ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA113

EPICS MAINTENANCE TOOLS AND PRACTICES AT FRIB'S DIAGNOSTICS DEPARTMENT*

D. O. Omitto†, S. Cogan, B. S. Martins
Facility for Rare Isotope Beams, East Lansing, USA

Abstract

The Beam Instrumentation and Measurements department is responsible for dozens of different diagnostics devices deployed at multiple locations at the Facility for Rare Isotope Beam. To manage such a high number of devices, different tools were created to address preventive and corrective maintenance tasks and check the overall health of the equipment. This work will present how the EPICS tools and frameworks, such as archiver, channel finder, and pyDevSup, were integrated with our environment to help achieve high availability for the beam diagnostic devices

INTRODUCTION

The FRIB's diagnostics group under the Beam Instrumentation and Measurements department is responsible for a variety of instruments and devices used to measure beam parameters that are essential for the machine's overall tuning and operation. While other groups at FRIB typically have a large number of similar devices, diagnostics devices are heterogeneous systems that have specific needs maintenance-wise. The team manages 15 device types with about 350 total device instances. In the controls layer, there are 36 Experimental Physics and Industrial Control System (EPICS) [1] Input/Output Controllers (IOC) types with about 250 instances. From a maintenance perspective, the diagnostics devices can be classified into intercepting and non-intercepting concerning the ion beam, each with their own set of preventive and corrective actions/requirements and tools.

INTERCEPTING DEVICES

Intercepting devices are usually large systems that have to be moved into the path of the beam and then either partially or fully block the beam. Sometimes they also need to be moved into a shared space in the beam pipe and require some extra insertion coordination via interlocks, which is handled by Allen Bradley PLCs to avoid potential collisions. Examples for such devices are Allison Scanners, Profile Monitors and Faraday Cups. They are usually critical systems responsible for determining the machine mode and helping mitigate beam power loss and trajectory deviation. Because they are physically moving devices, they require some extra attention regarding mechanical wear and tear.

* This material is based upon work supported by the U.S. Department of Energy Office of Science under Cooperative Agreement DE-SC0000661, the State of Michigan and Michigan State University.
† omitto@frib.msu.edu

NON-INTERCEPTING DEVICES

Non-intercepting devices are non-invasive systems that are installed in a fixed position alongside the beam path with electronics that help monitor beam energy, beam position stability, and beam power losses. They are usually not subject to mechanical wear and tear but require active monitoring and calibration of the underlying electronics responsible for their data acquisition. Examples of such devices are Beam Position Monitors, Beam Current Monitors, Neutron Detectors and Halo Monitor Rings. Figure 1 shows examples of both intercepting and non-intercepting devices that share space in the beam pipe.

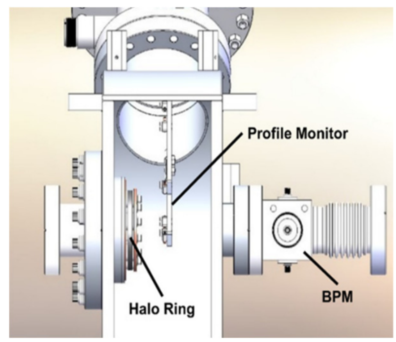


Figure 1: Intercepting and non-intercepting devices.

HARDWARE AND SOFTWARE STANDARDIZATION

Standards are essential to achieve an integrated, maintainable, and affordable control system [2]. To help minimize installation efforts and beam downtime during the initial commissioning of the machine, the diagnostics group has adopted a few devices and technologies as standard, most of which are widely used by other laboratories. For instance, we adopted the Delta Tau Geobrick LV as the standard stepper motor controller, which is also the standard controller at the Diamond Light Source and the National Synchrotron Light Source II. Also, the MicroTCA system, developed by Desy, was chosen as the standard for fast-acquisition electronics.

The EPICS framework and some of its tools, such as iocStats [3] for IOC health monitoring, the Archiver Appliance [4] for PV data archival and Channel Finder [5] as a PV directory server, are used as the central distributed control system. Jenkins [6] is used for automatic building of Debian packages [7], Stash [8] is used for software version control, and Puppet [9] is responsible for automatic deployment of Debian packages and EPICS IOCs.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA113

CORRECTIVE ACTIONS AND TOOLS

The EPICS iocStats module is the core of most of the corrective software maintenance tasks such as restarting a specific IOC, providing heartbeats for system status and reporting machine resource utilization. To facilitate and combine some of these activities, an IOC called healthIOC was developed to act as a supervisor for all the diagnostics IOCs.

HealthIOC finds all existing diagnostics IOCs by searching Channel Finder (via its web API) for IOCs that have the owner tag set to "diag". Then, it spawns one monitor for each IOC found. A diagram of how healthIOC works are shown in Fig. 2.

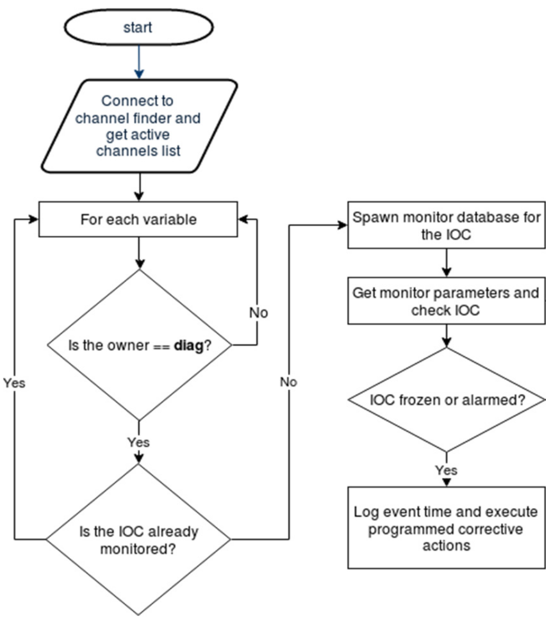


Figure 2: HealthIOC operation diagram.

The diagnostics electronic boards have to be flexible enough to support multiple operating modes with varying time structure along with different peak intensities. That means having to monitor and set a myriad of configuration parameters, such as electronics gain, acquisition mode, and time of flight settings, before every operation run. To minimize response time and ease the burden of checking the configuration state in each subsystem, the restoreIOC was developed to automate this task. A diagram of its operation is shown in Fig. 4.

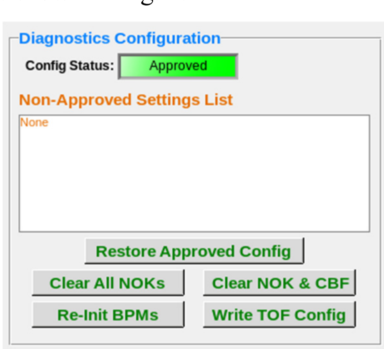


Figure 3: RestoreIOC screen.

RestoreIOC reads the desired configuration state from a database file and uses pyDevSup [10] with callbacks comparing the desired setpoints with the actual value. If divergences are found, they are presented to the user, as shown in Fig. 3. Also, specific groups of setpoints can be individually restored on command.

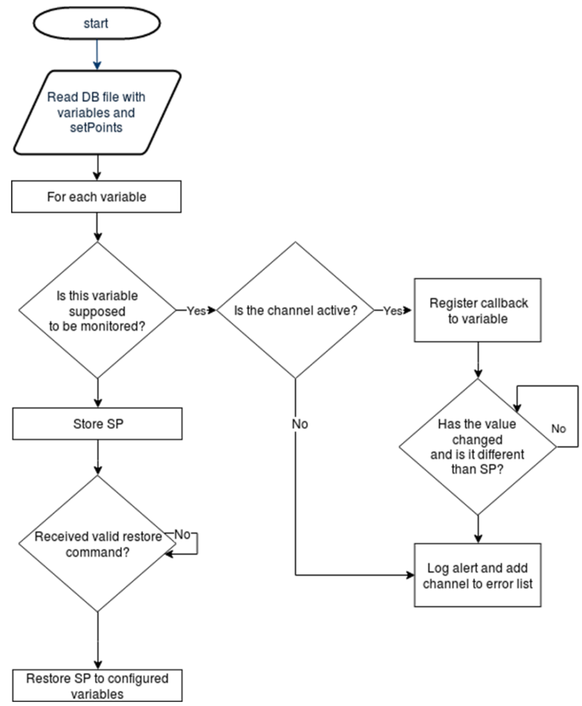


Figure 4: RestoreIOC operation diagram.

PREVENTIVE ACTIONS AND TOOLS

While almost all maintenance activities require some machine downtime, a good preventive maintenance program can decrease and optimize that downtime. With that in mind, the diagnostics group has developed a set of tools to avoid solely relying on reactive maintenance.

For the non-intercepting devices, automatic electronics calibrations are run overnight or on machine idle time with calibrated test signals, and the results are archived for posterior analysis and action.

The intercepting devices are monitored during normal operations where all the mechanical missteps and positional drifts are automatically monitored, archived, and alarmed over set thresholds. To help with this objective, a maintenance IOC was developed to help with common mechanical maintenance tasks such as cleaning and lubrication of moving parts. This IOC uses the Archiver Appliance to account for cycle counts and total traveled distance since the last maintenance task was performed and since the device was installed. This usage information is vital to help estimate the overall health and wear of particular devices and if deemed necessary, trigger preventive maintenance activities on them. A diagram showing the operation of the maintenanceIOC is shown in Fig. 5, while a screenshot of its operator interface is shown in Fig. 6.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA113

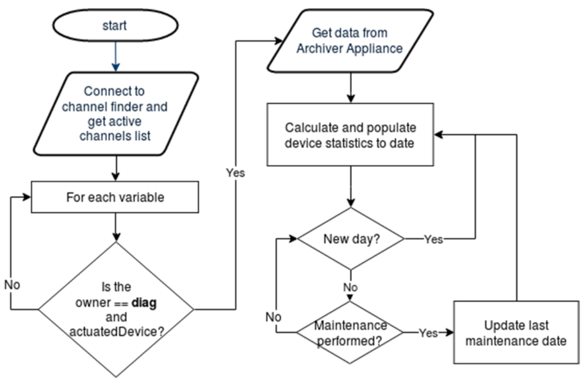
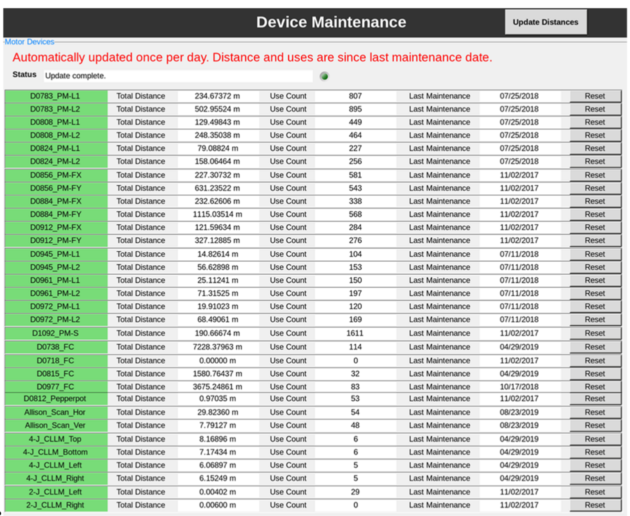


Figure 5: MaintenanceIOC diagram.



D0783_PM-L1	Total Distance	234.67372 m	Use Count	807	Last Maintenance	07/25/2018	Reset
D0783_PM-L2	Total Distance	502.95524 m	Use Count	895	Last Maintenance	07/25/2018	Reset
D0808_PM-L1	Total Distance	129.49843 m	Use Count	449	Last Maintenance	07/25/2018	Reset
D0808_PM-L2	Total Distance	248.35038 m	Use Count	464	Last Maintenance	07/25/2018	Reset
D0824_PM-L1	Total Distance	79.08824 m	Use Count	227	Last Maintenance	07/25/2018	Reset
D0824_PM-L2	Total Distance	158.06464 m	Use Count	256	Last Maintenance	07/25/2018	Reset
D0856_PM-FX	Total Distance	227.30732 m	Use Count	581	Last Maintenance	11/02/2017	Reset
D0856_PM-FY	Total Distance	631.23522 m	Use Count	543	Last Maintenance	11/02/2017	Reset
D0884_PM-FX	Total Distance	232.62606 m	Use Count	338	Last Maintenance	11/02/2017	Reset
D0884_PM-FY	Total Distance	1115.03514 m	Use Count	568	Last Maintenance	11/02/2017	Reset
D0912_PM-FX	Total Distance	121.59634 m	Use Count	284	Last Maintenance	11/02/2017	Reset
D0912_PM-FY	Total Distance	327.12885 m	Use Count	276	Last Maintenance	11/02/2017	Reset
D0945_PM-L1	Total Distance	14.82614 m	Use Count	104	Last Maintenance	07/11/2018	Reset
D0945_PM-L2	Total Distance	56.62898 m	Use Count	153	Last Maintenance	07/11/2018	Reset
D0961_PM-L1	Total Distance	25.11241 m	Use Count	150	Last Maintenance	07/11/2018	Reset
D0961_PM-L2	Total Distance	71.31525 m	Use Count	197	Last Maintenance	07/11/2018	Reset
D0972_PM-L1	Total Distance	19.91023 m	Use Count	120	Last Maintenance	07/11/2018	Reset
D0972_PM-L2	Total Distance	68.49061 m	Use Count	169	Last Maintenance	07/11/2018	Reset
D1092_PM-S	Total Distance	190.66674 m	Use Count	1611	Last Maintenance	11/02/2017	Reset
D0738_FC	Total Distance	7228.37963 m	Use Count	114	Last Maintenance	04/29/2019	Reset
D0718_FC	Total Distance	0.00000 m	Use Count	0	Last Maintenance	11/02/2017	Reset
D0815_FC	Total Distance	1580.76437 m	Use Count	32	Last Maintenance	04/29/2019	Reset
D0977_FC	Total Distance	3675.24861 m	Use Count	83	Last Maintenance	10/17/2018	Reset
D0812_Pepperpot	Total Distance	0.97035 m	Use Count	53	Last Maintenance	11/02/2017	Reset
Allison_Scan_Hor	Total Distance	29.82360 m	Use Count	54	Last Maintenance	08/23/2019	Reset
Allison_Scan_Ver	Total Distance	7.79127 m	Use Count	48	Last Maintenance	08/23/2019	Reset
4-J_CLLM_Top	Total Distance	8.16896 m	Use Count	6	Last Maintenance	04/29/2019	Reset
4-J_CLLM_Bottom	Total Distance	7.17434 m	Use Count	6	Last Maintenance	04/29/2019	Reset
4-J_CLLM_Left	Total Distance	6.06897 m	Use Count	5	Last Maintenance	04/29/2019	Reset
4-J_CLLM_Right	Total Distance	6.15249 m	Use Count	5	Last Maintenance	04/29/2019	Reset
2-J_CLLM_Left	Total Distance	0.00402 m	Use Count	29	Last Maintenance	11/02/2017	Reset
2-J_CLLM_Right	Total Distance	0.00600 m	Use Count	0	Last Maintenance	11/02/2017	Reset

Figure 6: Maintenance screen.

FUTURE IMPROVEMENTS

The diagnostics maintenance tools could benefit from better integration with the engineering alarm server, using the later to handle e-mail alerts and thresholds configuration. At this moment there's only one alarm server being used by the operations group that serves a different purpose; an engineering alarm server is scheduled to be deployed in the near term.

Moreover, because there's an air gap between office and controls network, there's also some benefit in implementing e-mails with reminders and summaries instead of only alarmed conditions.

CONCLUSION

The diagnostics maintenance tools presented in this paper have proved to be very useful in increasing the overall performance and availability of our devices. Keeping track of maintenance can be time-consuming, and there are lots of room for human error. The tools not only save experts significant time but also guarantees consistency across all the different subsystems within our group.

REFERENCES

[1] EPICS, `http://www.epics-controls.org`

[2] J.Y. Journeaux, D. Joonekindt, W.D. Klotz, P. Mäkijärvi, A. Wallander, I. Yonekawa, "Instrumentation and control standardization in the ITER project", in *Fusion Engineering and Design*, 2011, vol. 86, no. 6–8, pp. 1157-1162. `doi:10.1016/j.fusengdes.2010.12.014`.

[3] iocStats EPICS IOC Status and Control, `https://github.com/epics-modules/iocStats`

[4] The EPICS Archiver Appliance, `https://slacmshankar.github.io/epicsarchiver_docs`

[5] ChannelFinder, `http://channelfinder.github.io`

[6] Jenkins, `https://jenkins.io`

[7] M. Konrad, D. Maxwell and G. Shen, "Continuous Integration and Continuous Delivery at FRIB", in *Proc. 11th Int. Workshop on Personal Computers and Particle Accelerator Controls (PCaPAC'16)*, Campinas, Brazil, Oct. 2016, pp.145-147, `doi:10.18429/JACoW-PCAPAC2016-FRITPLCO01`

[8] Atlassian Stash, `https://www.atlassian.com/software/bitbucket`

[9] Puppet, `https://puppet.com`

[10] pyDevSup, `https://github.com/mdavidsaver/pyDevSup`