

SuperKEKB の CSS アラームシステム運用状況 OPERATION STATUS OF CSS ALARM SYSTEM FOR SUPERKEKB

中村卓也 *^{A)}、岩崎昌子 †^{B)}、帯名崇 ^{B)}、佐々木信哉 ^{B)}、浅野和哉 ^{C)}
Takuya Nakamura*^{A)}, Masako Iwasaki†^{B)}, Takashi Obina^{B)}, Shinya Sasaki^{B)}, Kazuya Asano^{C)}

^{A)}Mitsubishi Electric System and Service Co.,Ltd.,

^{B)}High Energy Accelerator Research Organization (KEK),

^{C)}Kanto Information Service Co., Ltd.,

Abstract

We have developed a new alarm system for the SuperKEKB accelerator operation based on CSS. For the stable operation under the 40 times higher luminosity environment, we have implemented the CSS alarm system, which is used in several accelerators. We operated the new CSS based alarm system under the SuperKEKB operation (Phase-1) in February to June 2016. For the SuperKEKB operations, where the operation situations changed day by day, we have improved the CSS alarm system, and have developed several alarm client tools. We have fixed the problems which occurred in the real operations. In this paper, we report the operation status of the CSS alarm system for SuperKEKB.

1. はじめに

2016年2月より、SuperKEKB 加速器の運転が開始された。SuperKEKB では、KEKB での電子・陽電子衝突頻度を約40倍高めることを目指しており、このような高輝度加速器の運転下において、安定して動作可能なアラームシステムの構築が重要である。我々は、SuperKEKB の運転開始前より Control System Studio (CSS) [1][2] を利用したアラームシステムの動作試験、およびその性能評価を行い、その後 SuperKEKB で運用するよう整備を行った [3]。

実際に SuperKEKB の運転が開始されてからも、日々変化する加速器の運転状況に応じて、アラームの監視設定の変更や、クライアントツールの変更を行ってきた。また、実際の運用を開始してから判明した問題点などもあり、対策を検討する必要もあった。こうした変更要求への対応や、問題点への対策を順次行っていく、加速器の運転状況に則したアラームシステムを運用するよう整備を進めている。本件では、SuperKEKB での CSS アラームシステムの運用状況について報告する。

2. CSS アラームシステムの構成

2.1 CSS アラームシステムのソフトウェア

CSS のアラームシステムは、複数のソフトウェアの組み合わせによって構成されている (Figure 1)。CSS には Alarm Server(アラーム監視サーバー)、Alarm Client(アラーム表示クライアント)、Alarm Config Tool(設定変更ツール)、JMS2RDB(アラーム履歴保存)といった、アラームシステムの根幹をなすソフトウェアが用意されている。アラームシステムの各ソフトウェアは、JMS(Java Message Service) を通じて互いにメッセージの通信を行っており、その JMS のソフトウェアとして Apache ActiveMQ [4] を使用している。また、アラームの設定や履歴情報の保存には RDBMS(Relational Database Management System) を利用しており、我々の環境では PostgreSQL [5] を採用している。

* nakataka@post.kek.jp

† Present affiliation is Osaka City University.

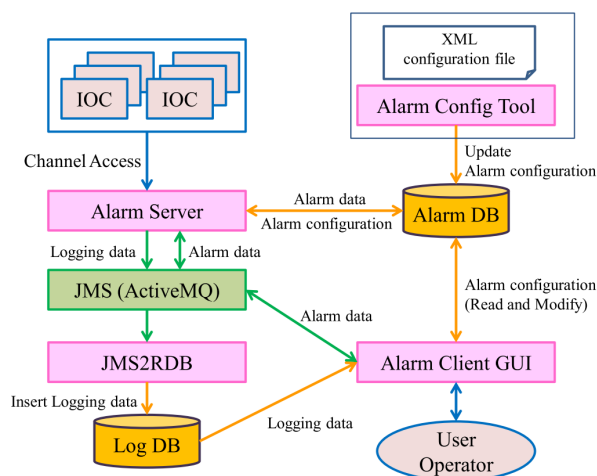


Figure 1: CSS alarm system structure.

3. 独自のクライアントツール

3.1 Python のクライアントツール

PF-AR 加速器では 2011 年より CSS アラームシステムを導入しているが、CSS 標準のクライアントツールだけではなく、独自のクライアントツールを Python [6] で作成して利用している [7]。例えば、アラーム履歴情報について自動的に表示を更新する機能など、CSS 標準のツールには備わっていない機能を補うよう、独自の Python のプログラムを整備した。

SuperKEKB のアラームシステムについても、PF-AR で使用している Python のプログラムをベースに、SuperKEKB の運用環境で使用できるよう整備を行った。特に、今回の SuperKEKB のアラームシステムでは、PF-AR のようなハードウェアグループごとの大きな分類だけではなく、各グループの中にもサブグループを作成して細かく分類できるよう整備している。そうしたサブグループを用いた分類にも対応できるよう、Python プログラムの修正を行った (Figure 2, 3)。

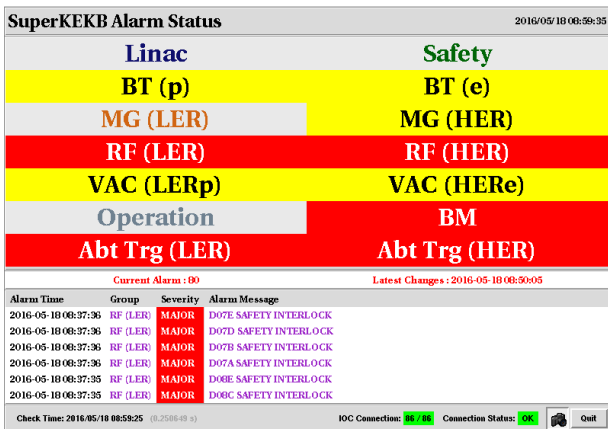


Figure 2: Python alarm status panel.

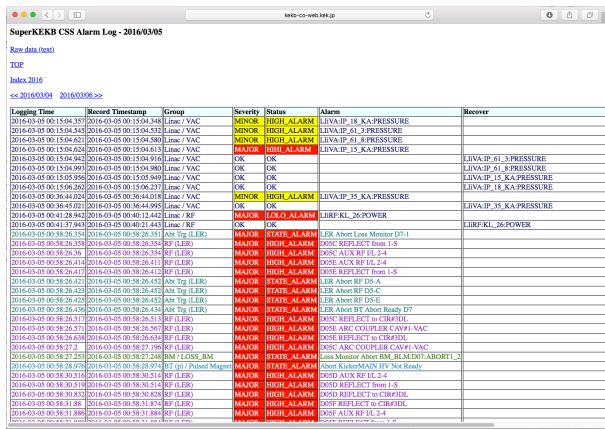


Figure 4: Alarm message history web viewer.

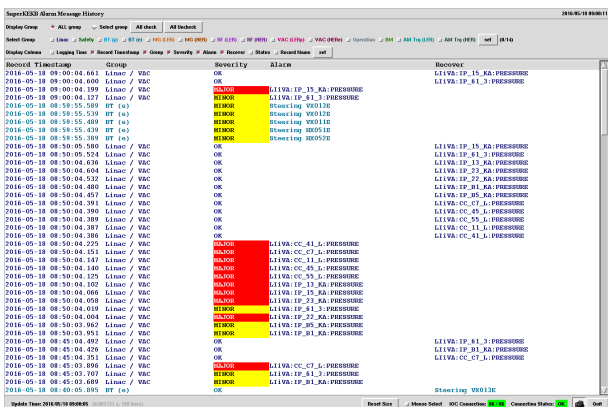


Figure 3: Python alarm message history panel.

3.2 アラーム履歴の Web ページ

アラーム履歴の情報について、Web ブラウザから閲覧できる環境を整備した (Figure 4)。毎分アラーム履歴のデータベースから情報を抽出し、HTML ファイルへと書き出すプログラムを作成した。CSS の標準のクライアントツールや、Python のクライアントツールを使用しなくても、一般的な Web ブラウザを使用してアラームの履歴情報を閲覧することができる。1 分に 1 度の更新のためリアルタイムな情報ではないが、Web ブラウザから手軽にアラームの履歴を閲覧して運転状況を確認することができる。また、SuperKEKB の運転ログは、KEKB に引き続いて Web アプリケーションの電子運転ログ [8] を使用している。運転中に発生したアラームの情報を運転ログに記録する際に、Web ブラウザから簡単に情報をコピーすることができ、運転員が手早く効率的にログの入力を行うことができる。

3.3 アラーム設定変更ツール

CSS 標準のアラーム設定変更ツールでは、登録内容の追加や更新、及び全ての登録内容を一旦削除して追加する再登録の機能は用意されているが、登録内容の一部削除を行う機能は用意されていない。また、一件ずつ手動で変更するクライアントツールも用意されているが、多数の登録を変更する場合には膨大な手間が必要となり、現実的な運用には向いていない。こうし

たアラームの監視設定の変更について、自由度の高い操作を行うことができる独自の Java のプログラムを作成した (Figure 5)。

この Java のプログラムでは、CSV 形式のファイルを利用してアラーム監視設定の変更を行う。CSV 形式のファイルに登録内容の変更点を記述し、そのファイルを読み込んで追加、更新、削除といった登録内容の変更操作を行う。

また、CSS 標準のアラーム設定変更ツールでは、アラームの登録情報を XML 形式で記述したファイルを用意する必要がある。今回の Java プログラムでは一般的に広く用いられている CSV 形式のファイルを利用するため、XML 形式のファイルに比べて閲覧や編集の作業を行いやすい。そのため、各ハードウェアグループ側でもアラーム監視設定ファイルの作成を行うことが容易となり、アラーム設定変更に関する作業を分担することが可能となった。

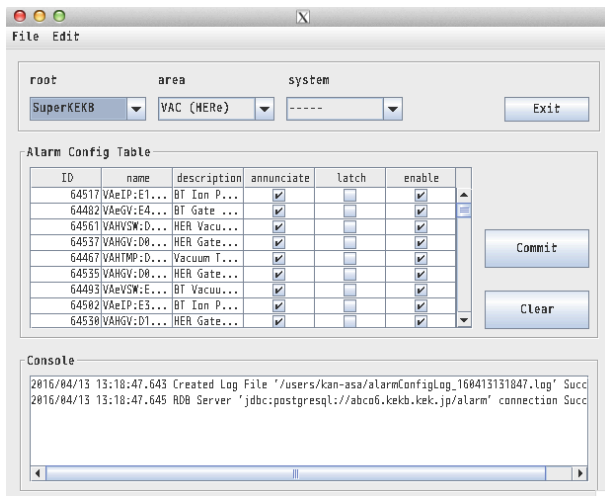


Figure 5: Java alarm configuration tool.

4. SUPERKEKB でのアラームシステム運用状況

4.1 計算機環境

SuperKEKB のアラームシステムを構築するにあたり、アラームシステム専用の 1U のサーバー計算機を整備して運用している (Table 1)。PF-AR のアラームシステムに比べ、CPU やメモリなどの計算機の性能が向上している。また、計算機のディスク領域として SSD を使用しているため、HDD に比べてデータベースの読み書きを高速に行うことができる。

Table 1: Specification of Linux Computers

	SuperKEKB	PF-AR
Type	1U server	Blade server
CPU	Intel Xeon E5-2603 1.6GHz	Intel Xeon 5160 3.00 GHz
# of cores	12 (6 x 2)	2
Memory	32GB	2GB
OS	CentOS 7.1	CentOS 5.8
Disk Type	SSD	HDD
CSS	3.1.2, 3.2.16	3.0.0
Java	1.7.0_79	1.6.0_26
ActiveMQ	5.11.1	5.5.0
PostgreSQL	9.4.4	9.1.3

4.2 CSS バージョン

SuperKEKB のアラームシステムでは、CSS 3.1.2 と CSS 3.2.16 の 2 つのバージョンでの並行運用を行った。運転開始前の負荷試験の中で、CSS 3.2.16 のクライアントツールの時刻表示に関するバグの判明や、CSS 3.1.2 に比べて CPU やメモリの負荷が高いといった点があったため、CSS 3.2.16 を運転用のアラームシステムとして採用するには不安があった。そのため、CSS 3.1.2 を通常の運転で使用し、CSS 3.2.16 を試験的に動作させる運用方針とした。CSS 3.2.16 は試験的な運用とはなったが、今回整備した計算機の性能が十分に高いため CPU やメモリの負荷が問題になることは無く、安定して運用できることを確認した。

4.3 アラーム判定用ソフトウェア IOC

SuperKEKB で運用するアラームシステムの基本的な方針として、各ハードウェアグループ側の IOC でアラームの設定を管理するようにしている。ただし、アラームシステムや加速器運転の都合から、いくつかのアラーム判定用のソフトウェア IOC を別途用意して運用している。例えば、ハードウェア自体の正常や異常の判定ではなく、加速器の運転に応じたアラームの判定を行いたい場合や、第 5 章に記載するアラームシステムの問題点への対処など、必要に応じてアラーム判定用のソフトウェア IOC を整備している。

4.4 アラーム監視設定の更新作業

SuperKEKB の運転が開始されつつある 2 月上旬に、各グループの初期段階のアラーム監視設定が登録された。その後 SuperKEKB の運転が開始され、日々運転が進められていく中で、アラームとして監視すべき対象の追加や変更の要望が挙がり、随時アラーム監視設定の変更が行われた (Table 2)。SuperKEKB の運転開始から最初の 2 ヶ月は、アラーム監視設定の追加が特に多く行われた。その後 SuperKEKB の運転が徐々に安定になるにつれて、アラーム監視設定の変更件数は減っていった。レコードの登録数は、2 月上旬には 10,000 個を超える程度の数だったが、運転終了時には 14,000 個を超えるまでに増加した (Table 3)。現時点では、KEKB の 24,000 個には及んでいないが、この先 SuperKEKB の運転が Phase-2, Phase-3 へと進むにつれて、レコードの登録数がさらに増加することが予想される。

また、監視するレコードの増加や、IOC の構成変更などによって、当初アラームシステムと IOC との接続数は 50 個程度だったが、最終的には 91 個の IOC と接続する構成となった。

Table 2: Number of Modified Records

Month	Add	Delete	Update	Total
Feb 2016	2,340	95	0	2,435
Mar 2016	1,665	111	319	2,095
Apr 2016	16	332	12	360
May 2016	4	0	32	36
Jun 2016	150	0	127	277
Total	4,040	538	490	5,203

Table 3: Number of Monitoring Points

Group	KEKB	SuperKEKB (Phase-1 Start)	SuperKEKB (Phase-1 End)
Linac	815	1,014	1,004
BT (positron)	560	402	475
BT (electron)	675	462	581
MG (LER)	8,794	1,958	3,083
MG (HER)	5,497	1,488	2,564
RF (LER)	1,141	1,577	2,208
RF (HER)	1,656	1,386	1,892
VAC (LER)	2,190	142	150
VAC (HER)	1,169	134	178
Operation	408	109	119
BM	1,243	1,676	1,679
Safety	330	0	0
Abort (LER)	—	58	70
Abort (HER)	—	67	64
Total	24,478	10,473	14,067

5. CSS アラームシステムの問題点

5.1 レコードの値の桁溢れ問題

アラームシステムでは、アラームが発生した時のレコードの値を文字列形式でデータベースに保存しており、その値の長さを最大 100 文字として定義している。今回アラームサーバーが監視するレコードの中に、100 文字を超える値となるレコードが存在していたため、データベースへの登録時にエラーが発生する状況となった。この問題に対処するため、元のレコードのアラーム情報のみを継承する中間のソフトウェア IOC を用意し、アラームサーバーは中間の IOC を監視する構成へと変更した。

なおこの問題は、CSS 3.1.2 のバージョンで発生し、CSS3.2.16 では対策が行われていることが分かった。CSS 3.2.16 でも同様の動作を行うが、レコードの値が 100 文字を超える場合には、100 文字以降を切り捨てて登録するような処理が追加されており、データベースのエラーは発生しないように対策されていた。

5.2 レコードとの再接続不調

IOC の再起動や CA Gateway の再起動などによってレコードとの接続が切れた後に、レコードとの再接続が行われない症状がしばしば発生した。

通常、アラームサーバーはレコードとの接続切断を検知すると Disconnect 状態として処理し、再びレコードにアクセスできるようになると、自動的に再接続が行われてアラームの監視を再開する。しかし、再びレコードへのアクセスが可能となった後にも、Disconnect 状態のままレコードとの再接続が行われないという症状がしばしば発生した。一度この状況になると IOC や CA Gateway を再起動しても接続が復旧しないため、アラームサーバーを再起動してレコードと改めて接続し直す必要があった。

なお、この現象は特に CSS 3.1.2 のバージョンでよく発生しており、CSS 3.2.16 では概ね順調にレコードとの再接続が行われていた。

6. まとめ

今回 SuperKEKB のアラームシステムとして、我々は CSS 3.1.2 を正式運用とし、CSS 3.2.16 を試験運用として、2つのバージョンの並行運用を行った。SuperKEKB 加速器の運転が開始し、実際の運転環境下においてもアラームシステムは概ね順調に動作していたが、レコードとの接続が切断される状況が起きると、しばしば接続が復旧しない症状が発生した。今後、より安定したアラームシステムを実現するためには、レコードとの接続に関する問題点を改善する必要がある。今回試験運用としていた CSS 3.2.16 ではレコードとの再接続が安定して動作していたため、今後はこちらを正式運用として整備を進めたい。

6.1 今後の予定

CSS のプロジェクトでは、今回我々が使用した CSS 3 の次のバージョンである CSS 4 へと開発が移っている。その中でも、CSS 4.0.2 ではレコードとの接続周りの大幅な改修が行われていたり、またその後のバージョン

でも CA Gateway との接続に関するバグ修正など、日々開発が進められている。今後 CSS 4 についても動作試験や負荷試験を行い、SuperKEKB Phase-2 の運転に向け、より安定したアラームシステムを実現できるよう備えたい。

参考文献

- [1] Control System Studio
<http://cs-studio.sourceforge.net/>
- [2] Control System Studio (CSS) at KEK
<http://www-linac.kek.jp/cont/epics/css/>
- [3] T. Nakamura *et al.*, “SuperKEKB への CSS アラームシステム導入における性能評価”, Proceedings of the 12th Annual Meeting of Particle Accelerator Society of Japan, Fukui, Aug. 5-7, 2015.
- [4] Apache ActiveMQ
<http://activemq.apache.org/>
Apache は Apache Software Foundation の登録商標です。
- [5] PostgreSQL
<http://www.postgresql.org/>
PostgreSQL は PostgreSQL の登録商標です。
- [6] Python
<https://www.python.org/>
Python は Python Software Foundation の登録商標です。
- [7] T. Nakamura *et al.*, “PF-AR 加速器制御におけるアラームシステムの更新”, Proceedings of the 9th Annual Meeting of Particle Accelerator Society of Japan, Osaka, Aug. 8-11, 2012.
- [8] K. Yoshii *et al.*, “Web-Based Electronic Operation Log System - Zlog System”, Proceedings of ICALEPCS07, Knoxville, Tennessee, USA, Oct. 15-19, 2007.
- [9] Java
<https://www.java.com/ja/>
Java は Oracle Corporation およびその関連企業の登録商標です。

その他、本稿中に使われている会社名、製品名は各社の商標または登録商標です。