

## J-PARC ハドロンビームライン用データアーカイブシステムの開発

### DEVELOPMENT OF DATA ARCHIVE SYSTEM FOR J-PARC HADRON BEAMLINE

豊田晃久<sup>#</sup>, 上利恵三, 青木和也, 家入正治, 岩崎るり, 加藤洋二, 森野雄平, 里嘉典, 澤田真也, 白壁義久, 高橋仁, 田中万博, 広瀬恵理奈, 皆川道文, 武藤亮太郎, 山野井豊, 渡辺丈晃  
Akihisa Toyoda<sup>#</sup>, Keizo Agari, Kazuya Aoki, Masaharu Ieiri, Ruri Iwasaki, Yohji Kato, Yuhei Morino, Yoshinori Sato, Shinya Sawada, Yoshihisa Shirakabe, Hitoshi Takahashi, Kazuhiro Tanaka, Erina Hirose, Michifumi Minakawa, Ryotaro Muto, Yutaka Yamanoi, Hiroaki Watanabe  
KEK

#### Abstract

The Channel Archiver is used as a data archive system of J-PARC hadron experimental facility. However the Channel Archiver is not maintained for nearly 10 years, so that we need a shift to the next term archive system. As the new archive system, we tested a RDB Channel Archiver which is one of the components of the CSS (Control System Studio) and compared the performance with the present archive system of the Channel Archiver. Test outcome of the data consumption rate, the performance by the server load, the high load capability, and so on will be presented.

#### 1. はじめに

J-PARC ハドロン実験施設では 2009 年の運転開始以来データアーカイブシステムとして、Channel Archiver [1]を使用してきた。この Channel Archiver は EPICS (Experimental Physics and Industrial Control System) [2]のアーカイブツールであり、EPICS のネットワークプロトコルである Channel Access を通して得られる全ての値を記録することができる。しかしこの Channel Archiver は 10 年近く全くメンテナンスされていないため、メンテナンスされている別の新しいアーカイブシステムへの移行が必要になっている。

我々はその新アーカイブシステムとして CSS (Control System Studio) [3]に付属する RDB (Relational Data Base) Channel Archiver を採用した。CSS はユーザーインターフェースのための共同プロジェクトであり、加速器ではデータ履歴表示や制御、アラーム管理など運転に必要なインターフェースを統合したシステムとして使用している。ハドロン実験施設ではユーザーインターフェースとして CSS を採用していないが、その一部である RDB Channel Archiver を試用した。この RDB Channel Archiver も上記の Channel Archiver と同様 EPICS Channel Access を通じて得られる値を記録するもので、一番の違いはデータの保存先が RDB であるということである。今回この RDB Channel Archiver の性能試験を行ったので報告する。

#### 2. アーカイブシステム構成

Figure 1 に現在のハドロン実験施設のアーカイブシステムの構成を示す。ハドロン制御 VLAN は Gb ethernet によって J-PARC 全体制御システムの下につながっている。従来の Channel Archiver は sampling machine (server06)、serving machine (server02)とストレージ(nas1, NAS (network attached storage), 2 TB, RAID6 構成)から構成されている。今回導入した RDB Channel Archiver

は server machine (server10)とストレージ(nas2, NAS, 10 TB, RAID6 構成)から構成されている。またデータ解析マシンとして、server08 を別途用意した。EPICS channel を提供する EPICS IOC(Input Output Controller)が立ち上がっている各種機器は実験ホールや制御室などの Gb Hub の下にある。server 4 台はそれぞれ購入時期が異なるため性能が異なり、番号が大きいほど性能が良いものとなっている。OS に関しては、ストレージは Windows Storage Server を、サーバーは Scientific Linux を採用している。

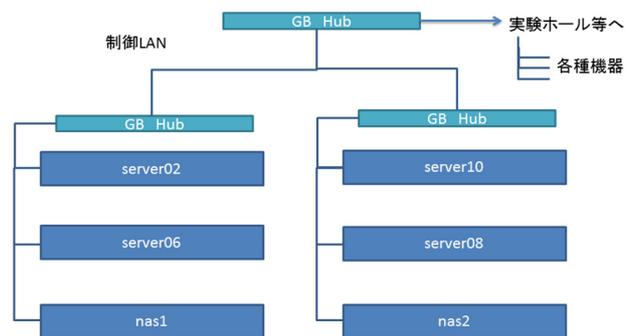


Figure 1: Archive system framework.

Channel Archiver は前述の通り sampling machine と serving machine に分かれている。sampling machine は EPICS Channel Access によって取得した値をストレージに書き込む。書き込み遅延防止の目的で、アーカイブプログラムである ArchiveEngine は複数用意され、同時に稼働している。serving machine は XMLRPC プロトコルを通じて client から要請された値をストレージから読み込んで提供する。データ構造は Channel Archiver 独自のバイナリ形式となっている。2009 年の運転開始から現在まで運用している。

RDB Channel Archiver は server machine1 台で値を取得、記録、およびデータの提供を行っている。Channel Archiver と同様に ArchiveEngine を複数同時に運用している。データ構造は RDB 形式となっている。client は

<sup>#</sup> akihisa.toyoda@j-parc.jp

RDB のプロトコルでデータを取得する。RDB は Oracle, MySQL, PostgreSQL が使用できるが、今回は MySQL を採用した。2015 年 10 月から試験運用を開始して現在まで運用している。

データ取得する EPICS channel 数としては、Channel Archiver, RDB Channel Archiver とともに 3400 channel 程度である。サンプリング周期はさまざまに 0.1 秒から約 1 分まで幅広い。EPICS Channel あたりのデータワード数も 1 word から 20000 word 程度まで幅広い。

### 3. 試験内容とその結果

試験としては以下の内容で行った。

試験期間:2015 年 10 月から 12 月、2016 年 5 月から 7 月。一部ビーム運転終了後を含む。運転に伴って 2016 年は 2015 年に比べて若干 channel 数が増えている。

試験内容:

- データ使用率の測定
- 書き込み速度の測定
- データ一致測定
- データ取得速度測定

以下の節でそれぞれの試験内容と結果を述べる。

#### 3.1 データ使用率の測定

Figure 2 に Channel Archiver (xmlrpc)と RDB Channel Archiver (mysql)のデータ使用量の比較を示す。縦軸はデータ使用量(MB)で、横軸は日付であり、ビーム運転中のデータ使用量を示している。この図から

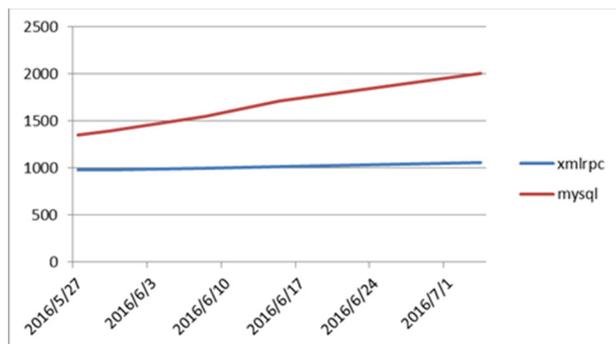


Figure 2: storage consumption rate.

Channel Archiver は 2.3 GB/day のデータ使用率に対して RDB Channel Archiver は 18.6 GB/day と約 8 倍の消費になっていることが分かる。新アーカイバーのデータ消費率が高いが、現状多すぎるということはない。新アーカイバーの使用量の内訳としては、全体の 35%程度が index、残り 65%がデータとなっている。また scaler のデータが全体の 26%となっていて、残り 74%が array のデータとなっている。

#### 3.2 書き込み速度の測定

この節では書き込み速度が十分かどうかを調べる。Figure 3 は Channel Archiver (xmlrpc)と RDB Channel Archiver (mysql)の各 ArchiveEngine プロセス(TEMP,

OTR, ... , SCALER) の idle 時間をプロットしたものである。RDB Channel Archiver は 100%で余裕があるのに対して、Channel Archiver は 88-96%程度と負荷が見えている。各 ArchiveEngine のカテゴリは信号の種類(温度、真空、モニターなど)で分けているので、図から分かるように負荷は均等ではない。

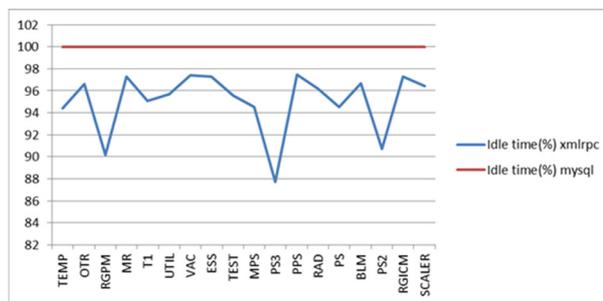


Figure 3: Process idle time.

Figure4 は Channel Archiver(xmlrpc)と RDB Channel Archiver (mysql)の書き込みにかかった時間(秒)を示す。どちらも 30 秒に一度書き込む設定となっているため、30 秒以上時間がかかっていると書き込みが間に合わないことがあることを示している。RDB Channel Archiver は速度に余裕があるため、PS2 と PS3 は統合され、PS2 となっている。Channel Archiver は TEMP, T1, PS3,PS など書き込み時間が厳しい ArchiveEngine があることが分かる。詳細については次の節で議論する。

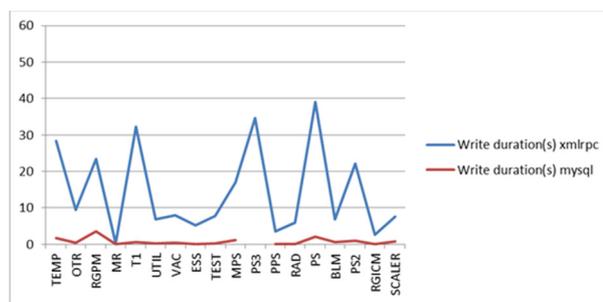


Figure 4: Process write duration time.

#### 3.3 データ一致測定

前節で書き込み遅延の問題があったため、全データ一致解析を行った。具体的には Channel Archiver と RDB Channel Archiver のデータを取得し、一致しているかどうかを調査した。よって両方とも記録できなかった場合は検知できない。期間は 2015 年 11 月から 12 月(約 30 万 shot)、2016 年 5 月から 6 月(約 16 万 shot)の運転データを解析した。解析対象は以下の通り。

- Scaler データとしてビーム強度を採用。1 spill(5.52 秒)に 1 回更新(1 word/spill)のため、低負荷のデータの代表として解析した。
- Array データとしてビームプロファイルを採用。1 spill に 1 回更新だが、データが多いため 64 word/spill のデータ量である。
- FastSampling データとして電磁石電源電流を採用。これは scaler データだが、0.1 秒に一回更新

なので 1 word/0.1 秒 = 55.2 word/spill のデータ量。更新スピードが速いのでデータが欠けやすい。これだけはデータが多いため、spill の切れ目のデータだけをピックアップして比較した。

それぞれに対して全データ解析を行った。Channel Archiver にはデータ切り替えに伴う記録途切れが 1 週間に 1 回(約 3 分間)あるため、その時間帯を除いて解析した。

まず 2015 年のデータについて解析した。Scaler データでは 1 spill だけ Channel Archiver の記録がないところがあったがそれ以外はすべて一致した。Array データではすべてのデータが一致した。それに対して FastSampling データでは多数の不一致が観測された。Channel Archiver 側に欠損が見られ、RDB Channel Archiver には欠損は見られない。時間帯には偏りが見られ、5 時前後と 6 時前後にのみ起きている。欠損期間は 4 秒から 4 分程度と幅がある。同時に Channel Archiver のログにエラーが記録されていることが多いが必ずエラーが出ているわけでもない。

続いて 2016 年のデータについて解析した。ちなみに 2016 年の 6 月には数回ネットワーク通信の不調が生じている。まず Scaler データについては 1 spill だけデータが入っていないイベントがあったがその他は完全一致した。上記の通信不調の時間帯にはデータ欠けはなかった。影響を受けていないのか、Channel Archiver, RDB Channel Archiver の両者ともに影響を受けているのかのどちらかである。次に Array データについてはすべてのデータが一致した。こちらも通信不調の影響がなかったか、両者に影響があったかのどちらかである。最後に FastSampling についてだが、2015 年同様に 5 時前後と 6 時前後には Channel Archiver に欠損が多数みられる。それ以外だと 7 時前後に 2 度 Channel Archiver に欠損が見られた。また上記ネットワーク不調時間帯には RDB Channel Archiver 側に欠損が見られた。

上記の結果、サンプリングが遅い(1 spill に 1 回程度)場合はほぼデータが欠けないことが分かったので、ビームデータを取得している時間帯のデータに対して同じ解析を行った。これは読み込み負荷がかかった状態における書き込みデータの健全性のチェックになる。結果は全データ一致したため、この程度の読み込み負荷がかかった状態であれば問題なく書き込みが行えることが分かった。ハドロン実験施設では通常運転中は履歴をアーカイブから取得せず、随時トレンド情報を更新するのみである。何かトラブルがあった場合(例えば注意報発報など)に履歴参照する使い方なので、上記の試験程度の負荷が予想されている。よって現状では性能に問題はない。

サンプリングが速い場合は、Channel Archiver にはデータ欠けがあるが、RDB Channel Archiver には(通信不調時を除いて)データ欠けはない。主な原因としては、システムが古く処理能力が不足していることが挙げられる。ハドロン実験施設では 2015 年 4 月の再開からこのような速いサンプリングのデータを扱うようになった。それ以前は spill 毎のデータのみであったため書き込み速度に問題なかったと考えられるが、現在は新アーカイブシステムの性能が必要になったと言える。

### 3.4 データ取得速度測定

最後にデータ取得速度測定を行った。前節で測定したデータセットのうち、Scaler データと Array データの 2 つのデータセットに対して条件を変えつつ測定した。FastSampling データについてはデータ欠けの影響により長い時間連続してデータ取得することができず測定精度が悪いため行わなかった。データ取得クライアントには python-xmlrpc および python-MySQL ライブラリを利用した。ちなみに MySQL の index については適切に働いていることをチェック済みである。

データセットの条件の説明を以下に示す。ちなみに第 2 節にあった通り、Channel Archiver の xmlrpc サーバーは server02 で、RDB Channel Archiver の mysql サーバーは server10 である。

1. Scaler データ(クライアントは server10)。クライアントが server10 なので、Channel Archiver に対しては書き込みによる負荷があり、Network 負荷もある。RDB Channel Archiver に対しては書き込みによる負荷があり、Network 負荷はない。
2. Array データ(クライアントは server10)。クライアントが server10 なので、Channel Archiver に対しては書き込みによる負荷があり、Network 負荷もある。RDB Channel Archiver に対しては書き込みによる負荷があり、Network 負荷はない。
3. Array データ(クライアントは server08)。クライアントが server08 なので、Channel Archiver に対しては書き込みによる負荷はなく、Network 負荷はある。RDB Channel Archiver に対しては書き込みによる負荷はなく、Network 負荷はある。
4. Array データ(クライアントは server02)。クライアントが server02 なので、Channel Archiver に対しては書き込みによる負荷はなく(server06 で書き込んでいるため)、Network 負荷もない。RDB Channel Archiver に対しては書き込みによる負荷はなく、Network 負荷はある。
5. Array データ(クライアントは server10 で運転終了後に測定)。2 と同じだが、書き込みが行われない運転終了後にデータ取得した。よって Channel Archiver に対しては書き込みによる負荷はなく、Network 負荷はある。RDB Channel Archiver に対しては書き込みによる負荷はなく、Network 負荷もない。

以上の内容を表にまとめたのが Table 1 である。

Figure 5 にデータ取得時間計測のためのデータのフィットの様子を示す。横軸が取得するデータ点数、縦軸がかかった時間(秒)である。データ点 60000 点(Scaler データで 6 万 word, Array データで約 400 万 word が約 4 日弱分の運転時間に相当する。青点が RDB Channel Archiver の結果で、赤点が Channel Archiver の結果である。それぞれ 1 次関数でフィットしてデータ取得レートを評価した。

Table 1: Measurement Condition

データセット	Channel Archiver		RDB Channel Archiver	
	書込 負荷	Network 負荷	書込 負荷	Network 負荷
Scaler データ (クライアントは server10)	有	有	有	無
Array データ (クライアントは server10)	有	有	有	無
Array データ (クライアントは server08)	無	有	無	有
Array データ (クライアントは server02)	無	無	無	有
Array データ (クライアントは server10 で運転終了後に測定)	無	有	無	無

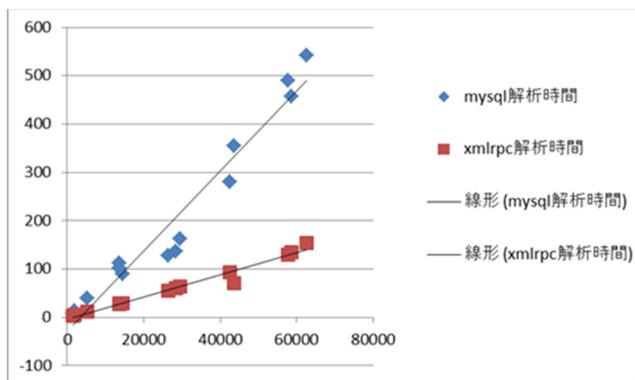


Figure 5: Fitting of data acquisition time.

Table 2 にフィットした結果を示す。それぞれ一番左の列のデータセットに対する Channel Archiver と RDB Channel Archiver のデータ取得レート(points/s)を示している。以下に結果の考察を述べる。

Scaler データについては RDB Channel Archiver の方が少し早いですが、Scaler データは RDB Channel Archiver データの不定性が若干大きく精度が低いので、Array データの方が、信頼度が高い。Array データについては Channel Archiver の方が約 4 倍速い。Scaler データとの比較では、データ量は 64 倍違うが、スピードは 3 倍から 20 倍の範囲なので、データが増える比率の分だけ時間がかかるわけではない。

クライアントサーバーへの依存性については、どちらも速度として server10 が一番早く、server02 が一番遅いのでクライアントの性能に主に依存しているように見える。ちなみに計測時のネットワークトラフィック負荷は 0.5

MB/s 程度で速度遅延するほどではなく、結果もそれを反映している。

運転中の書き込み負荷の影響については、運転終了後 Channel Archiver 側で 1 割程度速度が上昇するので、その程度の影響であると言える。RDB Channel Archiver では 4 倍弱速度が改善しており、これは query 処理がぶつかって時間をロスしている分が大きいことを示している。

Table 2: Fitting Result

データセット	Channel Archiver の DAQ レート (points/s)	RDB Channel Archiver の DAQ レート (points/s)
Scaler データ (クライアントは server10)	$1.4 \times 10^3$	$2.2 \times 10^3$
Array データ (クライアントは server10)	$4.4 \times 10^2$	$1.2 \times 10^2$
Array データ (クライアントは server08)	$2.4 \times 10^2$	$9.1 \times 10^1$
Array データ (クライアントは server02)	$1.9 \times 10^2$	$8.2 \times 10^1$
Array データ (クライアントは server10 で運転終了後に測定)	$4.8 \times 10^2$	$4.3 \times 10^2$

#### 4. まとめ

データアーカイブシステムとして従来の Channel Archiver に加えて RDB Channel Archiver を試験運用し、性能評価を行った。RDB Channel Archiver では Channel Archiver で見られる速いサンプリング時のデータ欠け問題がないことが分かった。この原因は主にサーバーの能力によっていると考えられる。また RDB Channel Archiver はデータ量が約 8 倍多いという欠点があるが、20 GB/day 程度で特に問題はない。また RDB Channel Archiver は、運転時に Channel Archiver より 4 倍弱読み出しが遅いという欠点がある。これは query がぶつかっていることが原因と考えられる。今後はリプリケーションを行うことでこの問題の改善を図りつつ正式運用に移行する。それと並行して別のアーカイブシステムを試す予定である。

#### 参考文献

- [1] Channel Archiver Web page; <https://ics-web.sns.ornl.gov/kasemir/archiver/>
- [2] EPICS Web page; <http://www.aps.anl.gov/epics/>
- [3] Control System Studio Web page; <http://controlsystemstudio.org>