



# J-PARC MR制御システムの 15年間の運用と今後の展望

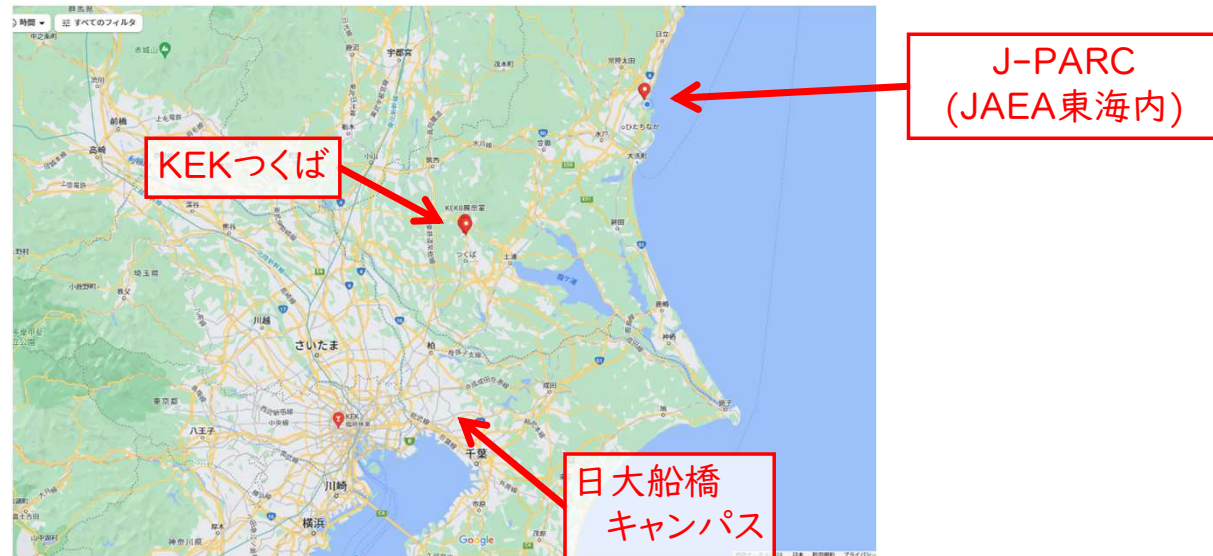


KEK/J-PARCセンター  
山田秀衛

- はじめに
  - J-PARC加速器とその制御
  - J-PARCの加速器制御システム
- 制御システムのインフラ
  - ネットワークとセキュリティ
  - ストレージシステム
- 制御計算機
  - サーバと仮想化
  - 制御端末
  - フロントエンド計算機 (IOC)
- 制御用ソフトウェア
  - OS
  - EPICS
- まとめ

## はじめに

- J-PARC : Japan Proton Accelerator Research Complex
- 大強度陽子加速器施設
- KEK(高エネルギー加速器研究機構)とJAEA(日本原子力開発機構)が共同で計画・開発・運営
  - 2001: 建設開始、2007: 運転開始



# J-PARC加速器とその制御

## • 3つの実験施設

ニュートリノ  
実験施設  
(NU)

物質・生命科学  
実験施設  
(MLF)

ハドロン  
実験施設  
(HD)



## • 3つの加速器

LINAC  
400 MeV  
全長330m  
1st beam in 2006

RCS  
(Rapid Cycling Synchrotron)  
3 GeV  
周長350m  
@ 25Hz

Main Ring Synchrotron  
30 GeV  
周長1600m  
@ 1.36 sec (FX to NU)  
@ 4.20 sec (SX to HD)  
1st beam in 2008

今日は主に  
MR加速器制御の  
はなし

EPICS

EPICSで3つの加速器を制御  
2005年ころから開発開始

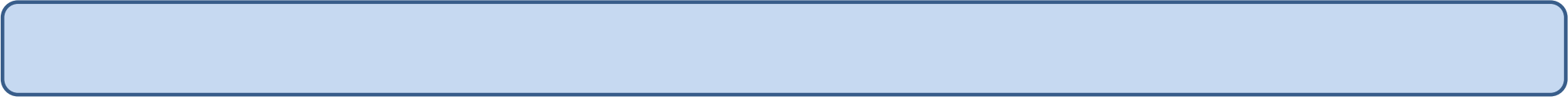
## J-PARCの加速器制御システムの特徴

- ふたつのマシンサイクル用の制御システムが緊密に連携している
  - 25Hz (Li&RCS) : JAEAが担当
  - 1.36s, 4.20s (MR) : KEKが担当
  - 共通の基盤の上に構築されている
- 共通のもの
  - タイミングシステム
  - ネットワークとストレージ
  - EPICS
  - (PPS, MPS)
- 独立しているもの:
  - 制御端末
  - サーバ計算機
  - フロントエンド計算機 (IOC)
  - OS (特にIOCのOS)
  - アーカイブシステム

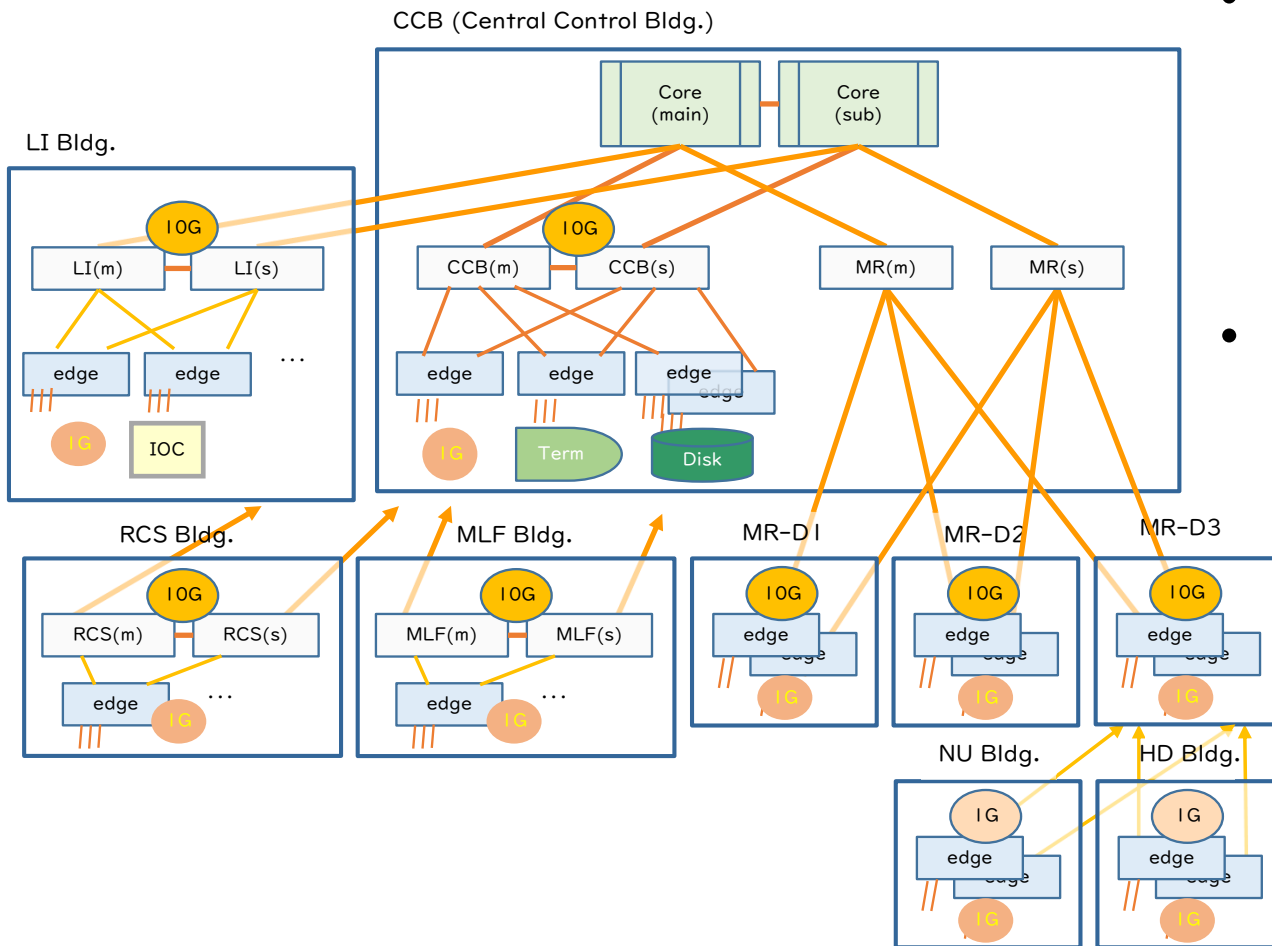
# J-PARCの加速器制御システムの特徴

- ふたつのマシンサイクル用の制御システムが緊密に連携している
  - 25Hz (Li&RCS) : JAEAが担当
  - 1.36s, 4.20s (MR) : KEKが担当
  - 共通の基盤の上に構築されている
- 共通のもの
  - タイミングシステム
  - ネットワークとストレージ
  - EPICS
  - (PPS, MPS)
- 独立しているもの:
  - 制御端末
  - サーバ計算機
  - フロントエンド計算機 (IOC)
  - OS
  - アーカイブシステム

今回は主に  
計算機関連の話

- 
- はじめに
    - J-PARC加速器とその制御
    - J-PARCの加速器制御システム
  - 制御システムのインフラ
    - ネットワークとセキュリティ
    - ストレージシステム
  - 制御計算機
    - サーバと仮想化
    - 制御端末
    - フロントエンド計算機 (IOC)
  - 制御用ソフトウェア
    - OSとEPICS
  - まとめ

# 加速器制御ネットワーク



## 冗長構成

- コアスイッチ・エッジスイッチが1か所死んでも運用は継続できる
- エッジスイッチはJ-PARC全体で約250台
- MRは12台 + SOHO向けハブ約70台

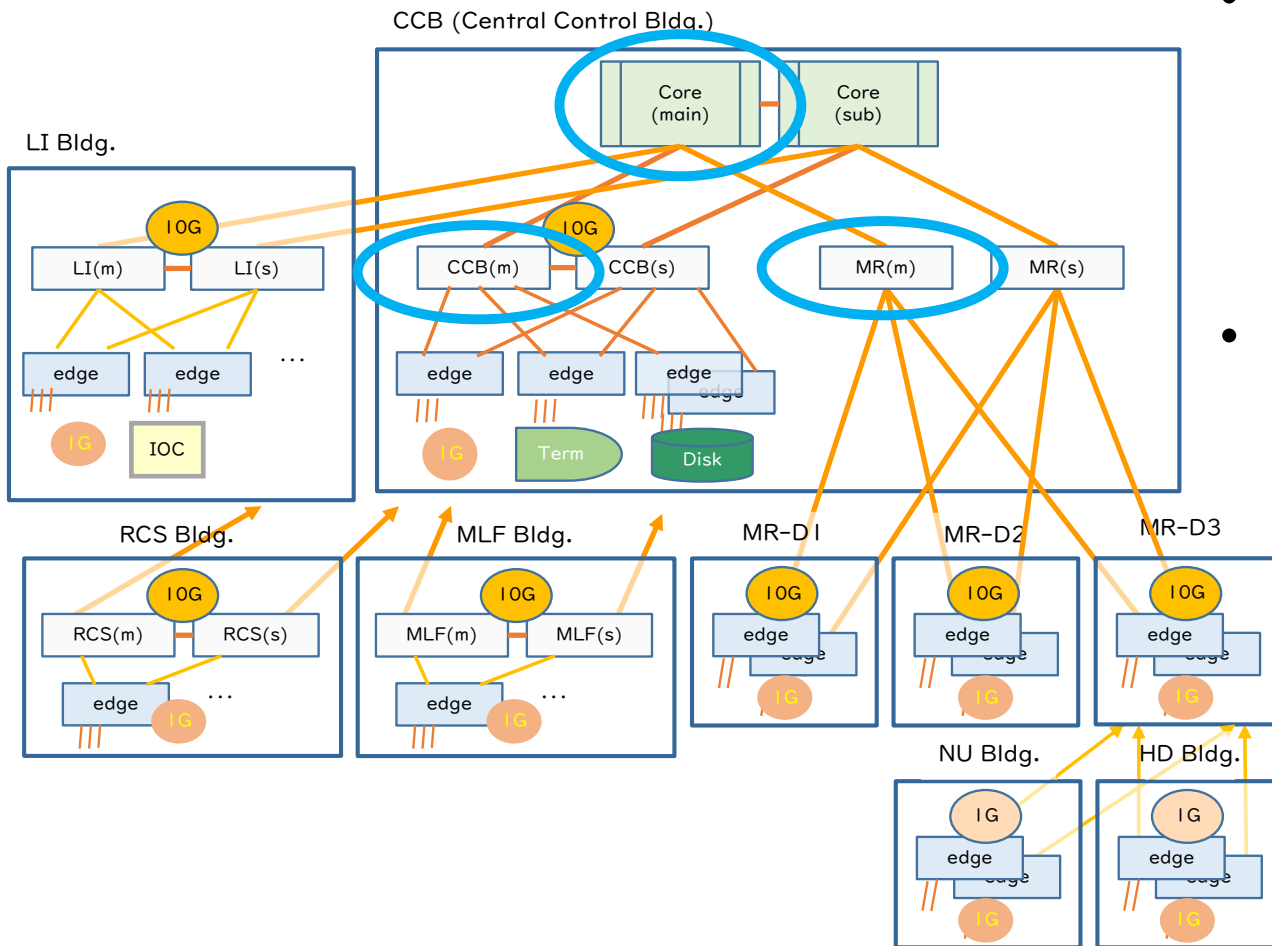
## VLANによるセグメント

- 3 acc, 3 exp. facilities, and CCR

会社による保守  
高いが安心感はある



# 加速器制御ネットワーク



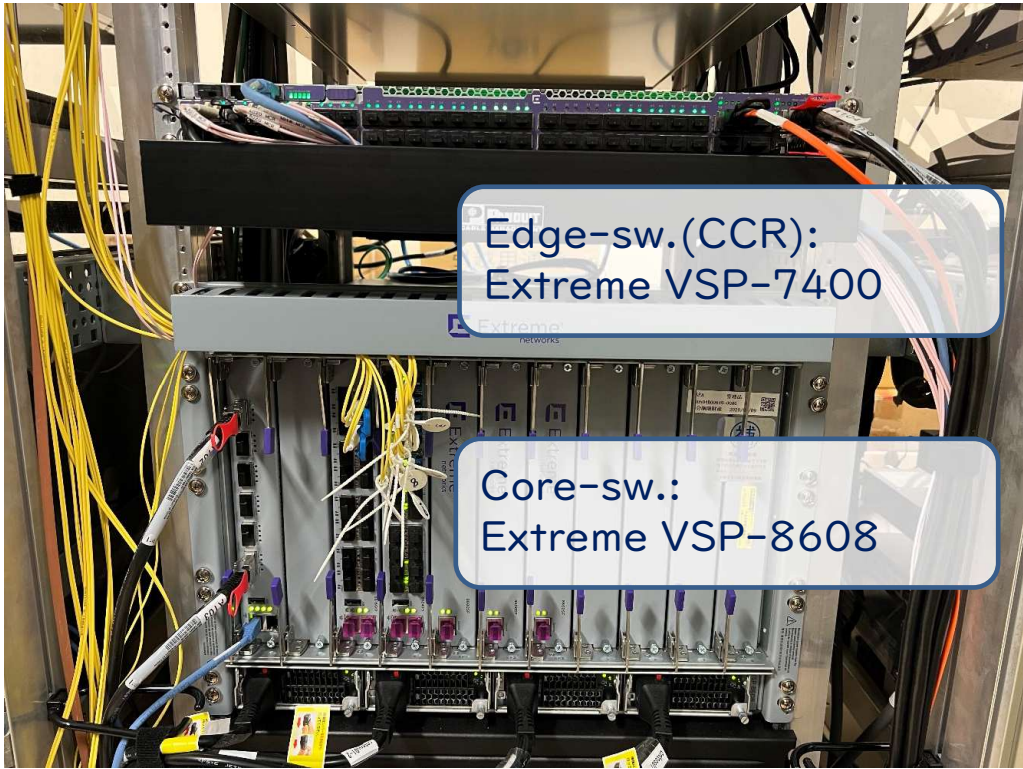
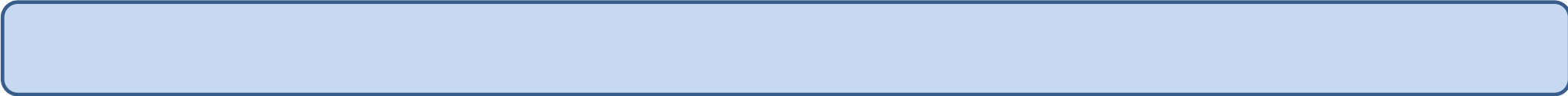
## 冗長構成

- コアスイッチ・エッジスイッチが1か所死んでも運用は継続できる
- エッジスイッチはJ-PARC全体で約250台
- MRは12台 + SOHO向けハブ約70台

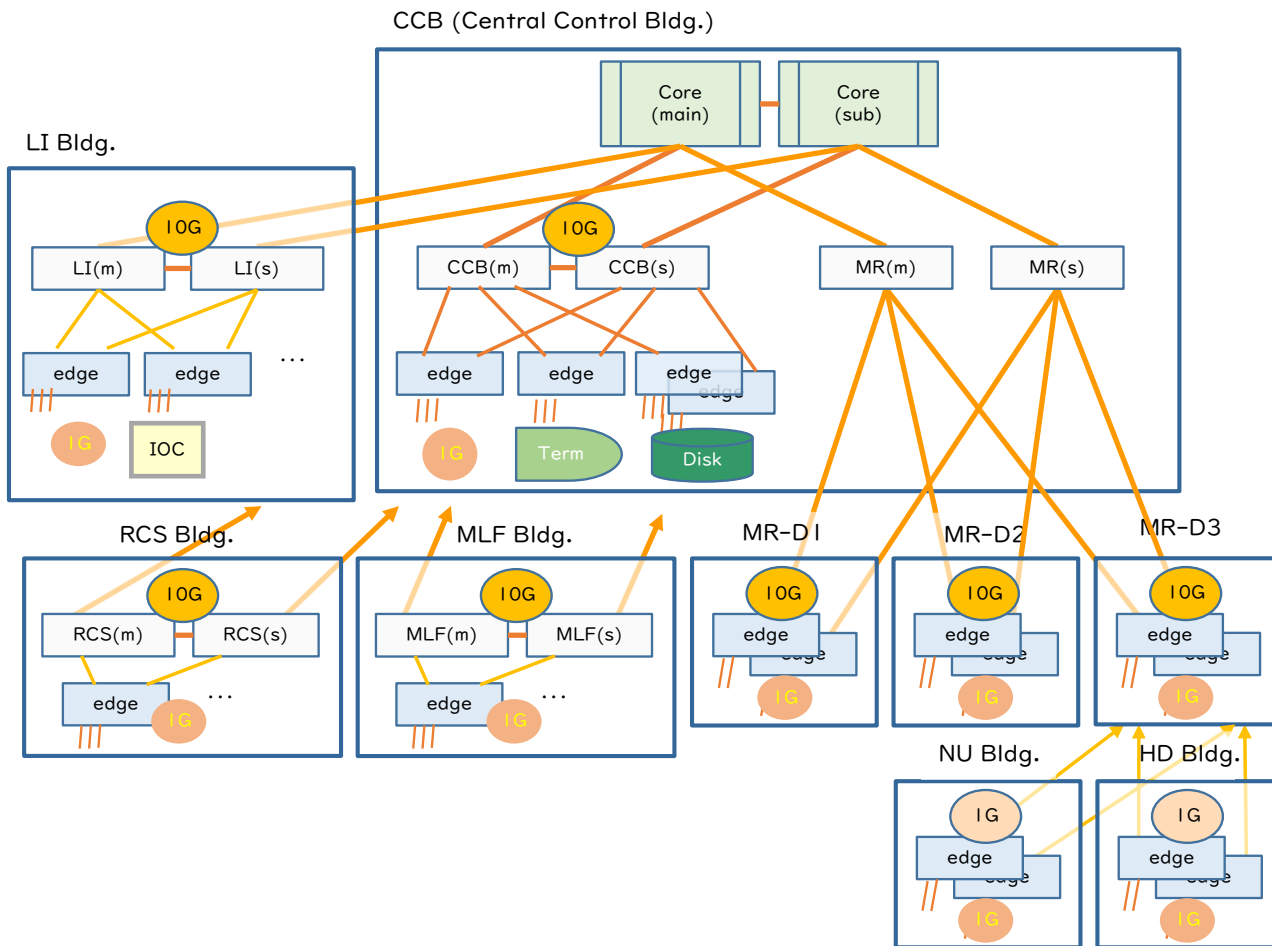
## VLANによるセグメント

- 3 acc, 3 exp. facilities, and CCR

会社による保守  
高いが安心感はある



# 加速器制御ネットワークの歴史



- 7-8年毎に機種更新
- 2005-2007: 建設期
  - Extreme社の機器を導入
  - バックボーン: 10GBps
  - 建屋間: 1Gbps
- 2011-2015: 1回目の機種更新
  - バックボーン: 40GBps
  - 建屋間: 10Gbps
- 2019-2024: 2回目の機種更新
  - バックボーン: 100GBps
  - 建屋間: 10Gbps

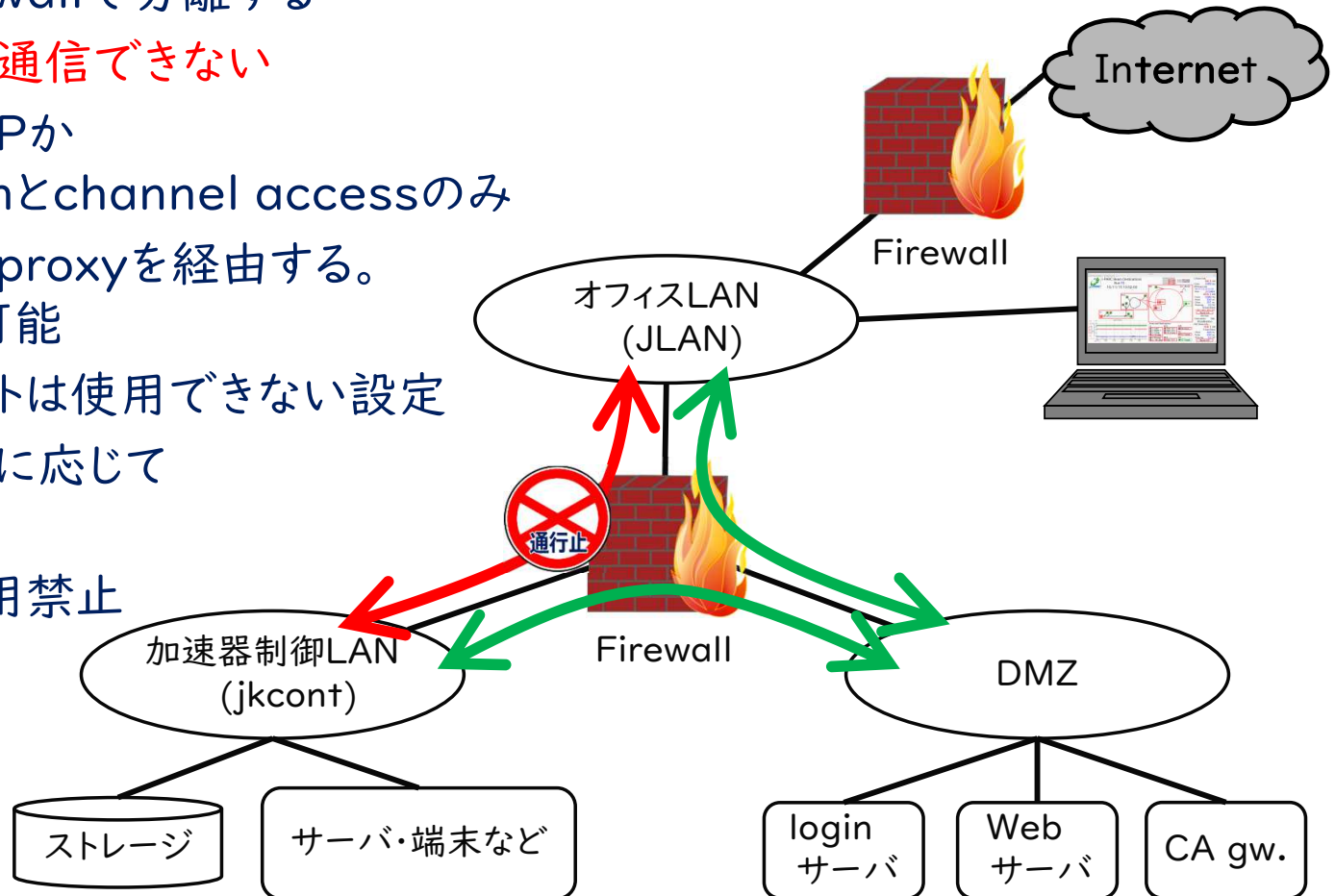
2回目の機種更新が  
現在進行中

# ネットワークとセキュリティ

N. Kamikubota et al.,  
ICALPCS 2017,  
THPHA047,  
pp.1470-1473 (2017)

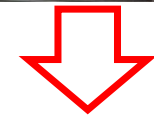
- 制御LANとオフィスLANはFirewallで分離する
- 制御LANとオフィスLANは直接通信できない
- オフィスLANからの接続はHTTPか許可されたIPアドレスからのsshとchannel accessのみ
- 制御LANからの接続はHTTP proxyを経由する。許可されたサイトのみアクセス可能
- 制御LAN内の端末のUSBポートは使用できない設定
- 制御LAN内の計算機は危険度に応じてAntiVirusソフトを導入する
- サポートの切れたOSは原則使用禁止

とにかくマルウェア・ウイルスを  
制御LANに持ち込ませない  
手立てを講じる



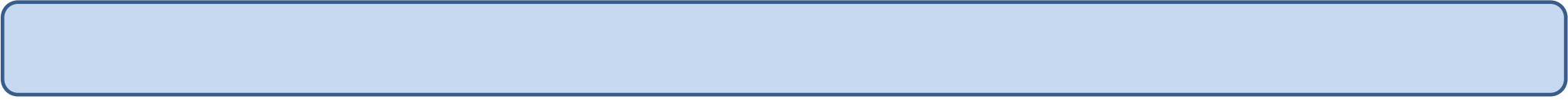
## ストレージ (I)

- 運転データ、運転プログラム、ホームディレクトリ用
  - ファイルサーバのアプライアンス (NetApp製)
    - IBM/Lenovoも中身はNetApp
    - 高いが安心感はある
  - ネットワークもディスクも冗長構成
  - シェルフ・ディスクを追加して拡張できる
  - ~500 NFS clients
- 2008: IBM N3600, 9TB
  - 2012: 28TBに拡張
- 2013: NetApp FAS2240, 48TBに更新
  - 2014: 84TBに拡張
- 2022: Lenovo DM3000H, 192TBに更新



## ストレージ (2)

- 当初はすべてをNetAppで賄うつもりであったが、目的に特化したストレージ (ラックマウントのLinuxサーバ)を導入した
  - あくまでもメインの計算機はbladeサーバ (後述)
- 古いshot-by-shotの波形データ用
  - データは増える一方 (~10TB/yr)だが、過去のデータはあまり参照されない
  - NetAppはオーバースペック
  - 2012: x3630M3 (24TB) x2台
  - 2016: x3630M4 (84TB)
  - 2022: SR550 (216TB)
- EPICSアーカイバ(時系列データ)のストレージ兼CPU用
  - 当初はアーカイブエンジンもbladeで実行していた
  - 2016: x3630M3 (過去データ用をアーカイバ用に転用)
    - CSSアーカイバとArchiverApplianceに1台ずつ
  - 2017: x3650M5 (112TB)
    - ArchiverAppliance

- 
- はじめに
    - J-PARC加速器とその制御
    - J-PARCの加速器制御システム
  - 制御システムのインフラ
    - ネットワークとセキュリティ
    - ストレージシステム
  - 制御計算機
    - サーバと仮想化
    - 制御端末
    - フロントエンド計算機 (IOC)
  - 制御用ソフトウェア
    - OSとEPICS
  - まとめ

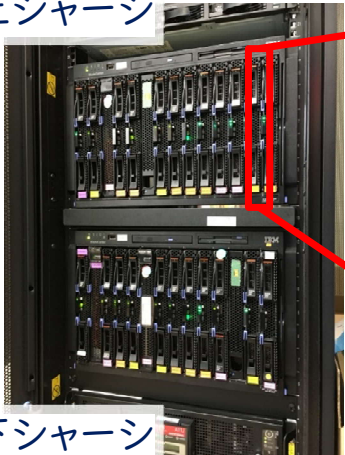
# サーバ計算機 - ハードウェア

- 運転用プログラムの実行用にblade計算機を選択した
  - 典型的なノード:(HS23e, Xeon E5-2470 (16スレッド, 3.1GHz), 48GB)
- 2006: 1シャーシ + blade 5枚で始めた
  - BladeCenterE : 1シャーシに14枚実装できる
- 2007: 2台めのシャーシを導入、随時bladeを増強
- 2020: 最終的な構成はblade22枚
- 2021-2022: ラックマウントサーバ12台に移行した
  - 汎用4台、VMホスト5台、波形データの記録用1台、ログインサーバ、webサーバ

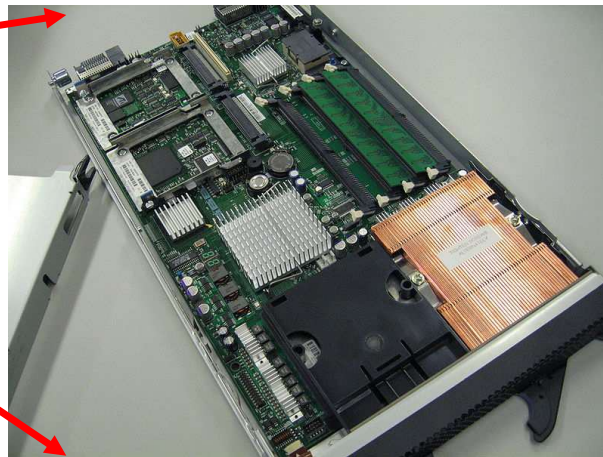
## 基本的な考え方

- 故障したら残りのサーバへ負荷を振り分け、運用を継続する  
(計算機的能力に余裕を持たせておく)
- 後日交換・修理(メーカー保守)

上シャーシ



下シャーシ



## Bladeのメリット

- 機種更新が低コスト
  - HS20~HS23eの4世代7種類を運用した
- 初期導入のコストが高い
  - 価格、スペース、受電容量、...
- シャーシの寿命が全体の寿命を決める



## サーバ計算機運用の歴史

- 2006–2007：各bladeに個別の機能を割り当てていた
  - 制御プログラム実行用、WEBサーバ、データベース、DNS、DHCP、…
  - まだ1枚のbladeの性能が低かった
- 2010：仮想マシンをMR制御に導入した
  - ハードウェアの性能が向上してきた
- 2023：仮想サーバ5台で約100台のVMを運用中
  - IOC用VMホスト: (40スレッド、192GB)×2台 + (20スレッド、96GB)×1台
  - 各種サーバと試験・開発用のVMホスト: (20スレッド、96GB)×2台
  - サーバはプログラム実行用からVMのホストへの転用がすすんでいる

## サーバ計算機と仮想化

- 仮想化を導入したことで
  - 多数の小規模サーバ (dhcp, dns, rdb, IOCなど)を少数のホストに集約できる(省スペース)
  - 耐障害性が向上した (ライブマイグレーション)
  - 柔軟なIOCの運用が可能になった
    - IOCの負荷分散、試験用IOCの立ち上げ
  - 旧バージョンOSのための開発環境の維持が容易になった
- コンテナ技術 (Docker, LXC)も試験・評価がすすんでいる

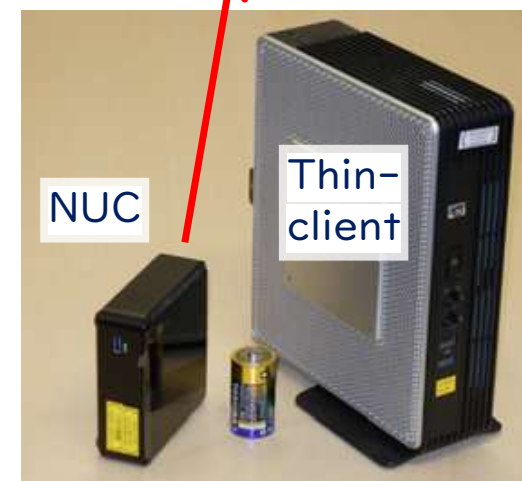
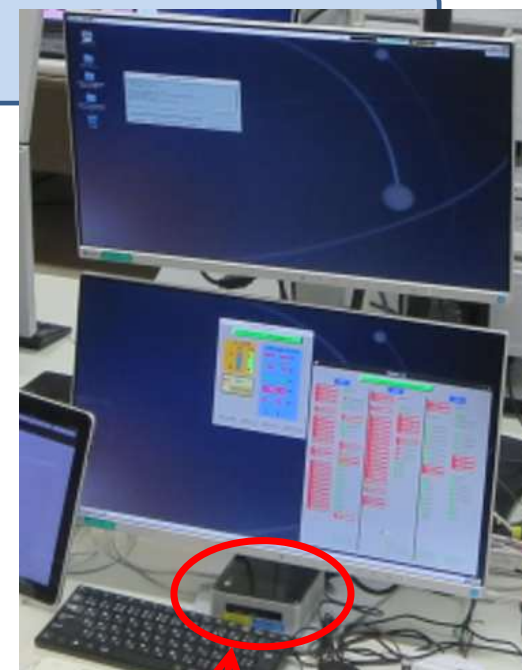
## MRの制御用端末

- 2007: ディスクレスのthin-client (HP/Compaq)
  - X端末的な運用:  
bladeサーバでプログラムを実行し、thin-clientで表示する
    - 例) t5720: Geode NX1500 (1-core, 1GHz), 512MB
  - 2または4ディスプレイ構成で約20台運用した
- 2014: Intel NUCに移行
  - 小型デスクトップPC:  
ローカルにプログラムを実行する
    - 例) Core i5-8259U (8スレッド、3.8GHz), 32GB (2018モデル)
    - 毎年新機種が登場する
  - 2ディスプレイ構成で約45台運用中

bladeサーバに  
余裕ができたので  
KVMホストに転用

### 基本的な考え方

- 故障したら予備機と入れ替える
  - 意外と壊れない。e.g) 2014-2015年購入の30台は、2023までに10台故障した
- 随時新機種をテストして機種更新

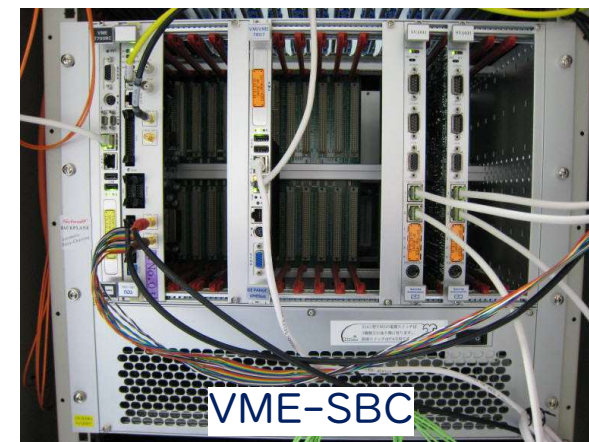


## フロントエンド計算機 (IOC)

- 2007: 当初はVME-SBCをIOCとして選択した
  - ディスクレス・ネットワークブート
  - 例) SVA041: Celeron-M (1-core, 600MHz) 512MB
  - 結果的には制御対象の大半がネットワーク機器で、VMEバスを使わなかった (一部例外を除く; 後述)
- 2008: Linux対応PLC-CPUの導入 (横河)
  - F3RP61: ppc603e (1-core, 600MHz), 128MB
- 2010: 仮想IOCの導入
- 2014: 小型ファンレスサーバの導入 (PiNON)
  - サバ太郎Type-P: Celeron J1900 (4コア 2GHz) 8GB

### 基本的な考え方

- 故障したら予備機と入れ替える
- 意外と壊れない。これまでに~100台購入し、故障したのは3台
  - あまりにもトラブルが少なくて苦勞話を披露できない



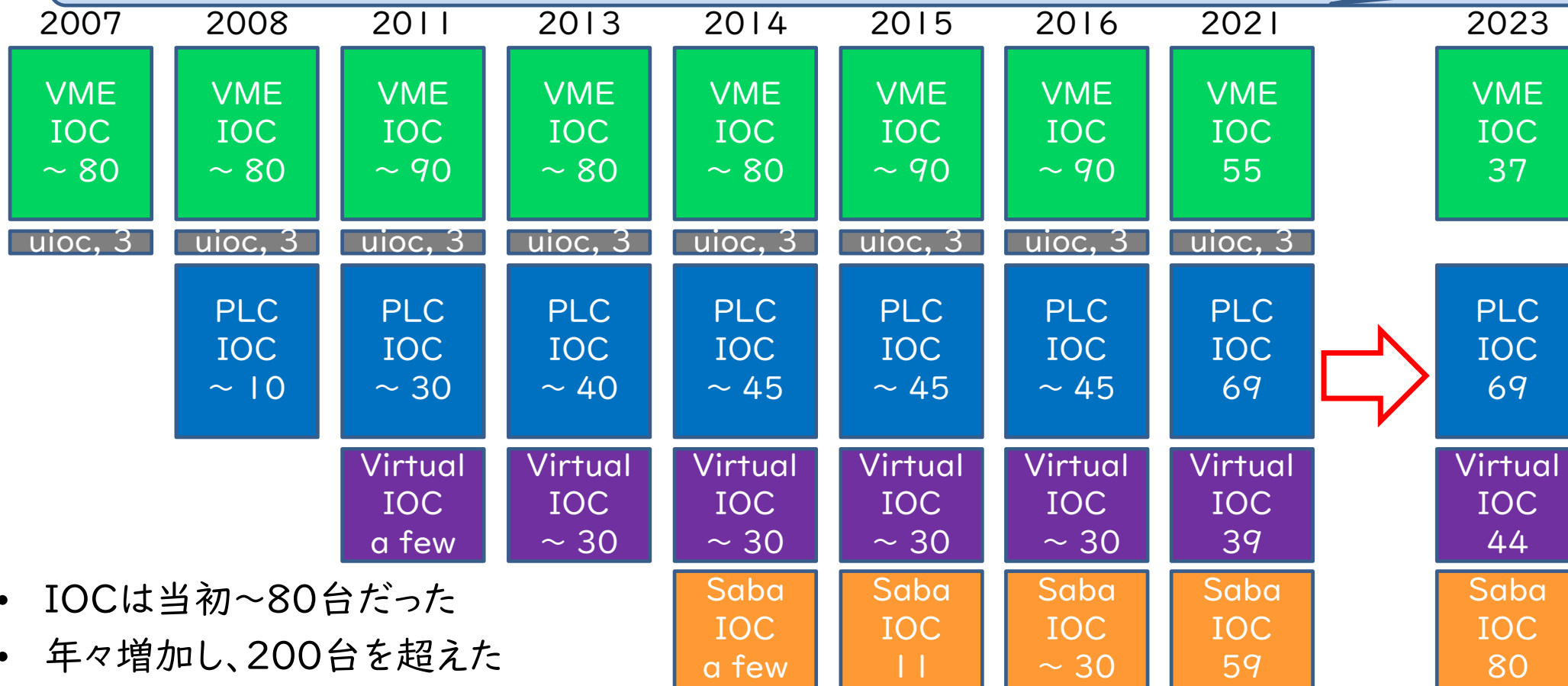
サバ太郎



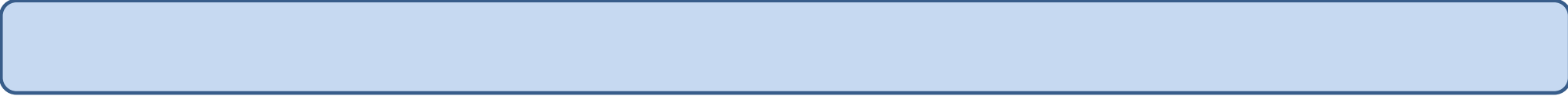
F3RP61

# MR運転用IOCの種別と変遷

N. Kamikubota et al.,  
PASJ 2020,  
WEOOP04,

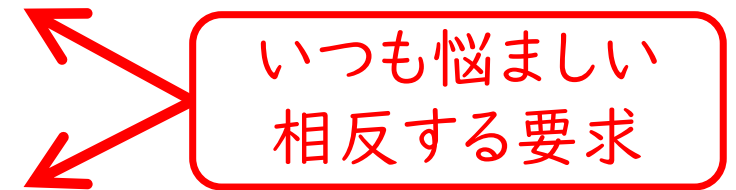


- IOCは当初~80台だった
- 年々増加し、200台を超えた
- VME-SBCからサバ太郎へ移行した
- これらの他に試験用Virtual IOCが~15台

- 
- はじめに
    - J-PARC加速器とその制御
    - J-PARCの加速器制御システム
  - 制御システムのインフラ
    - ネットワークとセキュリティ
    - ストレージシステム
  - 制御計算機
    - サーバと仮想化
    - 制御端末
    - フロントエンド計算機 (IOC)
  - 制御用ソフトウェア
    - OSとEPICS
  - まとめ

## OSとEPICSの基本的な考え方

- サーバ・制御端末・IOCは同じバージョンのOS・EPICS baseを使う
- そのバージョンのOS・EPICSを使い続けたい (加速器運転の観点)
- 頻繁にOS・EPICSを更新したい (セキュリティ上の観点)



いつも悩ましい  
相反する要求

## OSとEPICSの実際

- OSには毎年2-3回パッチをあてる
  - ソフトウェアは可能な限りrpmパッケージをインストールする
  - 夏季、年末年始、年度末等の加速器長期メンテナンスの際
    - サーバを再起動するのは夏季だけ
- 深刻な問題があったらEPICS baseもパッチレベルを上げる
  - 例: R3.15.5 → 3.15.8 (CASモジュールでCPUが暴走する問題があった)
- EPICSのデバイスサポートは必要に応じて更新する
  - IOCによって違うバージョンのデバイスサポートが混在するが、目をつむる
  - 特に内製のデバイスサポートの場合
    - 例) 横河PLCのデバイスサポート (netDev, f3rp61)
- いずれも予備の計算機・制御対象機器で試験・開発し、その後運転環境に投入する
  - 予備機を用意しておけば、機器故障時に加速器の停止期間をみじかくすることもできる



## …といくつかの例外

- タイミング受信機用のVME-IOC
  - VME-7700RC (Celeron 400MHz, 512MB)
  - いまだにScientific Linux 4 + EPICS R3.14.7で稼働中
  - 23台をPLCベースのタイミング受信機に移行する予定
    - のこる2台もなんとかしたい
- BLM用のVME-IOC
  - V7865 (CoreDuo 2GHz, 3GB)
  - Scientific Linux 6 + EPICS R3.14.12で稼働中
  - 全12台をSVA-061 (Atom E3950 1.6GHz 4コア, 8GB) + CentOS7 + EPICS R3.15.8に移行予定
- YokogawaのLinux対応PLC-CPU
  - F3RP61 (PPC603e 600MHz, 128MB) + ELDK 4.1
  - F3RP71 (Coretex A9 866MHz 2コア, 1GB) + Yocto 1.6.1への移行が進行中

いずれもCPUの都合で  
OS(とEPICS BASE)が  
決まっている

# MRの加速器制御ソフトの変遷

- 2008 : Scientific Linux 4 + EPICS R3.14.7で始めた
  - GUI: MEDM/EDM/Strip-tool
  - EPICS Channel Archiver
  - 2012/02 : SL4 EoL
- 2012-2014 : Scientific Linux 6 + EPICS R3.14.12.3
  - 32-bit から 64-bit への移行も
  - 2012: blade をSL6に, 2013:IOCをSL6に
  - 2014: NUCの導入で端末もSL6に、CS-Studioも導入可能になった
  - 2017 : Archiver Applianceに移行
  - 2020/11 : SL6 EoL
- 2019-2020 : CentOS 7 + EPICS R3.15.5
  - 2020 : EPICS R3.15.8に更新
  - 2024/06 : CentOS7 EoL
- 2023-2024 : Alma Linux 9 + EPICS R7.0.7へ移行開始
  - EPICS7のEPICS7たるpvAccessはまだ使っていない

ソフトウェア(OS・CSS)の都合で  
端末をNUCに移行した

2019購入のNUCの都合で  
CentOS8を待たずにCentOS7に

CentOS8は2022/12にEoLを  
迎えた。結果的にCentOS7は正しい  
選択だった

EPICSは導入の時点で  
最新の安定板を選択

## まとめ

- J-PARC MRの加速器制御システムの運用を開始した2008年から15年間で、その構成要素は2~3回の世代更新を経た
  - 拡張・増強：ネットワークとディスク
  - 利用形態の変化：サーバ計算機
  - ハードウェアの変化：端末、IOC
- 制御システム全体を維持するには継続的なR&Dが必要不可欠である
  - コンポーネントの寿命は長いものでも10年、短いものは2-3年で、年々短くなる傾向にある
  - ソフトウェアと計算機は足並みをそろえて更新する必要がある
  - Linux + EPICSベースなところは今後も変わらないであろう
- 予備機で新OS・ソフトを試験し、運転用の環境やサーバ計算機に展開するアプローチは今のところうまく機能している