to operations, fueled by the fast evolving and innovative world of cloud, data, AI, connectivity, software, digital engineering and platforms. The Group reported in 2021 global revenues of €18 billion.

Get the Future You Want | [www.capgemini.com](https://www.capgemini.com/)

Document Control

This document is under document control. All members of the distribution list will receive updated copies whenever alterations are made. All other copies are uncontrolled.

Document Owner

|  |  |  |
| --- | --- | --- |
| Name | Role | Company |
| Peter Heesterman | Owner | Capgemini UK plc |

Document Approval

The following are required to approve this document. Their approval signifies that the document is fit for purpose and approved for use. Evidence of appropriate review will be filed in NPD.

|  |  |  |
| --- | --- | --- |
| Name | Approval Role | Company |
| Kevin Woods | Project Manager | Capgemini UK plc |

Distribution

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Role | Company | | |
| Uron Kruzei | Group Leader, ITER | | ITER |
| Notes Projects Database 11868 | Filing |  | | |

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Author | Issue | Date | Description |
| Peter Heesterman | V1.R1.M0 | 30-Jan-2023 | 1st version written |
|  |  |  |  |

References

|  |  |  |  |
| --- | --- | --- | --- |
| Document | Ref | Date | Details & Issue |
|  |  |  |  |
|  |  |  |  |

