

HPCインフラ技術 + IoT

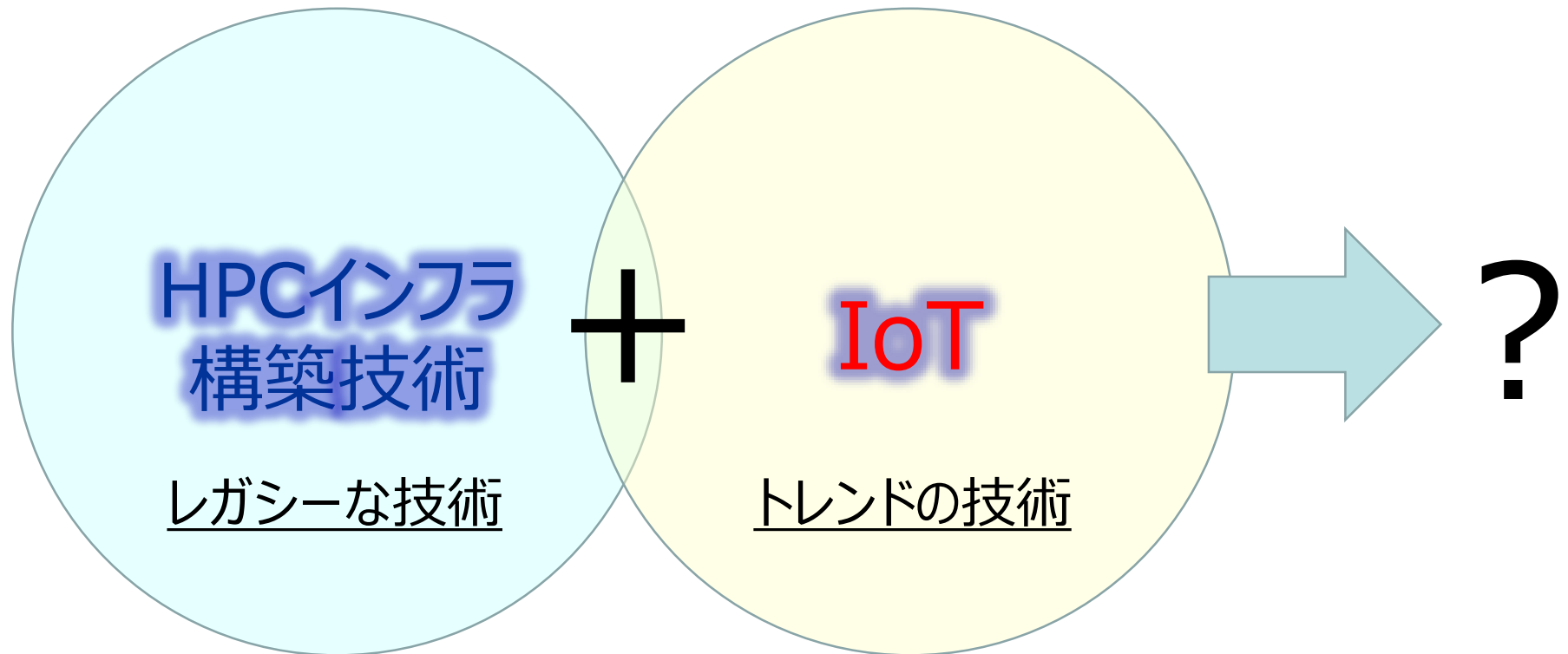
2020年 1月 22日

サーバ技術部第2課

1. 本発表の検証テーマ
2. 検証環境
3. HPCインフラの構築
4. Raspberry Pi での処理検証
5. まとめ

1. 本発表の検証テーマ

サーバ技術部第二課が得意とするHPCインフラ構築技術を
IoT分野にどう生かせるかやってみようをテーマに検証



HPC (ハイ・パフォーマンス・コンピューティング)



たくさんのコンピュータによる並列処理で大規模な計算を行う
システム（大規模なものでは1000台以上）

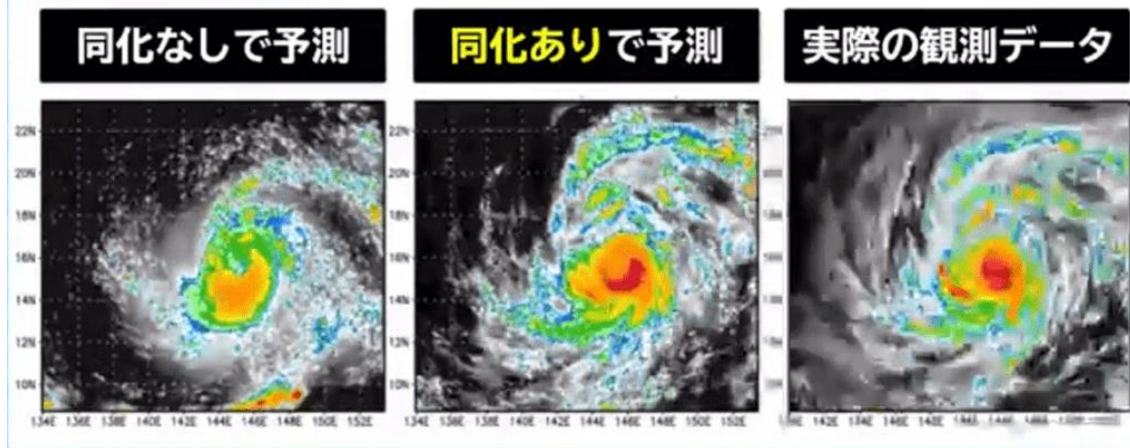
(補足) HPCはどんなことに使われているか

(創薬)

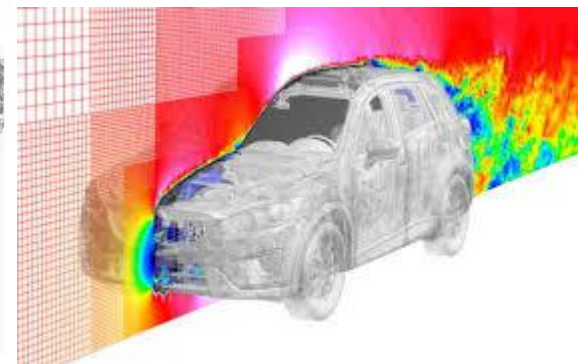
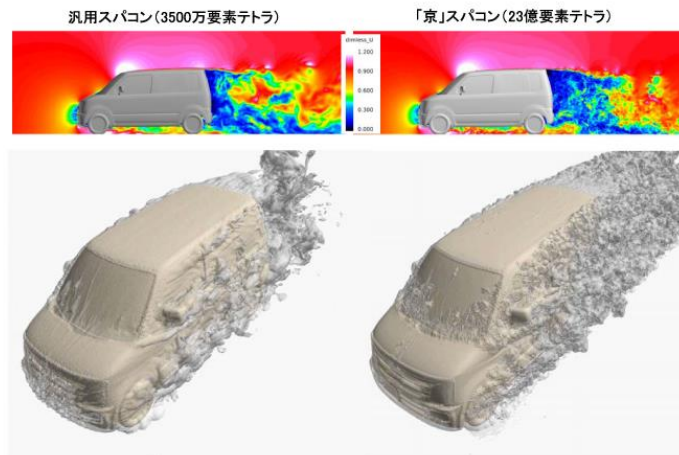


(気象予測)

台風の発達予測



(車の開発)



(補足) HPCインフラを構成する技術

たくさんのコンピュータを構築、管理する技術

① 大量のサーバへのOSの配信

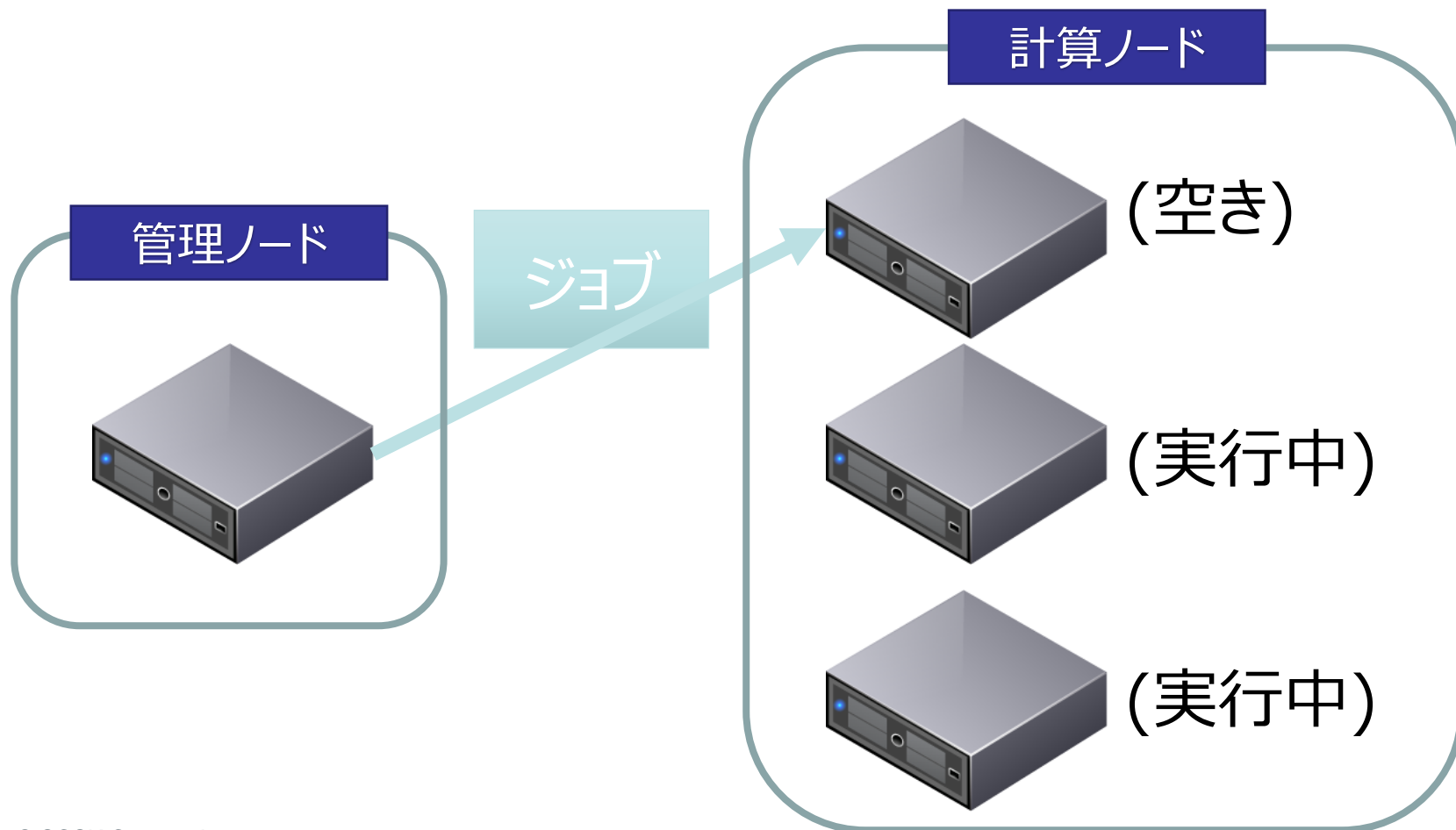
② 監視

③ 処理（ジョブ）の管理

④ 並列演算ライブラリ(MPI等)

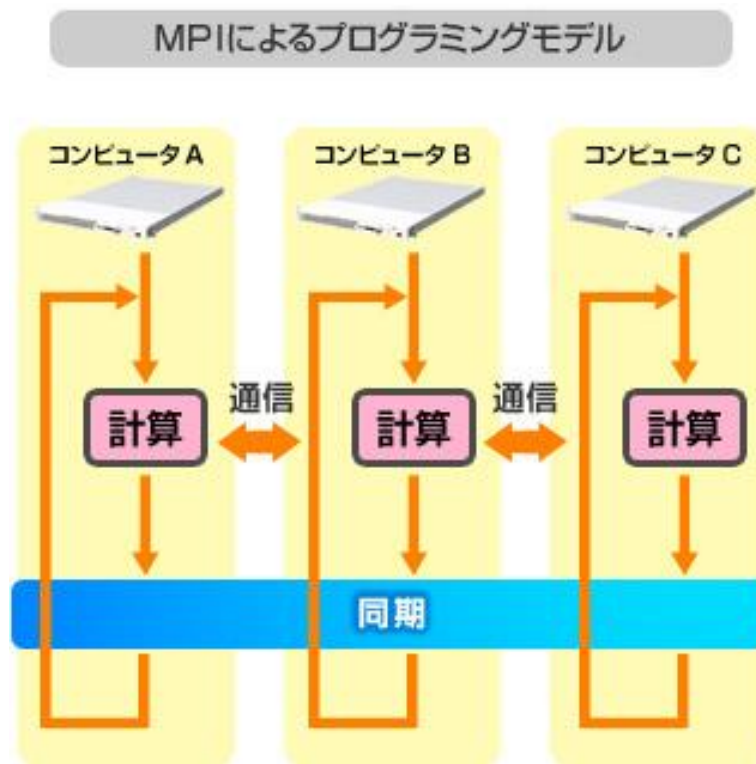
(補足) ジョブ管理とは

- 管理ノードが各計算ノードのジョブ実行状況を管理
- 各計算ノードの空き状況に応じて、ジョブを実行させる

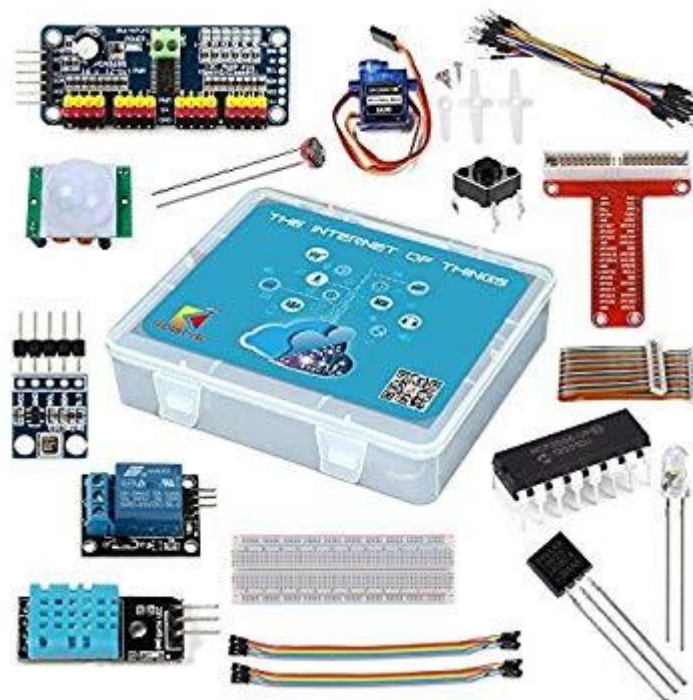


(補足) MPIとは

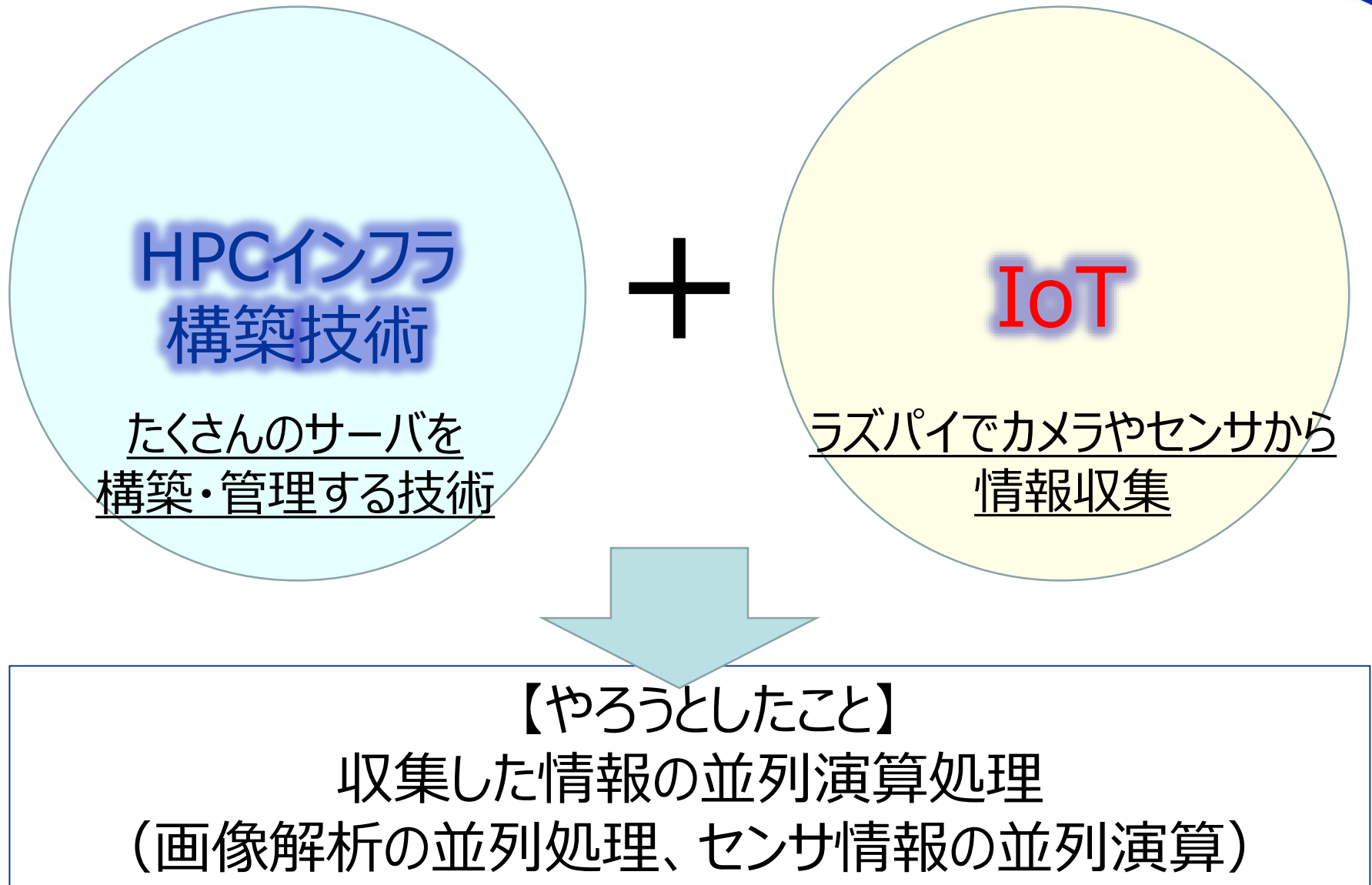
- Message Passing Interface (メッセージ パッシング インターフェース、MPI) とは、並列コンピューティング利用するための標準化された規格です。



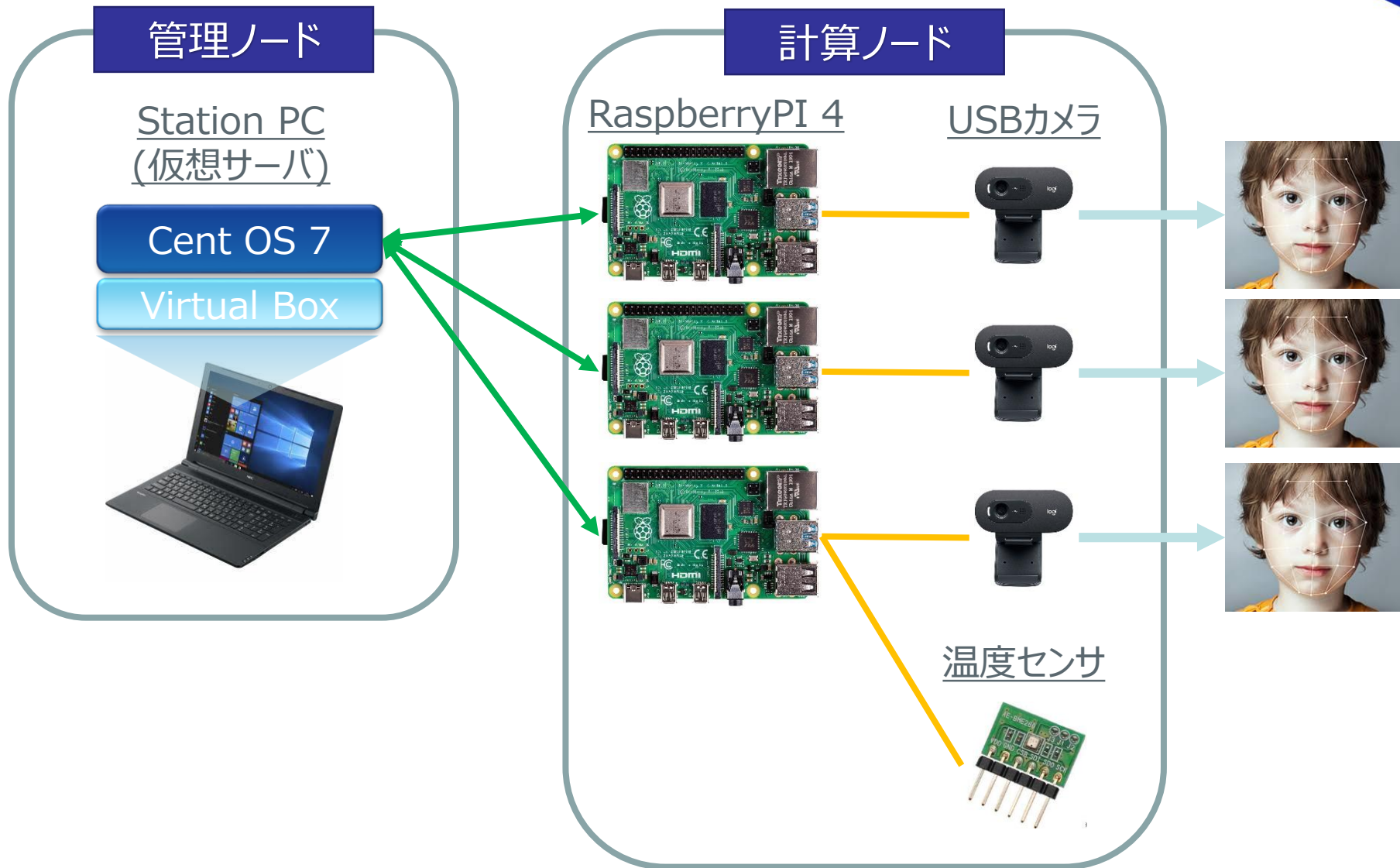
- HPC環境として汎用OS(Linux)の利用が必要
- 簡単に様々なセンサ等と接続できること



1. 本発表の検証テーマ



2. 検証環境



2. 検証環境 – 管理ノード



モデル	NEC VersaPro VKT16X-4 (Station PC)
CPU	Intel Core i-5-8250U 1.60GHz(64bit)
Memory	8GB
Disk	500GB HDD SATA 7200rpm
OS	Windows 10 Pro
仮想化ハイパーバイザ	Oracle Virtual Box
仮想OS	CentOS 7.7
アプリケーション	Slurm 19.05.4 (コントロールサーバ)
	MySQL 8.0.18
	Apache 2.4.6
	PHP 2.4.0
	Zabbix Server 4.0-2
	Python 3.6.8

2. 検証環境 – 計算ノード



モデル	Raspberry PI 4 Model B
CPU	クアッドコア 1.5GHz ARMv8 (64bit)
Memory	4GB
Disk	Micro SDカード(32GBまたは64GB)
OS	Ubuntu 19.10
アプリケーション	Slurm 19.05.4 (エージェント)
	Zabbix Agent 4.0.4
	OpenCV 4.1.2
	Python 3.7.5

2. 検証環境 – カメラ、温度センサ



モデル	Logitech C270n
最大fps	30fps
最大解像度	720p
インターフェース	USB



モデル	AE-BME280
測定温度	-40℃～85℃
測定湿度	0～100%
インターフェース	GPIO

3. HPCインフラの構築 ①OSの配信

「① OS配信」の為のインフラ構築

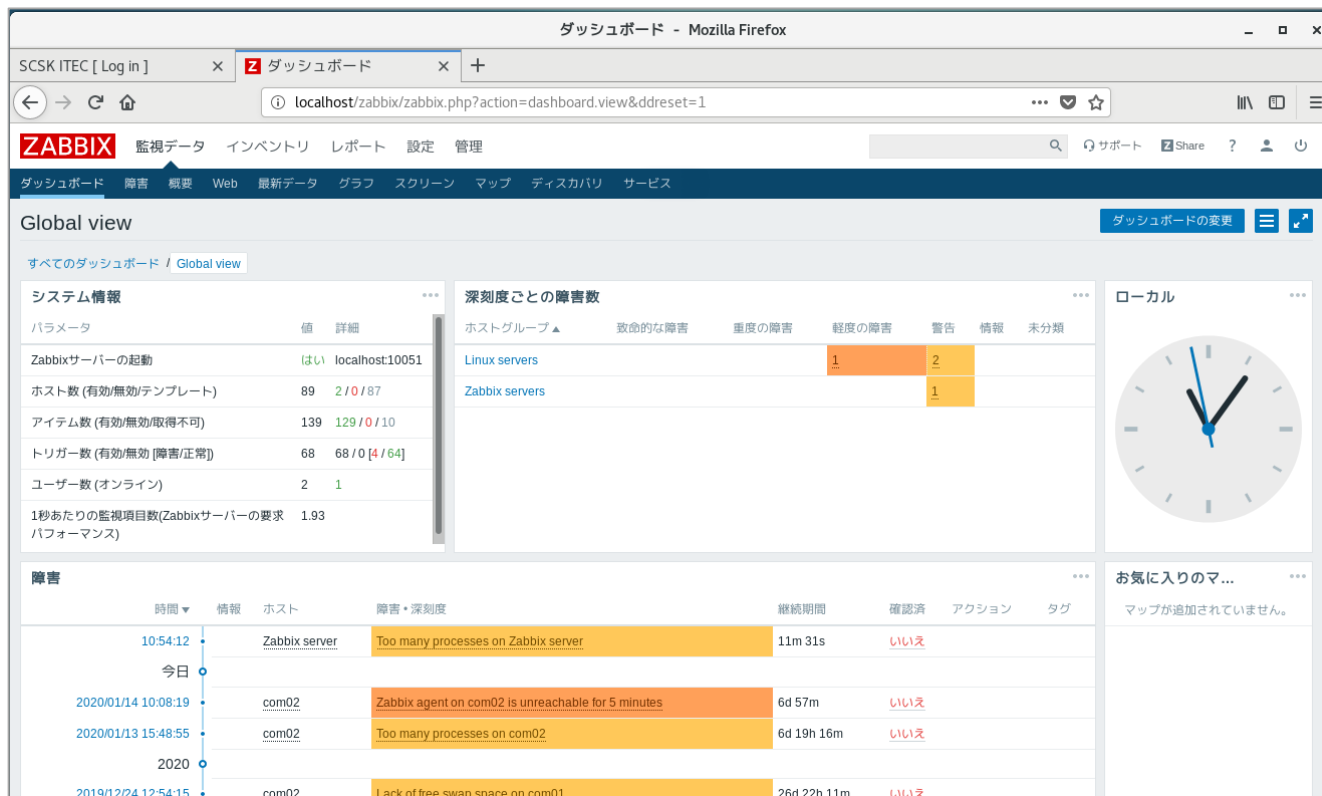
⇒ ラズパイのCPUに対応した OS/アプリ が無く、実施出来ず。

Ubuntu	・ OS配信ソフト（Warewulf）が対応していない。 （依存するライブラリ等が対応していない）
CentOS(32bit版)	・ HPC用ソフト（OpenHPC）がCPU(Armv7)に対応していない
OpenSUSE	・ OSインストール後に操作不可 （マウスやキーボード等が認識されない、DHCP動作しない）
Manjaro	・ OS配信ソフト（Warewulf）が対応していない。 （依存するライブラリ等が対応していない）
CentOS(64bit版)	※ 今回時間の関係上、検証未実施だが、 本OSでOS配信可能となる可能性あり。

3. HPCインフラの構築 ②監視

「② 監視」の為のインフラ構築

⇒ 監視用ソフトウェアとして Zabbix を導入



「③ ジョブ管理」の為のインフラ構築

⇒ Slurm を導入

PBS Professional

【インストール出来ず】

- Ubuntu + aarch64に対応したパッケージ無し
- ソースからのコンパイル不可
(Configure内にx86_64としてハードコードされている)

Slurm

- ソースからのコンパイルによりインストール完了

3. HPCインフラの構築 ④MPIの導入

「④並列演算ライブラリ」の為のインフラ構築

⇒ MPIを導入

⇒ Linpack (ベンチマークソフト) を実施

```
=====
||Ax-b||_oo/(eps*(||A||_oo*||x||_oo+||b||_oo)*N)=      0.0024202 ..... PASSED
=====
T/V          N    NB    P    Q          Time          Gflops
-----
WR00R2C4      2000   32    2    2          3.67          1.458e+00
HPL_pdgesv() start time Mon Aug 13 16:21:12 2018
HPL_pdgesv() end time   Mon Aug 13 16:21:16 2018

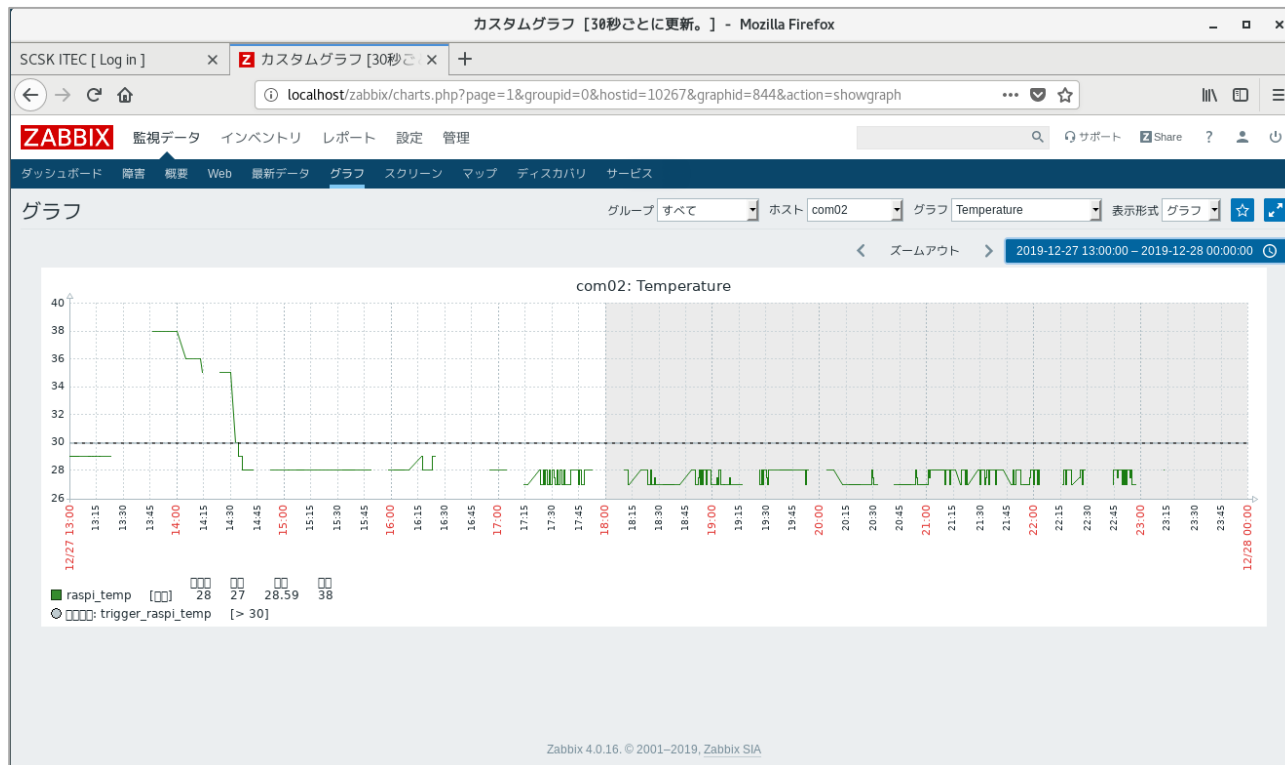
=====
||Ax-b||_oo/(eps*(||A||_oo*||x||_oo+||b||_oo)*N)=      0.0027086 ..... PASSED
=====
T/V          N    NB    P    Q          Time          Gflops
-----
WR00R2R2      2000   32    2    2          3.64          1.469e+00
HPL_pdgesv() start time Mon Aug 13 16:21:16 2018
HPL_pdgesv() end time   Mon Aug 13 16:21:20 2018

=====
||Ax-b||_oo/(eps*(||A||_oo*||x||_oo+||b||_oo)*N)=      0.0024202 ..... PASSED
|
0 bash 1 bash 2 ttyUSB0 3 bash 4 bash 5 bash 6 bash 7 bash 18/08/14 1
```


4. Raspberry Pi での処理検証

『温度センサからの情報取得』

⇒ センサで取得した情報を Zabbix で管理サーバに収集



4. Raspberry Pi での処理検証

『カメラからの画像解析』

⇒ Raspberry Pi にUSB接続したカメラで画像取得

⇒ 単一ノード内で画像処理（顔検出・認証）を実施

4. Raspberry Pi での処理検証



※ 顔認証の詳細は、この後のサーバ技術部第一課(ミルクボール)の発表に続きます。

■ HPC技術のRaspberry Piへの適用

➤ ① OS配信のソフトウェアの導入は実施出来なかった

⇒ 今後各OSやソフトウェアがRaspberry Pi のハードに対応していくことにより導入は可能と考える。

➤ ②Zabbixによる監視、③ジョブの実行、管理は問題なく実施できた

⇒ Raspberry Pi上でも変わらずにHPC技術を適用していけるものと思われる。

■ センサ情報の収集、画像解析

⇒ 並列処理では無いが、単一ノード内では問題なく実施できた。

■ 今回の環境(ラズパイ+HPC)は従来のIoTと比較して

メリット

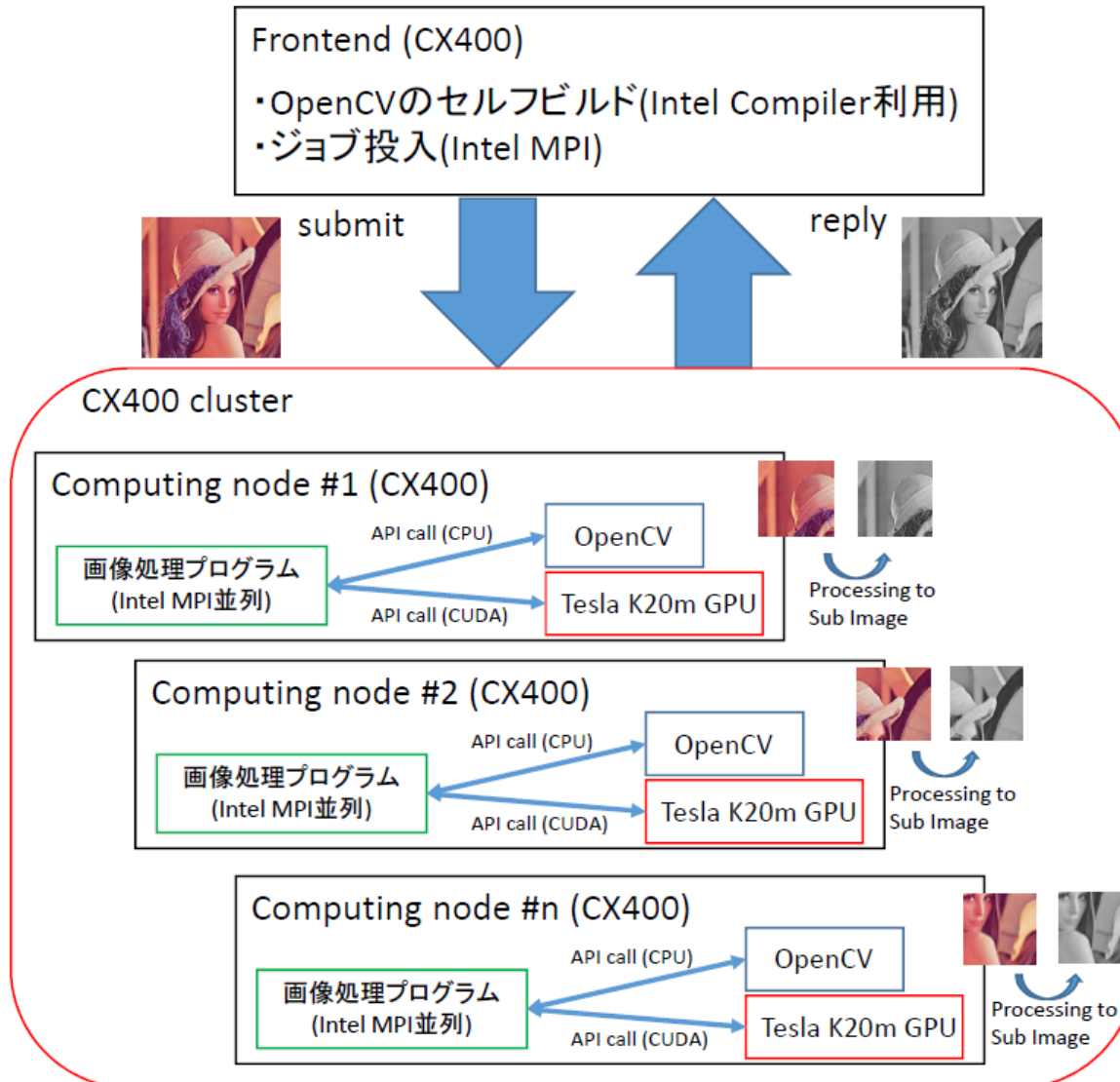
汎用的なOSが利用できる	組み込みシステムのスキルが無くても開発可能
	処理の自由度が高い(一般的なAPIが利用可能)
HPCの利点を享受できる	制御用のボードコンピュータを増やせば増やすほど総合的な計算能力が増大する。
	計算用に別途専用のコンピュータを用意する必要がない。
	ステートレスなシステムを構築可能であり、システム拡張が容易

デメリット

制御用のボードコンピュータ	比較的高価であること
	消費電力が比較的高いため、設置環境を選ぶ
汎用OS利用でのデメリット	リアルタイム性が低い ⇒ kernel拡張やリアルタイム用プロセッサを搭載する事である程度 解決可能だがコスト増大や専用スキルが必要になる

5. 今後の検証について(OpenCV + MPI)

【並列画像処理】



5. まとめ

■ 今後の検証について

- MPI + OpenCV による並列処理を実装していくなど、より実用的なシステムを目指していきたい。

