

DATABASE FOR THE MANAGEMENT OF NSLS-II ACTIVE INTERLOCK SYSTEM*

J. Choi[†], R. Fliller, K. Ha, Y. Tian, BNL, Upton, NY 11973, USA

Abstract

NSLS-II is operating the active interlock (AI) system to protect the machine components from the synchrotron radiation from the accidentally mis-steered electron beam. For the systematic management, a relational database is dedicated to the AI system and working as the data provider as well as the archiver. The paper shows how the database is structured and used for the AI system.

INTRODUCTION

NSLS-II is a 3rd generation light source commissioned in 2014 [1]. It has a double bend achromat lattice with 30 cells on a 792 m circumference. The 30 straight sections are alternatively long and short with high and low horizontal beta values, and the ring consists of the 15 super-period with 60 dipole magnets and the electron energy is 3 GeV.

To protect the NSLS-II storage ring components from the damage by the synchrotron radiation generated at insertion devices (IDs) and bending magnets, dedicated active interlock (AI) system is implemented to keep the electron beam in the storage ring within the safe envelope called active interlock envelope (AIE) in the transverse phase space.

A lot of data, mainly from the beam position monitors (BPMs), and corresponding logics are involved to evaluate the situation from the AIE point of view and decide whether the situation is in danger enough to dump the beam. This process also needs to be very fast (< 1 ms for ID radiation). Because dumping the beam cannot be decided by any chance from the wrong data taken by mistake or miscalculation, we systematically maintain the information and logics implementing the AI system using the relational database which was developed by the NSLS-II control group [2].

The database has web interface for the management and is also connected to the Control System Studio (CSS) AI pages to download the data and apply them to the implementation.

AI LOGICS

There can be various scenarios that the synchrotron radiation can damage the SR components, especially frontend and beamline components. To prevent such scenarios as well as to minimize unnecessary beam trips, AI system has been developed for many years and carefully reviewed [3–6]. In fact, the system is still under development in the sense that it is updated whenever needed by studying the every case when it trips the beam [7–9].

Because the frontend and beamline components can be damaged by the synchrotron radiation from the dipole mag-

nets as well as from the IDs, there are two independent logics are being under operation to evaluate the deviations in the sector BPMs and ID BPMs.

Logic for the Sector BPMs

The logic is prepared to protect the components from the synchrotron radiation from the bending magnets (BMSR). It detects the global deviation of the electron beam around ring and, if the conditions are satisfied, the beam is tripped. The parameters used to evaluate the operation status and their current threshold values are:

- Storage ring beam current (50 mA)
- Horizontal offset beam position (± 5 mm)
- Vertical offset beam position (± 3 mm)
- Number of BPMs deviated (13)

That is, once the stored beam current is higher than or equal to 50 mA, either case of more than 13 horizontal BPM readings are out of range ± 5 mm or more than 13 vertical BPM readings are out of range ± 3 mm triggers the beam-dump.

Logic for the ID BPMs

The damage by the ID radiation (IDSR) can be more serious than that by BMSR. On the other hand, because of the number of IDs, making the logic more conservative can result in intolerable increase of non-necessary beam-dumps.

For more efficient and clear implementation, the AI system defines the concept *AI Engaged*. The system should be engaged first in order that the deviation of the beam path or other conditions can trigger the beam-dump.

In the ID logic, in addition to the beam current and the BPM readings, the statuses of photon shutters should be also monitored and involved in the system. The first upstream component of the frontend is the bending magnet photon shutter (BMPS) which intercepts the bending magnet radiation when there is no frontend or when the frontend is under maintenance. Another photon shutter (IDPS) is located before the safety shutter and both BMPS and IDPS will be closed depending on the needs of equipment protection system (EPS).

The condition for a beamline to be engaged to the AI system is that [**IDPS open and beam current > 0.2 mA**] or [**beam current > 2 mA and ID is closed**]. In fact, the condition **beam current > 0.2 mA** should be **beam current > 0 mA**. That is, the AI system is engaged when the IDPS open and beam current is not 0, because the path through the frontend is free and any radiation can directly reach to the beamline. The threshold current is 0.2 mA because of the reliability of the DCCT reading. In the second condition,

* Work supported by U.S. Department of Energy (DOE) under contract No. DE-SC0012704. DE-SC0012704

[†] jchoi@bnl.gov

each ID has definite threshold gap value for the AI system between *ID Open* and *ID Closed*.

Once a beamline is engaged, one of the following three conditions will dump the stored beam.

- BMPS closed
- The position of the beam in the transverse phase space is out of AIE.
- Power supply current of the canting magnet is out of the safe region.

The BMPS of the frontend is designed to intercept the BMSR. If the BMPS is hit by the IDSR, it might be damaged. As the result of the considerable discussions and reviews, the AIE for all ID sections is decided as ± 0.5 mm and ± 0.25 mrad in both the horizontal and vertical plane at the center of the IDs [5, 6].

The last condition applies only to the canted ID section. Each power supply for the canting magnet has pre-set safe current range and the stored beam is dropped if the readback current value of the canting magnet power supply is out of the range.

DATABASE

As can be seen in the previous section, many data are involved in the decision-making of the AI system. While some of them are used directly, many of the data are needed to be processed to get the value to be used. For example, the position of the beam in phase space at the ID center is calculated from the ID BPM readings, BPM offset from the beam based alignment (BBA) and the BPM locations in the ring. Because any error in the AI related data can cause unnecessary beam-dump and reduction of operation time, a relational database management system, MySQL [10], is adopted to maintain and update the data. The data is managed by the web interfaces and the AI system is communicating with the database through CSS pages and eventually the scripts in the EPICS IOC AI system server.

Database Structure

Not including auxiliary tables, the database of active interlock system consists of 5 tables and the database schema is shown in Fig. 1.

As shown in Fig. 1, the central tables of AI database are as follows.

```
active_interlock
active_interlock_device
active_interlock_logic
active_interlock_prop
active_interlock_prop_type
```

The `active_interlock` table contains the general information of each version with the version number and the status of the version: Editable, Approved, Active, Backup,

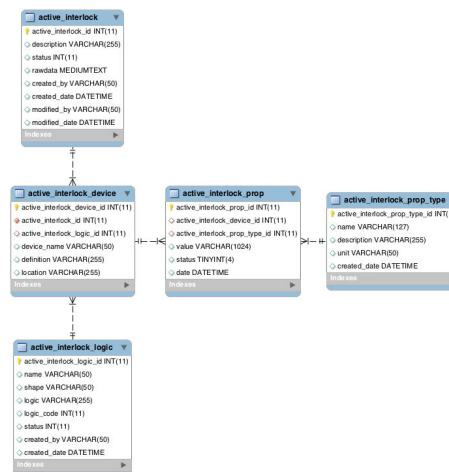


Figure 1: Schema of the AI system database.

and History, which will be explained in the next subsection. Every version is associated with all involved devices through `active_interlock_device` table. Therefore, in the device table, almost all the devices are repeated for every version. And, the parameter values for the given device are also associated with the corresponding version through `active_interlock_prop` and `active_interlock_prop_type` tables.

The types of the parameters are defined in the `active_interlock_prop_type` table. The parameter values associated with the devices, versions, and the types are written in the `active_interlock_prop` table and are used together with the AI logic to decide the current operation status is safe or not. These values can be managed through the web interface pages.

The logics are saved in the `active_interlock_logic` table. At this moment, here are two entries, one for the BMSR logic and one for the IDSR logic. The entries are expressed in mathematical formulae using the device parameter values as symbols.

Development Tools and Interfaces

The high-level programming language, Python [11] is used for the communication between the AI system and the database, and the main tool used to develop the database system is Python-Django [12]. However, for accessing the database, MySQLdb [13] is used instead of model scheme of the Python-Django. The web interface pages are showing all BPM related data being used in the AI system. One page is dedicated to the BMSR and one to the IDSR. It can also show the logics in the programming how the BPM information is used. Figure 2 shows the web interface page for the BMSR data management.

As for the BPM information, there are 4 categories in versions: *Editable*, *Approved*, *Active*, *Backup* and *History*. As can be expected, the parameters of the *Active* version are controlling the AI system running now. If it should be updated, an *Editable* version should be created first. The *Ed-*

Cell	Element Name	Element Type	s	BPM-SR	AI Offset Limit Hor. (AIOLH)	AI Offset Origin Hor. (AIORH)	BPM-SR	AI Offset Limit Ver. (AIOLV)	AI Offset Origin Ver. (AIORV)	Safe current	In use
Sequence					mm	mm		mm	mm	mA	
30 1	PH1G2C30A	BPM	4.935	AIE-BM	5.0	0.0	AIE-BM	3.0	0.0	50	Yes
30 2	PH2G2C30A	BPM	7.46	AIE-BM	5.0	0.0	AIE-BM	3.0	0.0	50	Yes
30 3	PM1G4C30A	BPM	13.145	AIE-BM	5.0	0.0	AIE-BM	3.0	0.0	50	Yes

Figure 2: BMSR data web interface page for the AI system.

itable version can be made not only from the *Active* version but we can use any pre-existing version.

Once the *Editable* version is prepared, it can be modified and new devices (e.g. BPMs for new IDs) can be added with the appropriate information. After reviewing the update, we can approve it so that *Editable* version is becoming the *Approved* version. Now the new version is ready, and the new parameters can be downloaded to the AI system through the CSS page. Only *Active* or *Approved* version can be downloaded and if the *Approved* version is downloaded, the new version becomes the *Active*, and the old *Active* becomes the *Backup* and the old *Backup* version is added to the *History* versions.

CONCLUSION

We briefly reviewed the active interlock status and logics in use at the NSLS-II storage ring. Because the reliability of the AI system is very critical to the machine operation, we use the fully fledged relational database system to maintain and update the information needed in the AI system. The database is also maintaining the logics of the AI system.

By using the well-defined web interfaces to review and update the AI system data and logics, we can prevent any chance for the wrong data to be provided to the AI system by mistake. Also, when updating is needed as in the case like adding the new IDs or adjusting the trip conditions from the experiences, the database and web interfaces are providing the clear and efficient tools. Furthermore, by maintaining all the history of the AI relevant devices and their parameters, we can review and compare any version which once used for the AI system.

The database structure is also organized that it can easily accommodate the new situation when even the basic AI logics are changed though the experiences.

REFERENCES

- [1] F. Willeke, "Commissioning of NSLS-II," in *Proc. 6th Int. Particle Accelerator Conf. (IPAC'15)*, Richmond, VA, USA, May 2015, paper MOYGB3, pp. 11–16.
- [2] G. Shen, "Development of the database for the active interlock system," unpublished.
- [3] P. Ilinski, "NSLS-II active interlock settings," unpublished.
- [4] S. Seletskiy, C. Amundsen, K. Ha, A. Hussain, "Safe operating conditions for NSLS-II storage ring frontends commissioning," BNL-107813-2015-IR, April 2015.
- [5] S. Seletskiy, T. Shaftan, "Reviewed approach to defining the Active Interlock Envelope for Front End ray tracing," BNL-108464-2015-IR, Sep. 2015.
- [6] S. Seletskiy, J. Choi, K. Ha, R. Smith, Proceedings of IPAC2016, "Comprehensive approach to synchrotron radiation protection of NSLS-II," in *Proc. 7th Int. Particle Accelerator Conf. (IPAC'16)*, Busan, Korea, May 2016, paper THPOY047, pp. 4211–4213.
- [7] S. Seletskiy *et al.*, "Commissioning of the synchrotron radiation protection system and beamlines frontends at NSLS-II," *AIP Conf. Proc.* 1741, pp. 020042-1–020042-4 (2016).
- [8] S. Seletskiy *et al.*, "Commissioning of active interlock system for NSLS-II storage ring," in *Proc. 6th Int. Particle Accelerator Conf. (IPAC'15)*, Richmond, VA, USA, May 2015, paper TUPMA057, pp. 1962–1964.
- [9] S. Seletskiy, J. Choi, K. Ha, R. Smith, "Upgrade of NSLS-II active interlock system," in *Proc. 7th Int. Particle Accelerator Conf. (IPAC'16)*, Busan, Korea, May 2016, paper THPOY023, pp. 4140–4102.
- [10] MySQL, <https://www.mysql.com>
- [11] Python, <https://www.python.org>
- [12] Python-Django, <https://www.djangoproject.com>
- [13] MySQLdb, <http://mysqlclient.readthedocs.io>