

# MADDOCA II 制御フレームワークの SPring-8 への実装及び機能拡張 IMPLEMENTATION OF MADDOCA II CONTROL FRAMEWORK INTO SPRING-8 AND THE EXTENSION FUNCTIONS

松本崇博<sup>#, A)</sup>, 古川行人<sup>A)</sup>, 石井美保<sup>A)</sup>, 松下智裕<sup>A)</sup>

Takahiro Matsumoto<sup>#, A)</sup>, Yukito Furukawa<sup>A)</sup>, Miho Ishii<sup>A)</sup>, Tomohiro Matsushita<sup>A)</sup>

<sup>A)</sup> Japan Synchrotron Radiation Research Institute (JASRI)

## Abstract

We developed MADDOCA II for the next-generation MADDOCA control framework at SPring-8 as reported at the last PASJ meeting. In MADDOCA II, new functions such as messaging with a variable length data, and controls on Windows, were implemented to have high flexibility in our control system. In this proceeding, we report the implementation of MADDOCA II control framework into SPring-8, which is our next milestone. On April 2014, we succeeded to start operation by replacing all operator workstations with MADDOCA II. The replacements of front-end computers (VME etc.) with MADDOCA II are now in progress and were partially performed in vacuum system etc. We also report the extension functions in MADDOCA II. To have high flexibility in access control at SACLA BL, the messaging scheme was extended to perform message routing via multiple hosts. To handle various data in the messaging with the unified procedure, we developed general-purpose data format using MessagePack.

## 1. はじめに

MADDOCA (Message And Database Oriented Control Architecture)は SPring-8 において独自に開発された分散制御のための制御フレームワークであり[1]、SPring-8をはじめ、HiSOR、SACLA等の放射光施設における加速器・ビームラインの制御、及び実験データ収集系の制御へと幅広く利用されている。MADDOCAを用いた制御系はSPring-8 供用開始から17年間にわたり安定運用されてきたが、現在の状況においては、設計当初と比べ機能面において改善すべき項目が明らかになってきた。例えばMADDOCAではメッセージ通信にONC/RPC及びSystem V IPCを用いるが、このことによりWindowsへの移植を困難にしていた。これらの課題を解消し高度な柔軟性をもつ制御を実現するため、メッセージ通信のライブラリをZeroMQ[4]に刷新した次世代の制御フレームワークMADDOCA IIを構築した。[2][3]

MADDOCA II 制御フレームワークの活用例はFigure 1に示される。MADDOCA IIでは、MADDOCAと同様にSVOC構文に基づくメッセージのやりとりにより分散制御を行う。さらに、メッセージと共に画像データや波形データ等の可変長データも送受信できる。また、制御に用いるプラットフォームはLinuxやSolarisの他にWindowsも対応することから、C、C++言語の他にLabVIEWを介したMADDOCA IIでの制御も可能である。

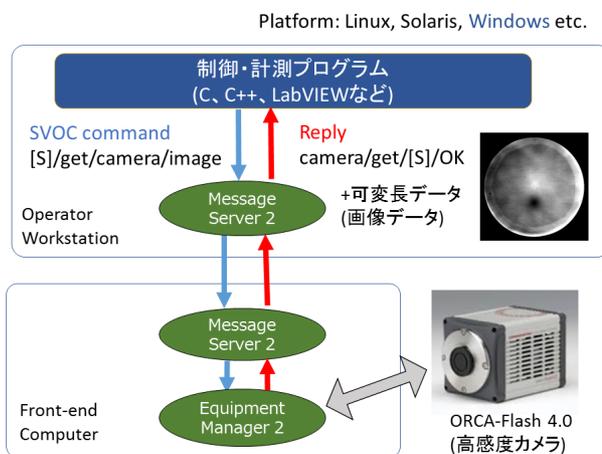


Figure 1: An application with MADDOCA II control framework. ([S] in the SVOC command is composed from process number and name of application, account and host. The [S] is automatically determined inside the control framework.)

第10回加速器学会ではMADDOCA IIの利用例として、SPring-8において2次元干渉計及びBPMシステムにおいて画像や波形データをメッセージングで扱った例について報告した。[3][6]この部分導入によりMADDOCA IIの安定性や新機能を確認してきた。次に我々はSPring-8制御システムのメッセージングの全系をMADDOCA IIへ移行する。本プロシーディングスではSPring-8におけるMADDOCA II制御フレームワークへの移行について報告する。

また、MADDOCA IIの新たな機能拡張、多段ホストメッセージング及び汎用データフォーマットについて報告する。

<sup>#</sup> matumot@spring8.or.jp

## 2. SPring-8 への MADOCA II 制御フレームワークの実装

SPring-8 の加速器・ビームライン制御システムは約 80 ホストの運転端末及び約 450 ホストのフロントエンド計算機(VME 等)から構成される。このように多くの計算機から SPring-8 の制御系が構成されるため、MADOCA II 移行を一括で行うことは現実的ではない。加速器運転に支障を与えない形で段階を経て MADOCA との共存も考慮しながら制御系を更新していく必要がある。

これらの点を考慮し Figure 2 に示されるようにまず運転端末側(GUI 側)の全てを MADOCA II に移行する方針を採用した。これは、運転制御に用いる加速器の運転端末は混乱をさけるため同一の制御方式に速やかに移行しておいた方が好ましいこと、運転端末の OS は全て同一の SUSE Linux Enterprise 11 で運用されているため移行がやりやすいことが理由として挙げられる。フロントエンド計算機側では MADOCA の制御系も存在する。MADOCA II の運転端末から MADOCA の Equipment Manager と ONC/RPC 経由で制御できるように、運転端末において MADOCA の Access Server と同等の機能を持つ Access Server 2 を別途設置して対応した。MADOCA II 用に用意した関数は MADOCA 用の関数と後方互換性を持たせている。よって MADOCA II アプリケーションは、プログラムのソースは変更せずに MADOCA II 用ライブラリを指定して再コンパイルするのみで作成可能にした。この互換性により運転端末側の GUI 作成は速やかに遂行できた。

一方、フロントエンド計算機側では多様な機器を制御するため Solaris、Linux、ARM 計算機等さまざまな種類の OS が利用されていることから MADOCA II 化の可否は明白ではない。MADOCA II では ZeroMQ と MessagePack のライブラリを用いるが、MessagePack では gcc4.1 以上のコンパイル環境が必要となるため古い OS において MADOCA II 移行が難しい場合もある。このため、個別に MADOCA II 対応の検討を行う必要がある。

SPring-8 において一番重要な役割を担うフロントエンド計算機は VME であり、その OS には Solaris が利用されている。多くの VME CPU では Solaris 10 または Solaris 9 の OS が利用されているが、これらに関しては MADOCA II 専用の Solaris 開発環境を整備することにより MADOCA II 対応ができることがわかった。しかし、初期に導入された Solaris 8 の VME CPU ボードに関しては、古い OS のため MADOCA II 化難しい状況にあった。また、VME CPU ボードが古い機種のため OS の更新も困難であった。この課題については、該当する VME CPU ボードの台数が少ないことから VME CPU ボードを全て新機種に変更し OS を Solaris 10 に更新することで解決した。現在では VME 系は全て MADOCA II

対応が可能な状況になっている。

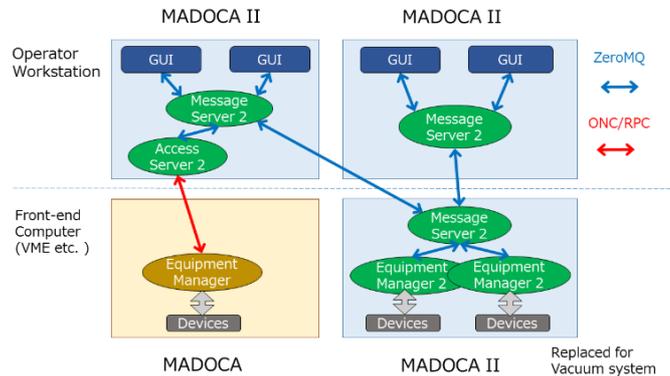


Figure 2: Implementation of MADOCA II control framework at SPring-8.

他に MADOCA II 化の目途が立ったフロントエンド計算機は横河 Linux PLC 及び  $\mu$ TCA の制御システムが挙げられる。組み込み計算機系はまだ未対応であり今後 MADOCA II 対応を進めていく予定である。Windows に関しては、LabVIEW 制御系の MADOCA II 化は目途が立っている。一部、バンチカレント測定において Cygwin を用いた計測システムがあり別途 MADOCA II 化の検討が必要とされる。

2014 年 4 月からの SPring-8 の運転では、全ての運転端末と一部のフロントエンド計算機を MADOCA II に更新させることで運用を開始することにした。現在、MADOCA II 化がされているフロントエンド計算機は真空関係と先に導入した BL36XU、2次元干渉計及び BPM システムが挙げられる。運転端末側の一括での MADOCA II 移行に関しては 事前に 2013 年 9 月から SACLA の実験データ収集の制御システムに導入し知見を得ている。今回はフロントエンド計算機側も一部 MADOCA II に更新することから、MADOCA と MADOCA II の混在環境への対応も行った。

Figure 2 において Access Server 2 は特定のグループに属するホストの Equipment Manager と ONC/RPC で通信を行う。しかしながら、特定グループに属するホストには MADOCA のホストのみでなく MADOCA II のホストも存在する。ONC/RPC で通信するホストを MADOCA に限定するため、各ホストが MADOCA か MADOCA II であるかの識別をデータベースに登録し管理する方式をとった。

MADOCA II での安定運用を担保するため、各種診断ツール類も整備した。Figure 3 は MADOCA II ・MADOCA 接続監視モニタである。各運転端末に設置され、運転端末に接続された MADOCA または MADOCA II のフロントエンド計算機が正常稼働しているかどうかを確認することができる。異常があるホストは監視モニタの上位に表示されるためシステムの健全性を容易に把握することができる。



る。

本実装は試験環境における試験において正常動作が確認されている。今秋から SACLA BL 制御システムにおいて運用が開始される予定である。

#### 4. メッセージング用汎用データフォーマット

MADDOCA II では通常の SVOC 構文に基づくメッセージの他に画像や波形データなどの可変長データも送受信することができる。これらのデータはメッセージングで送受信する際には MessagePack を用いてデータをシリアライズしている。MessagePack は自己記述的なデータ形式であり、シリアライズされたデータにはそのデータ構造も含まれている。このため、MessagePack を用いることで原理的には様々なデータ形式を柔軟に扱うことができる。

しかしながら、この高い自由度のため、アプリケーションを介してデータがどのような構造であるのかを理解しながら扱うのはなかなか難しい。

このため、多様なデータをより安易に統一の形式で扱えるようにメッセージング用の汎用データフォーマットを構築することにした。汎用データフォーマットでは データ型として MessagePack の Map 型を採用した。データはキー名とそのデータ値が関連付けられて構成される。また、Map のツリー構造も実装できるようにしている。

任意のキー名とデータ値の組み合わせを複数指定することでメッセージング用の可変長データを構築することもできるが、よく使われるデータタイプ、画像、波形、raw データ等に関しては、扱うキー名とデータ型のセットを予め規定した。画像データにおける汎用フォーマットの例を Table 1 に示す。

現在、汎用データフォーマットのライブラリ (MsgpackMadoca) は C/C++ 言語のライブラリで提供しており、SPring-8 においては 2 次元干渉計や COD データにおけるデータフォーマットに利用されている。また、LabVIEW でも汎用データフォーマットを利用できるように現在 VI の整備を進めている。

Table 1: General-purpose Data Format for Image Data

キー名	データ型	データ値
<b>image_data_type</b>	string	MONO,RGB,RGBA
<b>image_width</b>	int32_t	
<b>image_height</b>	int32_t	
<b>image_depth</b>	int32_t	
<b>image_num_type</b>	String	uint8_t,uint16_t,uint32_t, uint64_t,int16_t,int32_t, int64_t,float,double
<b>image_data</b>	[image_num_type で指定]	
<b>image_pixel_order</b>	string	lefttop,leftbottom

#### 4. まとめ

我々は SPring-8 制御システムの MADDOCA II 制御フレームワークへの本格的な移行を開始した。2014 年 4 月より全ての運転端末、及び真空系のフロントエンド計算機を MADDOCA II システムに置き換え安定運用を実現することができた。残りのシステムも順次 MADDOCA II 移行を進めていく予定である。主に利用している VME に関しては MADDOCA II 移行の目途を立てることができた。機能拡張に関しては、多段ホストメッセージング、及び汎用データフォーマットを実装した。SACLA BL のようなファイアーウォール越しの制御、及び実験計測系におけるデータハンドリング等さまざまな側面において本機能の活用が期待される。

#### 謝辞

SPring-8 制御システムへの MADDOCA II メッセージング移行に関しては(公財)高輝度光科学研究センター制御・情報グループの方々には有益な討論、ご意見を頂きました。ここに深く感謝致します。

#### 参考文献

- [1] R. Tanaka, et al., "The first operation of control system at the SPring-8 storage ring", Proceedings of ICALEPCS1997, Beijing, China, 1997, p.1.
- [2] T.Matsumoto, et al., "Next-Generation MADDOCA SPring-8 Control Framework", Proceedings of ICALEPCS2013, San Francisco, California, USA, 2013, p.944.
- [3] 松本崇博, 他, "SPring-8 における新しい制御フレームワーク MADDOCA II の開発", 第 10 回加速器学会年会プロシーディングス, p.14.
- [4] <http://zeromq.org/>
- [5] <http://msgpack.org/>
- [6] 古川行人, 他, "MADDOCA II-LabVIEW インターフェースを用いた BPM 読み出しシステムの開発", 第 10 回加速器学会年会プロシーディングス, p.239.