

MADDOCA 4.0 加速器運転端末システム更新の記録

RECORD OF SYSTEM UPDATES ON MADDOCA 4.0 ACCELERATOR OPERATION CONSOLE

岡田謙介^{#,A)}, 福井達^{B)}, 濱野崇^{A)}, 梶泰之^{C)}, 清道明男^{A)}, 杉本 崇^{A,B)}, 山鹿光裕^{A,B)}
Kensuke Okada^{#,A)}, Toru Fukui^{B)}, Takashi Hamano^{A)}, Yasuyuki Kaji^{C)}, Akio Kiyomichi^{A)}, Takashi Sugimoto^{A,B)},
Mitsuhiro Yamaga^{A,B)}
A) JASRI
B) RIKEN
C) SPring-8 Service Co., Ltd.

Abstract

In the MADDOCA 4.0 control framework, the accelerator is operated from multiple consoles in the central control room. Operators run GUIs to monitor equipment status from databases, and/or issue commands to the equipment. These consoles have used SUSE Enterprise 11 as their OS for over 15 years. Due to the challenges of installing an old OS to a new hardware, an OS update was planned. In conjunction with this, a migration of the GUI builder from old-fashioned X-Mate to Qt was also planned. The introduction of the new system was relatively smooth at NewSUBARU and NanoTerasu, where sufficient preparation time was available. However, the migration faced difficulties at SPring-8/SACLA because of the need to maintain existing GUIs while developing new ones under continuous user operation. Additional issues included dependencies on various configuration files, differences in C language compiler versions, and the porting of old MADDOCA-based GUIs. This presentation will explain the management policies for the MADDOCA 4.0 operation consoles, and the philosophy for maintaining the environment. It will also detail the system update process, including the trial introduction of new consoles and the progress in beamtime tests.

1. はじめに

SPring-8/SACLA で開発された制御フレームワーク MADDOCA4.0[1]は、中央制御室の運転端末から加速器の機器の状況を把握、制御を想定している。運転端末上のアプリケーションの多くは Graphical User Interface(GUI)で、GUIはMQTT broker[2]を介して加速器機器とのメッセージのやり取りを行い、機器状態はメッセージング及び、状態を記録するログデータベース(logDB)[3]を参照して把握する。また機器設定パラメータなどパラメータデータベース(parDB)[4]を利用して管理する。例えば SPring-8 Storage Ring (SR)の運転は、現在中央制御室(Fig. 1)の中央にコの字型に配置した運転端末の半分弱を占める 5 台を主に使用している。他は SACLA, 入射輸送系、または BL 制御のための運転端末が存在する。

これらの運転端末のオペレーティングシステム(OS)は15年以上も SuSE11[5]を採用してきた。昨今ハードウェアのライフサイクルに伴い、古い OS のサポートが切れ運転端末の維持が難しくなっている。健全な運転環境の維持管理、老朽化対策のため、OS の更新を計画した。

また、GUI アプリケーションとして同様に長い間 X-Mate[6]を利用してきた。長年の経験の蓄積で古参の運転員らは仕様に精通し、機能的にも今までのところ十分であるが、GUI 設計の自由度が限定されていること、シェアが小さいことから情報が少なく未経験者の参入障壁が高いという問題のため、今回の OS 更新に合わせて Qt GUI[7]への移行も進めることにした。

2 章では MADDOCA4.0 でのソフトウェア、ハードウェアの管理構成と、運転端末の運用方針を説明する。3 章では移行のための下準備として、X-Mate GUI 開発エキスパートの移行負担を減らすための開発を紹介する。4 章は実際の移行経過である。この試みは継続運転中の SACLA/SPring-8 では成功しなかったため、5 章で説明する再計画が必要だった。6 章は 2024 年に実施した SR 運転端末移行の詳細について述べ、7 章で SACLA 運転端末等の残件と今後の見通しについて触れる。最後に 8 章でまとめとする。



Figure 1: A panorama view of the SACLA/SPring-8 central control room.

2. システム構成

ここでは MADDOCA4.0 で加速器運転制御を行う上で、ユーザー(機器担当者や運転員)と制御グループの担当範囲における接点に注目してシステム構成を説明する。

[#] k.okada@spring8.or.jp

2.1 ソフトウェアの階層

Figure 2に示す通り、制御グループはMADOCA4.0のソフトウェアパッケージをメンテナンスし、ユーザーはそのDBやメッセージングアクセス関数をリンクして運転GUIのバイナリを作成する。基本はC言語の静的ライブラリを用いている。

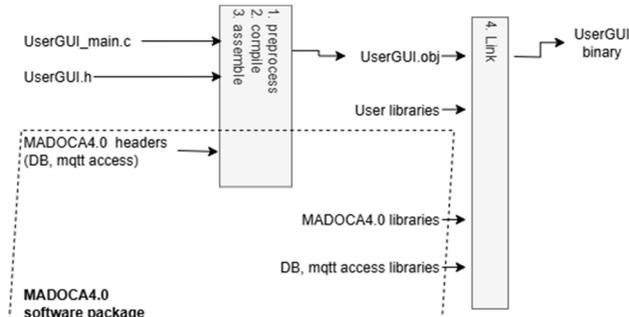


Figure 2: GUI SW structure. MADOCA4.0 software package covers the basic access to DB and mqtt broker.

2.2 ハードウェアの設定

運転端末は、ファイルサーバーをマウントし、NISでアカウント管理を行っている(Fig. 3)。SPRing-8では30年前の立ち上げ時の名残で各機器グループにアカウント(磁石:srmag, 真空:svvac, RF:srrf等)が割り当てられており改修前は一部のGUIはアカウント下に配置した設定ファイルを使用する関係上、特定のアカウントで立ち上げるようになっていた。

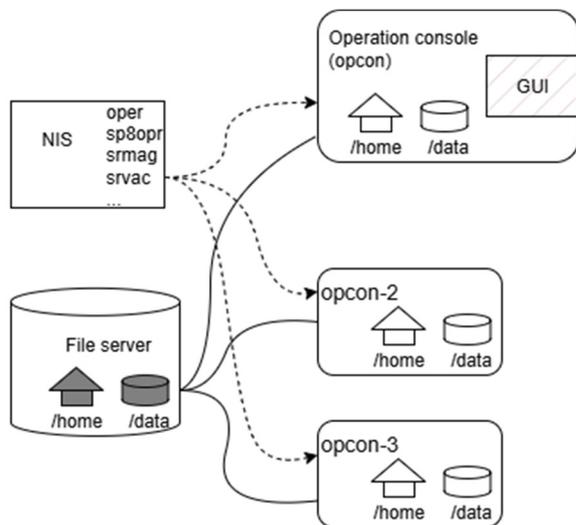


Figure 3: Operator console (opcon) HW setting.

2.3 運転端末運用方針

同一設定の運用端末を備えることで、端末故障時に別の端末でアプリケーションを立ち上げて運転継続が可能とし、予備機はネットワーク設定の変更で故障機と入れ替え継続性を確保している。

また操作ミスを防ぐため使用するGUIの操作は立ち上げた運用端末で行い、他の端末に画面を転送することは運用方法として推奨していない。

3. 移行にむけての下準備

3.1 次期オペレーティングシステムの選択

SuSEle11からの移行先の選定にあたって、運用端末は一年間連続運転されることを考慮した。テストの過程で、数ヶ月後に画面がフリーズするという問題で、SuSEle12、SuSEle15+gnome desktopは却下し、2020年時点ではSuSEle15+Xfce4[8]の環境を選択した。

その後SuSE licenseの取り扱い変更による使い勝手の改悪があり、現在はUbuntu22.04[9]+Xfce4を移行先として選択している。

3.2 GUIの移行準備

X-MateからQtへの移行に際しX-Mateで利用してきた加速器の制御に適した機能を実現する補助が必要と考えた。Table 1にQt開発環境で提供したアプリケーションを示す。X-Mateは描画部品を画面イメージに据えていくことで、それがボタンにもなり表示ボックスにもなるというプログラミング形式をとっていたが、それに近い形式を実現するためのQt Widget Designerのプラグインを数種と、logDBから取得した機器情報の最新値を並べて表示する用途を想定したテーブル管理補助、そしてプロセス間通信のための定型ライブラリを作成した。

一方X-Mateで特徴的だったアイコン化は、Qtでの実現を試みたもののコーディングが煩雑になるため割愛し、Xfceのwindow最小化機能に委ねることにした。

Table 1: Qt Apps for X-mate Users

項目	提供形態	内容
QShapeEX	Plugin & Static lib	基本図形の配置、ボタンとの連携
QLineEx	Plugin & Static lib	直線、曲線の描画、矢印などの作成
QSVGImageEX	Plugin & Static lib	SVG図形を貼り付け、部品の色 の制御、ボタン操作との連携
CQTableWidgetSource code tCtrl	Static lib	テーブル操作補助。複数テーブル の連携、カラムの色を操作、 値の取得等
MessageQueue	Static lib	posix queueを使ったプロセス間 通信補助

4. 運転端末システム移行計画

4.1 NewSUBARU (2021年5月運用開始)

入射器入れ替えのための運転シャットダウン期間を利用して、2020年末から集中的に既存のX-Mate GUIをQt GUIへ移植を行った。ロジック部分と描画部分が分離できていなかったため、実質ほぼ新規にコーディングするのと近い状況だった。その結果、運転開始時にはSuSEle15の運転端末上で全てQt GUIの体制を構築できた。

4.2 NanoTerasu (2024年4月運用開始)

新規建設の加速器であり、当初から運転端末はOSを

SuSEle15、GUIプラットフォームをQtとして計画し整備を行った。

4.3 SACLA/SPring-8

2021年の早い段階からSuSEle15の運転端末を制御室の隅に配置し、順次Qt GUIを実際の運用に投入、SuSEle11上のアプリケーションが不要になった時点で移行完了、との計画を立てたが、新規導入機器の単体試験的な利用を除けば、移行は進まなかった。

4.4 目論見と失敗の理由

SACLA/SPring-8で、運転端末の移行が進まなかった第一の理由は、NewSUBARU及びNanoTerasuの場合と違って、常にユーザー利用運転中であったことである。そのため、現在動作中のアプリケーションの置き換えの優先順位は常に低かった。第二として、運転に必須なアプリケーションが多い事、しかもSPring-8 SRの場合は古いものも多く、Fortranからの焼き直しの名残が残っているものも散見され、制作者がすでに引退していることもあった。このため作り直しのための新規コーディングの障壁が非常に高かった。第三にOS新規導入に当たって、不要なファイルで肥大化したHOMEディレクトリを一新したが、多くのアプリケーションでHOMEディレクトリ下に設定ファイルを使用し、またログを書き出すといった形でHOMEに縛られており、しかもpathがハードコーディングされているといった事情も移行を難しくする原因であった。

5. SACLA/SPring-8について再計画

SACLA/SPring-8の運転端末の移行をユーザー利用運転に支障をきたすことなく遂行するために以下の方針を立てた。

- OSはUbuntu22.04を採用しUbuntu22版のX-Mateを提供する。
- 旧HOMEディレクトリを同じマウントポイント(/home)としてアクセス可能とし、ログインHOMEは新規別の場所を作成する。(肥大化した/homeディレクトリの整理は一旦保留する)
- 加速器調整日を使ってアプリケーションの動作試験をする。

再計画の時点で、SuSEleのライセンス問題が発生しており、代替として広く使われ情報も多いUbuntu22.04を移行先候補として採用した。X-Mate GUIのQt GUIへの移行は運転端末のOS更新とは別に考えることとし、新規作成アプリケーションから順次進めていくことにした。X-Mateに関してUbuntu22.04上でのアプリケーションの作成、動作のための環境に加え、SuSEle11で作成したアプリケーションバイナリも動作できるように必要なランタイムライブラリを配置した。今回SPring-8 SRのアプリケーションについては、古いコードの見直しを兼ねてUbuntu22.04の開発環境での再コンパイルを行うことにした。

複数の運転端末で違うデスクトップ環境を動作させる際、環境設定情報がHOMEにお互い上書きされ、それぞれの環境を両立して維持できない問題があった。このため、Ubuntu22.04の環境でのログインはもとのHOME

ディレクトリとは別に用意した。複数の運転端末でHOMEを共有する問題については再度触れる。

運転に主に使用している5台のうち、2024年3月に1台にUbuntu22.04運転端末を先行導入、調整日を利用して試験を行い、夏または冬の加速器停止期間に全台移行という10ヶ月の計画を立てた。その他に最終的な保険として、元のSuSEle11端末も一台別に残しておく。

6. SPring-8 SR 運転端末移行実施

6.1 X-Mate GUIの再コンパイル

開発環境が古いSuSEle11からUbuntu22.04に移行し、gccのバージョンが4.3.4(August 4, 2009 release)から11.3.0(April 21, 2022 release)に上がった[10]。おおむねより厳格なコーディングが要求される傾向があり、ヘッダファイルに記述したグローバル変数定義を複数のオブジェクトで取り込むとリンク時に多重定義のエラーになるため一ヶ所以外はextern宣言に変更する他、コンパイル時に"-Wall"のオプションをつけてWarning出力を確認し、特に誤動作を招きやすいTable 2の内容を修正した。また時刻を扱うtm構造体の夏時間フラグの初期化が必要などが判った。

Table 2: Gcc Warning Example

Warning	出現箇所の例
stringop-overflow	strcpy 関連
aggressive-loop-optimizations	arrayの添え字ループ
array-bounds	arrayの添え字、特に最後
attribute-warning	fgetsで確保buffer size以上の読み込み
return-local-addr	ローカル変数のポインタ参照
return-type	Return文忘れ

6.2 動作試験について

アプリケーションの多くは動作条件として実際の機器の応答や加速器内のビーム条件が必要となっており、夏季停止直前のまとまった加速器調整枠を15時間割りあてて試験することにした。その後、残ったものの動作確認に1.2時間の加速器調整枠を複数回利用した。

6.3 動作試験経緯

SPring-8 SR運転に必要としてリストアップした73本のGUIアプリケーションのUbuntu22.4運転端末上の動作確認の経緯をFig. 4に示す。

夏季停止期間前の加速器調整日(7月31日)に向けて再コンパイルを進めバイナリの用意をした。当日は大部分(55本)の動作確認ができた。その他、後日の調整日に持ち越しになったものについてFig. 4の添え字に従って説明する。

- 真空関係の操作GUIは調整日には起動確認だけ行い、夏季停止期間中に動作試験を行った。
- 外部グラフ描画アプリケーション(gnuplot[11])のバージョンが異なっており、フォントの崩れやwindowのフォーカスが奪われる問題があった。テスト環境で

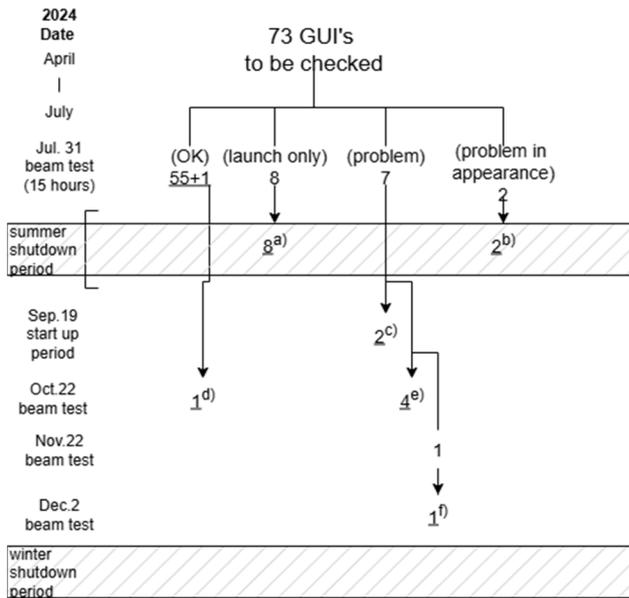


Figure 4: The history of SPring-8 SR GUI operation check of total 73 GUIs in 2024 (top to bottom). Details are in the main text.

その部分だけの試験を行って修正した。

- c) 電磁石操作に関する複数のプロセスが絡まった巨大な GUI で、操作命令が通らず即終了するような現象が起きた。症状をもとに当たりをつけてデバッグし、秋の運転再開の立ち上げ準備期間に実際の磁石操作をして動作確認を行った。
- d) 一旦動作確認済みとしたが、運転再開後、半日程度の運用でバグが出現した。抜けていた時刻 tm 構造体の初期化を追加し、次の調整日に動作確認と処理が止まった場合のアラート検知の動作確認を行った。
- e) 上記 b の gnuplot の仕様変更の対応、上記 c の磁石制御のライブラリ読み出しの関係と、そのほか時刻 tm 構造体の初期化を修正、旧 MADOCA 制御系の GUI の立ち上げ見直しを行った。

- f) これは旧 MADOCA 制御系の GUI で、立ち上げ設定の移行の問題と、さらに MADOCA4.0 との接続部分の問題があり、結果的に最大の4度の調整日を費やした。

以上、2024 年末までに全てのアプリケーションの動作確認を終え、冬の停止期間に SPring-8 SR の全 5 台の主に使う運転端末を Ubuntu22.04 仕様に移行した。

7. 今後の見通し

SACLA の運転端末について、2025 年の冬までに Ubuntu22.04 への移行を完了する予定である。SACLA の運転アプリケーションは比較的新しく、製作者の記憶もまだ期待できる。

SPring-8-II[12] (2027 年夏から建設のため SPring-8 ブラックアウト) では運転端末の運用は今回の Ubuntu22.04 と Qt GUI で始める計画となっている。

その先いずれまた更新が必要になるが、マイナーな X-Mate ではあまり問題にならなかった問題が想定される。それは Qt ではインターネットから便利な外部ツールが入手可能で、運転用アプリケーションに組み込まれ始めていることである。次の更新の際にその外部のツールの開発は止まってしまっているかもしれない、その場合 Qt のバージョン更新に追従できない危険がある。サポート可能な外部ツールを限定し、管理することが必要と思われる。

今回 SPring-8 SR の運転端末の移行の際、新旧の運転端末の並行運用にあたって、HOME の共有が問題になった。デスクトップの環境設定が HOME ディレクトリに置かれるため、異なる環境を同時に使用することができない。同じ問題は、他に python 実行環境を構築する場合でも起こる。今回は、ログインの際に別の HOME を割り当てることで対処したが、今後は運用アカウントを少数に絞り運転端末毎のアカウント管理とする方針が正解のように思われる。ただし HOME に運転に関する設定を一切置かないことが絶対条件となる。

次に運転に使用するアプリケーションを事前にテストベンチで全て動作試験できるのかについて。これは GUI と加速器の取り合い点の DB とメッセージング、機器の反

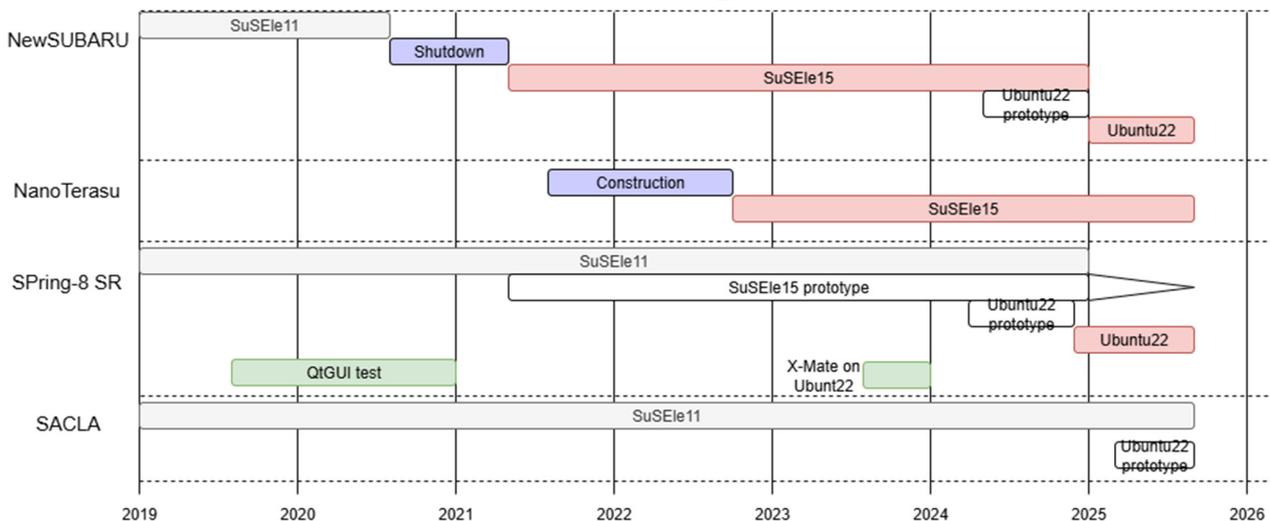


Figure 5: History of operator console upgrade.

応とその相関を再現できれば原理的には可能だが、その環境の作成と維持の労力は他に有効活用すべきであろう。もちろん問題点のあたりがついている場合のデバッグにはテストベンチは大変有用である。

8. まとめ

加速器の運転端末の老朽化対策の一環で長く使用した SuSEle11 OS を SuSEle15 もしくは Ubuntu22.04 に更新した。また長期的には X-Mate GUI の使用を止め Qt GUI に移行する計画である。

移行に際して準備期間が取れた NewSUBARU、及び NanoTerasu では問題なかったが、ユーザー運転中の SACLA/SPring-8 では新旧の運転端末を並行稼働させつつ調整期間のビーム運転を利用してアプリケーションの動作テストを行う必要があった。また大量の現 X-Mate GUI は、Qt GUI に改修せずそのまま新しい Ubuntu22.04 の運転端末で稼働させる方針に変更した。

以上の経緯を加速器毎に分類した Fig. 5 の年表に示す。

現時点で機能している運用端末の移行プロジェクトは短期的な利点が見えづらく実施タイミングが難しい。X-Mate GUI から Qt GUI への移行と組み合わせることで、見た目がスマートになったり新しい機能が使えたりと、移行の自然な方向付けにつながるかと期待したが、そうはならなかった。SPring-8-II プロジェクトに向けて今後も持続可能な運用に向けて努めていきたい。

参考文献

- [1] T. Sugimoto *et al.* "DEVELOPMENT OF NEW MADOCA CONTROL SYSTEM FOR SPring-8-II.", 16th International Particle Accelerator Conference, Taipei, Taiwan, 2025, pp.2942-2945.
- [2] <https://emqtt.io>
- [3] Kensuke Okada, Toru Fukui, and Toshiyuki Maruyama. "Assembling an user environment of accelerator log database at SPring-8/SACLA.", Proceedings of the 18th annual meeting of Particle Accelerator Society of Japan, 2021.
- [4] K. Okada *et al.*, "A database scheme to manage operational points and calibration values at SPring-8/SACLA.", Proceedings of the 16th annual meeting of Particle Accelerator Society of Japan, 2019.
- [5] <https://www.suse.com>
- [6] <https://www.fdsnet.co.jp/products/x-mate>
- [7] <https://www.qt.io>
- [8] <https://www.xfce.org>
- [9] <https://ubuntu.com>
- [10] <https://gcc.gnu.org/releases.html>
- [11] <https://www.gnuplot.info>
- [12] H. Tanaka *et al.*, "Green upgrading of SPring-8 to produce stable, ultrabright hard X-ray beams", J. Synchrotron Radiat., vol.31, Pt 6, pp.1420-1437, 2024. doi:10.1107/S1600577524008348