

2024年5月29日

次世代計算基盤に係る調査研究「新計算原理チーム報告会」

量子コンピュータとスパコンの連携

佐藤 三久

理化学研究所 量子HPC連携プラットフォーム部門 部門長

順天堂大学 健康データサイエンス学部 特任教授

辻 美和子

理化学研究所 量子HPC連携プラットフォーム部門 ユニットリーダー



JHPC-quantum

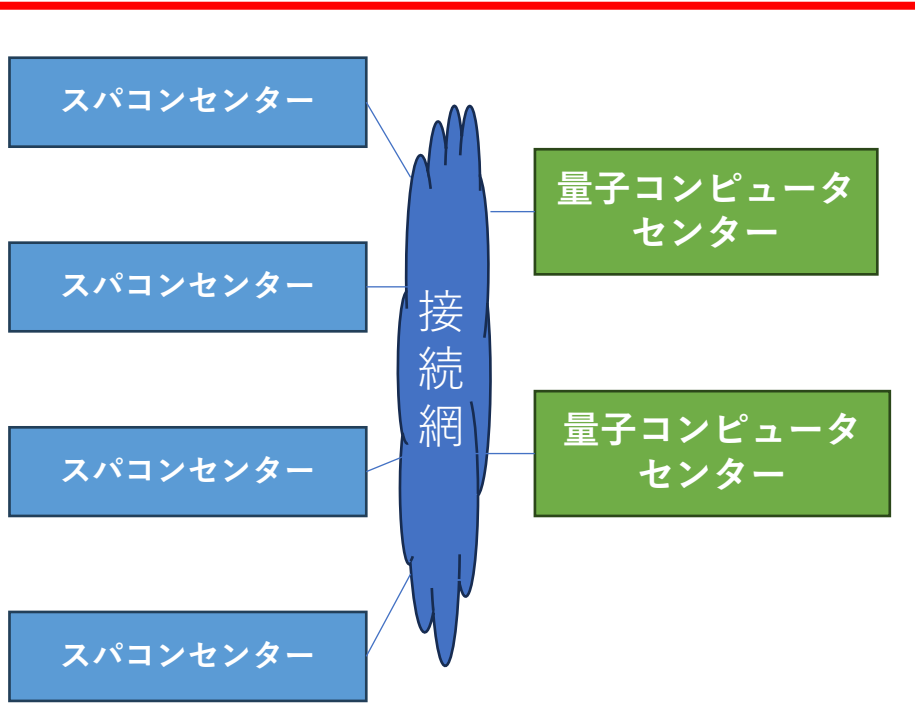


もくじ

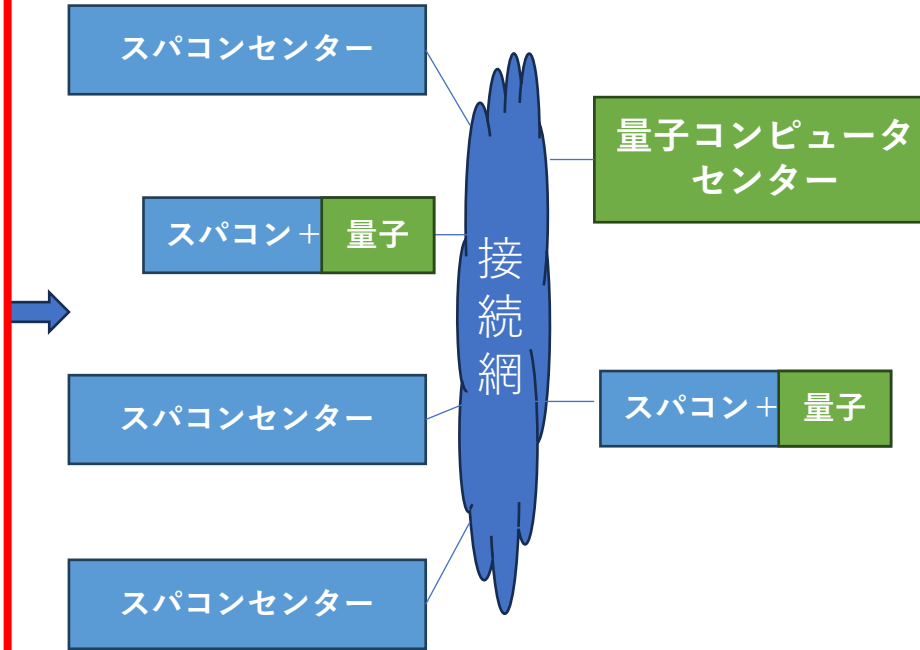
- 量子コンピュータとスパコン(HPC)との組み合わせ形態
- 量子コンピュータの利用のAPI・認証とプログラミング環境
- 量子HPC連携プログラミングモデルとスケジューリング
- ケーススタディ:
 - 量子HPCハイブリッドプログラムのための遠隔手続き呼び出し(RPC)の試作
 - **NEDO JHPC-quantum** プロジェクト
- 量子コンピューティングの現状と量子HPC連携の展望

将来の国の計算基盤（スーパーコンピュータ、量子コンピュータを
量子コンピュータを計算基盤の一つとして位置づけ、申請、認証、ジョブ管理、APIの方法を統一する

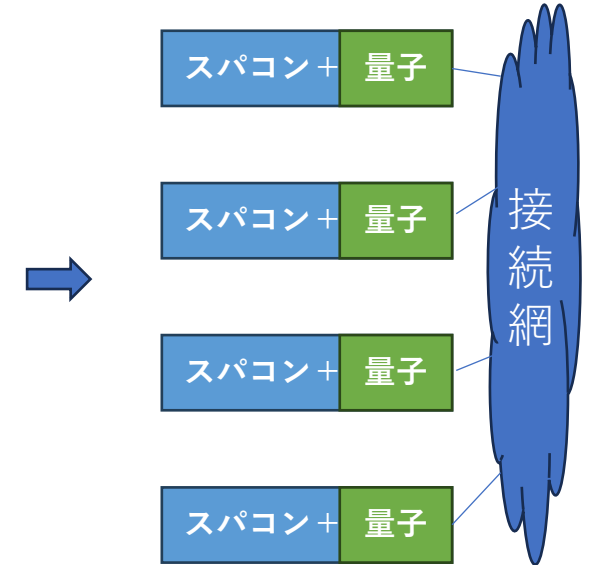
天野先生のスライドより



a) 独立したセンターを接続
スパコンはスパコン、量子は量子
申請や認証を統一、ジョブ管理、
APIを統一



b) 様々な構成のセンターを接続



c) 統合センター
全ての（多くの）スパコン
センターは計算リソースとして
量子コンピュータを装備

量子インターネットの利用はさらに未来の話
→ 複数量子コンピュータの接続に限られる

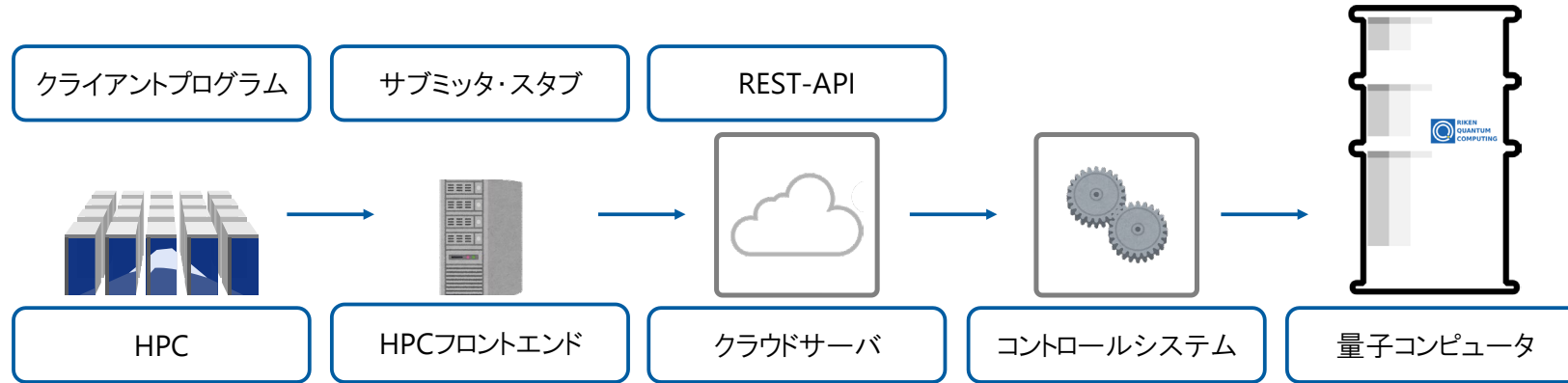
スパコン＋量子融合クラウドの利点

天野先生のスライドより

- 今のままではなぜ良くないか？
 - 量子コンピュータの研究が制約される
 - 量子コンピュータクラウドがなし崩しに実現されていく
- スパコン方式の申請、アカウント管理
 - 量子コンピュータの商用クラウド利用は超混雑が予想されるので資金がモノを言うようになる（混雑は今でも結構ひどい）
 - 広く研究を行うためには現在のスパコン方式は優れている
- スパコンにログインして用いることで、ユーザ環境の整備や量子コンピュータの柔軟な利用が可能
 - 実はクラウドでもdocker環境でAIプラットフォームと連携できたりして割と進んでいる。しかし、スパコンログイン方式の方がより柔軟にできる可能性がある
- 大規模高速なシミュレーションが可能。実機と簡単に入れ替えることができるのが魅力
 - 現在、クラウドでも量子コンピュータシミュレータは使えるが、脆弱
 - 例) AWS 状態ベクトル型 34qubitまで、テンソルネットワーク型50qubitまで
 - 富岳ならば状態ベクトル型 40qubit (46qubit)、テンソルネットワーク型100以上は可能
 - 例) スパコン不老を使っている量子化学計算の専門家杉崎さん (SQAI)のインタビュー (6・30)
- スパコンを既に利用している物理学者、量子化学、天文学等の研究者が自然に量子コンピュータに研究を展開できる
- 本当に量子コンピュータが計算資源として役に立つようになればスパコンセンターに入れるのは自然
- 実験的な量子コンピュータ（ムーンショットG6の成果など）を簡単にクラウドに導入可能
 - 量子コンピュータの研究者にはクラウドへの投入が困難で壁になっている
 - 統一したジョブ管理で、複数量子コンピュータ間の連携も可能になる

量子コンピュータの利用のAPI・認証とプログラミング環境

- 量子コンピュータ利用のためのAPI
 - クラウドサーバ等の上のREST-APIが主流
 - HPC側のクライアントプログラムから適切なライブラリを用いてREST-APIを呼び出すことが可能



- 量子コンピュータ利用のための認証
 - トークン形式(主流)、Virtual Private Cloud、クラウド連携認証
 - トークン形式でサービス提供: IBM, IQM, IonQ, Rigetti, AQT-cirq, Pasqal 等
 - 記述言語(アプリケーション開発者向け)
 - Python (Qiskit, qulacs, tiket 等), OpenQasm
 - 記述言語(REST-APIに与えられるもの)
 - 中間言語: OpenQasm and/or QIR
 - ベンダ独自の形式 (例: IBMのqpy, IQM の IQM transfer format)
- 多くの量子コンピュータで中間言語とベンダ独自形式の両方が利用可能

- 量子プログラムの記述言語はpythonのフレームワークが多いが、次第にC++/Rustのフレームワークに移りつつある。これにより、**スパコンで計算科学アプリと組み合わせることが容易になる。**
- GUIによる方法もあるが、...
- 将来的にはさらにハイレベルの記述が必要

```
from qulacs import QuantumState
qc = QuantumCircuit(5)
qc.add_H_gate(0)
for i in range(4):
    qc.add_CNOT_gate(i, i + 1)

state = QuantumState(5)
qc.update_quantum_state(state)
print(state)
```

qulacsの例

```
from qiskit import QuantumCircuit
qc = QuantumCircuit(5, 5, name="ghz")
qc.h(0)
for i in range(4):
    qc.cx(i, i + 1)
qc.barrier()
for i in range(5):
    qc.measure(i, i)
```

Qiskitの例

```
sim_backend = BasicSimulator()
job = sim_backend.run( \
    transpile(qc, sim_backend), shots=1024)
result = job.result()
print("Basic simulator : ")
print(result.get_counts(qc))
```


- 記述言語から生成され、コントローラで実行される
- OpenQASM や QIRがある。
- トランスパイル(transpile) – 中間コード上で量子デバイスの制限を満たすコードに変換・最適化
- コンパイル：中間コードでから、コントローラ上の実行コードに変換

```
OPENQASM 3;
```

```
qubit[3] q;
```

```
bit[3] c;
```

```
h q[0];
```

```
cnot q[0], q[1];
```

```
cnot q[1], q[2];
```

```
c = measure q;
```

```
OPENQASM 3;
```

```
qubit[4] q;
```

```
bit b;
```

```
angle[4] theta = 0;
```

```
reset q;
```

```
h q;
```

```
barrier q;
```

```
for i in [0:3]{
```

```
    theta >>=1;
```

```
    p(-theta) q[i];
```

```
    h q[i];
```

```
    measure q[i]->b;
```

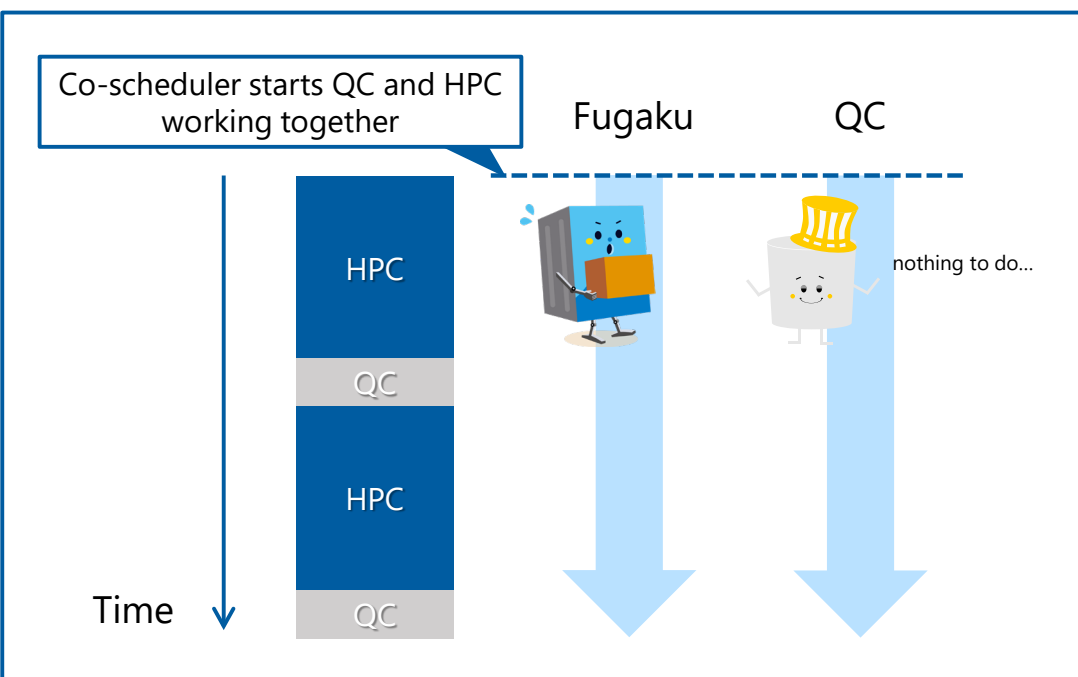
```
    theta[3] = b;
```

```
}
```

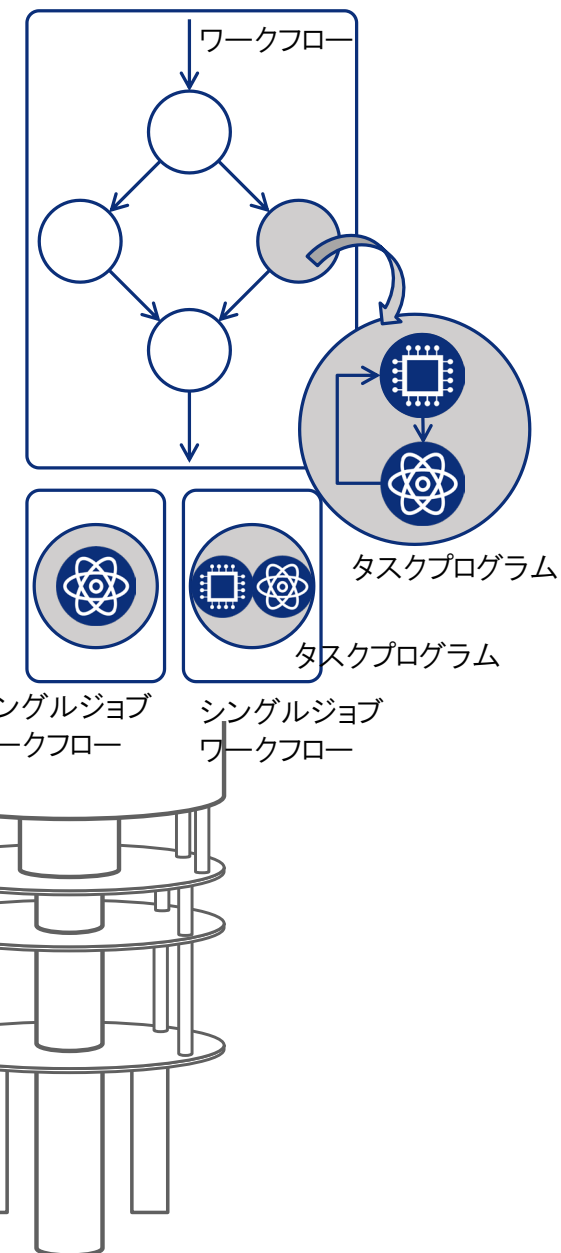
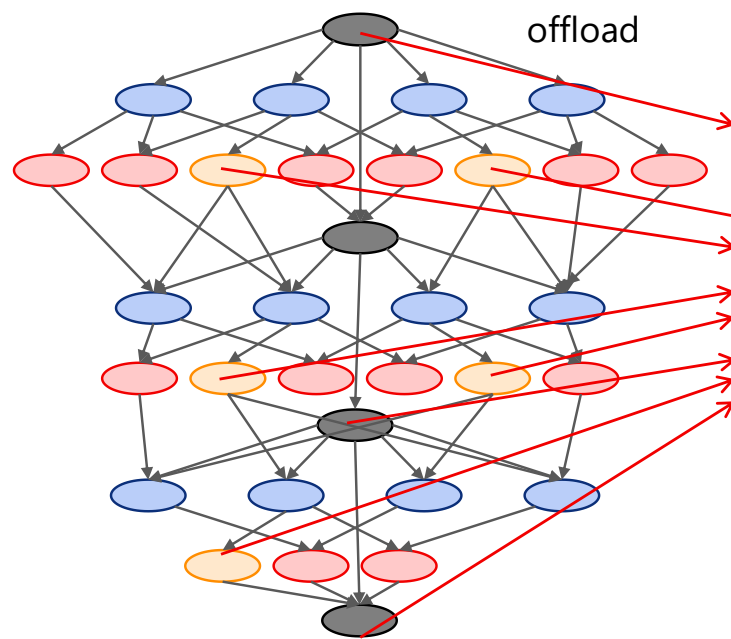
量子HPC連携プログラミングモデルとスケジューリング

- 量子HPC連携のための二階層プログラミングモデルとスケジューリング
 - (第一階層)スパコンおよび量子コンピュータのジョブをタスクとするワークフロー
 - 各々のジョブスケジューラにより各々のシステムの高い利用率が期待される
 - (第二階層)ジョブの内部での細粒度タスクプログラミング
 - 量子とHPCのより密な連携が必要な場合
 - こちらから量子コンピュータに投入されるジョブは最優先とする

粗粒度タスクの場合、コスケジューリング等でQC HPCを同時に開始するよりも個々のシステムのスケジューラに任せた方が高いスループットが期待できる(第一階層)



依存関係が複雑な細粒度タスクの場合、総実行時間やHPC側の待ち時間を軽減するため量子コンピュータへのオフロードは最優先で実行される



- 量子HPCハイブリッドプログラムのための遠隔呼び出しの試作を完了した。理研・筑波大が開発した遠隔呼び出しライブラリ(OmniRPC)を利用。

- 試作したライブラリを用いて、理研の量子コンピュータ「叡」の量子回路の実行ジョブを起動するためのREST API(web URLを用いたインターフェース)に接続し、富岳の計算ノードで実行した量子HPCハイブリッドプログラムから、「叡」を呼び出すことができることを確認
- 計画していた量子計算シミュレータを実行するGPUの整備が遅れたため、「叡」を利用し、確認のための実験を行った。

QC-HPC ユーザー

OmniRPCを用いた

量子HPC
ハイブリッド
プログラム

RPC
エージェント

起動

ログインノード

RPC スタブ
(Submitter)

サーバー

REST
-API

量子計算
シミュレータ
(GPU)

トークンを
手動で取得
(ユーザー
認証)

量子コンピュータへプロ
グラム実行をリクエスト

結果の取得

REST-API

インターネット

通常のRQCのユーザー

クラウドサーバー

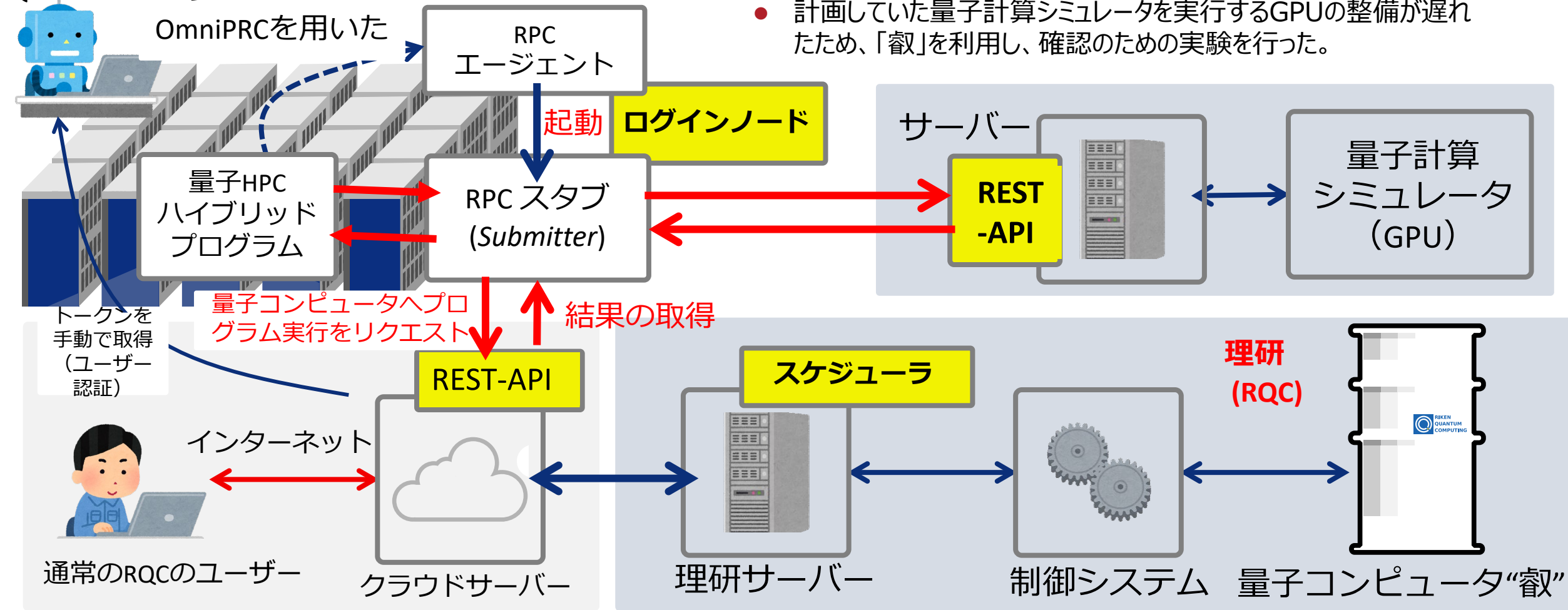
スケジューラ

理研サーバー

制御システム

理研
(RQC)

量子コンピュータ“叡”



JHPC-Quantum: 計算可能領域の開拓のための量子・スパコン連携プラットフォームの研究開発

実施者	国立研究開発法人理化学研究所、ソフトバンク株式会社（共同実施）東京大学、大阪大学
概要	量子コンピュータとスーパーコンピュータ（HPC）を連携するための量子・HPC連携システムソフトウェアを研究開発し、これを用いてこれまでのスパコンのみでは困難だった領域の計算を可能とする量子・スパコン連携プラットフォームを構築する。既存のスパコンのみの計算に対し量子・HPC連携アプリの優位性を実証するとともに、この計算プラットフォームで実行される量子・HPC連携ソフトウェアをポスト5G時代のネットワークで提供されるサービスとして展開する技術を開発する。

1. 開発目的

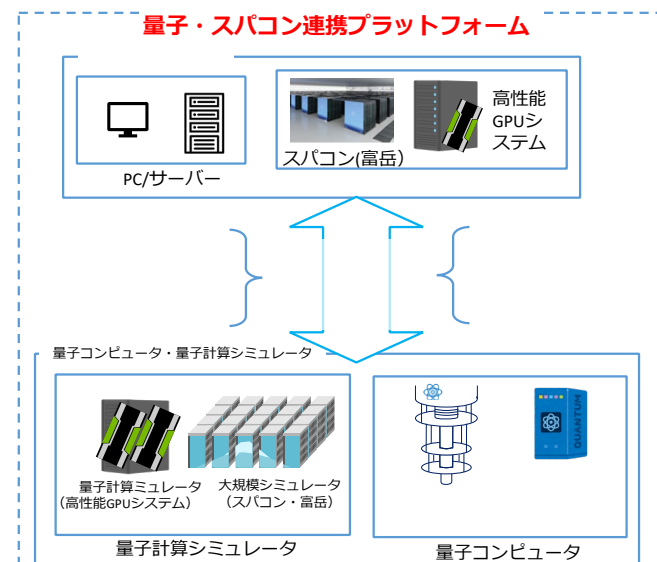
- 量子コンピュータは、従来のコンピュータと全く異なる原理で動作し、情報処理速度の劇的な高速化が期待されるが、現時点では、規模拡大と計算結果の誤り訂正の両立が困難であり、量子コンピュータ単独での実用化には時間を要する見込み。
- 一方で、デジタル化の進展により、情報処理能力の向上が急務であり、量子コンピュータの活用がいち早く求められているところ、古典コンピュータを組み合わせることで活用することが有望視されている。
- 本事業では、世界に先駆けて、量子コンピュータとスパコンを連携利用するためのソフトウェアやプラットフォーム、アプリケーションを開発・構築し、ポスト5G時代で提供されるサービスとして展開する技術としての有効性を実証していく。

2. 開発内容

- 量子・HPC連携ソフトウェア**：スパコンと量子コンピュータを連携させ、最適な計算資源をシームレスかつ効率的に利用するためのシステムソフトウェアを開発。
- モジュール型量子ソフトウェアライブラリ**：アプリ分野に合わせたモジュール型のソフトウェアを整備、量子コンピュータの特性に合わせたエラー緩和処理、回路最適化処理を実現する上位ソフトウェアライブラリを開発。モジュールとして組み合わせることで高度な量子アプリケーションを開発可能とする。
- 量子・スパコン連携PFのクラウド化技術**：事業展開を見据えて、量子アプリケーションの利用を支援するクラウド基盤ソフトウェアを開発。

3. 構築する量子・スパコン連携プラットフォームの構成

- 理研・計算科学研究センター（神戸）及び（和光）に特性の異なる2種類の量子コンピュータを整備。これらと富岳、及び東大・阪大スパコンと連携したプラットフォームを構築。



- ◆ 2026年度に量子・スパコン連携プラットフォームを運用開始し、それを用いて量子・HPC連携アプリケーションの有効性の実証に取り組む
- ◆ 2028年度下期、量子・スパコン連携プラットフォームのプレリリースを計画

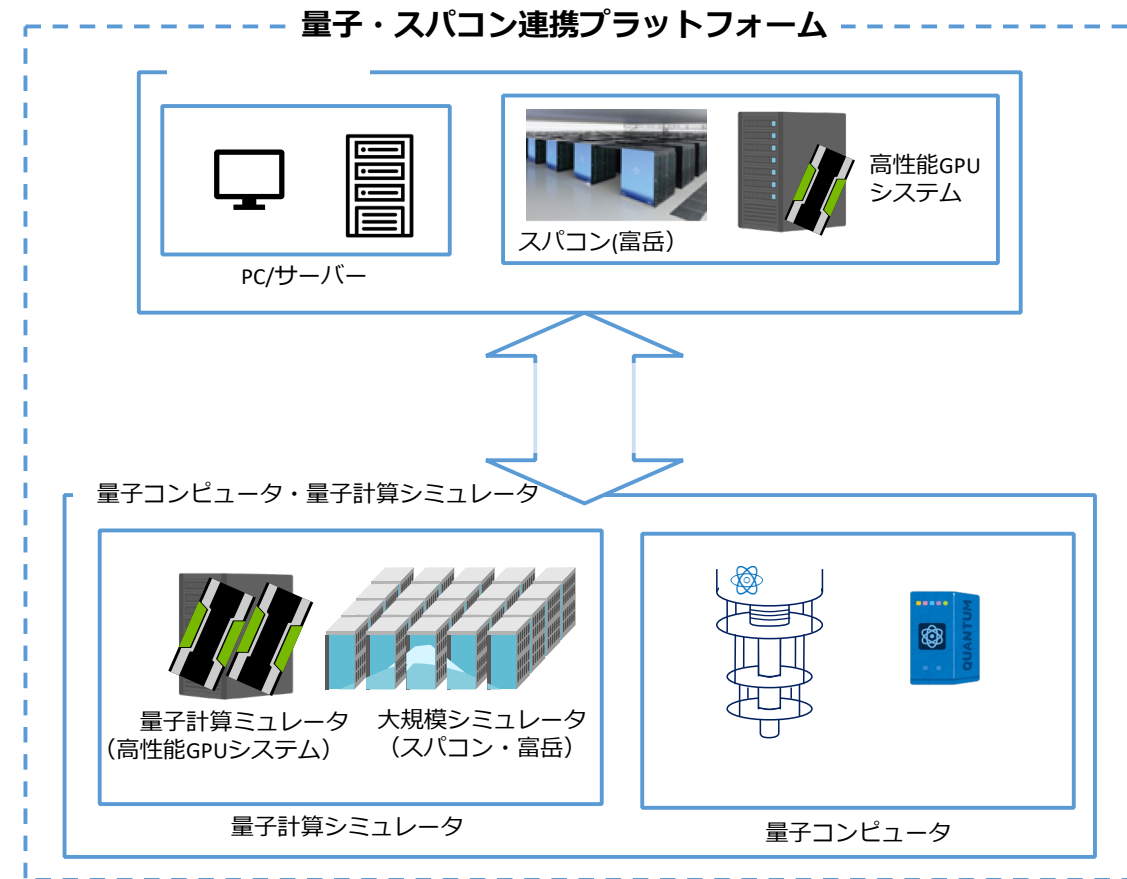
概要：量子コンピュータとスーパーコンピュータ（HPC）を連携するための量子・HPC連携システムソフトウェアを研究開発し、これを用いてこれまでのスパコンのみでは困難だった領域の計算を可能とする量子・スパコン連携プラットフォームを構築する。既存のスパコンのみの計算に対し量子・HPC連携アプリの優位性を実証するとともに、この計算プラットフォームで実行される量子・HPC連携ソフトウェアをポスト5G時代のネットワークで提供されるサービスとして展開する技術を開発する。

● なぜ、量子・スパコン連携プラットフォームが必要になるのか？

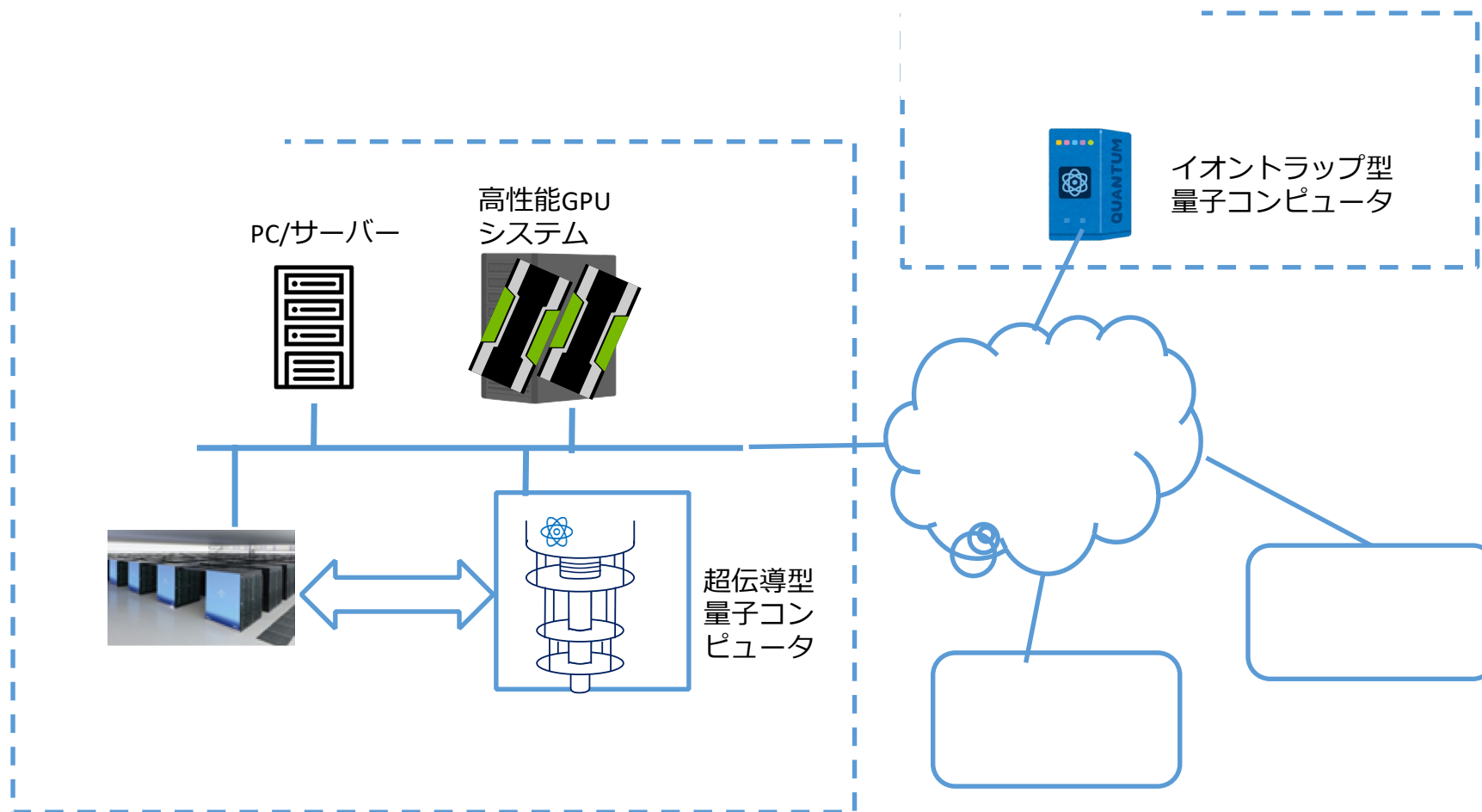
- ◆ **（スパコン側からの必要性）** これから開拓される量子コンピュータの利用を考えると、量子コンピュータに期待されるアプリケーション領域の多くは、これまでスパコンでの計算能力を用いてサポートされてきたアプリケーション領域であり、スパコン側からは量子コンピュータはこの領域のアプリケーションを大幅に加速する装置として捉えることが適切。
- ◆ **（量子コンピュータ側からの必要性）** 一方、現在、ハードウェアとして実現されている量子コンピュータの多くは、NISQ（Noisy Intermediate-Scale Quantum Computer）であり、活用するためには、従来のデジタルコンピュータによる適切な制御、ノイズによるエラーの緩和（error mitigation）、回路分割・最適化を行う必要がある。100qubitを越え、実用に向かうためには、スパコンの計算能力を合わせて上記を行うことが有効である。
- ◆ 量子コンピュータのアルゴリズムの研究開発には、量子計算シミュレータが不可欠。qubit数が大きくなると、スパコンが必要となる。

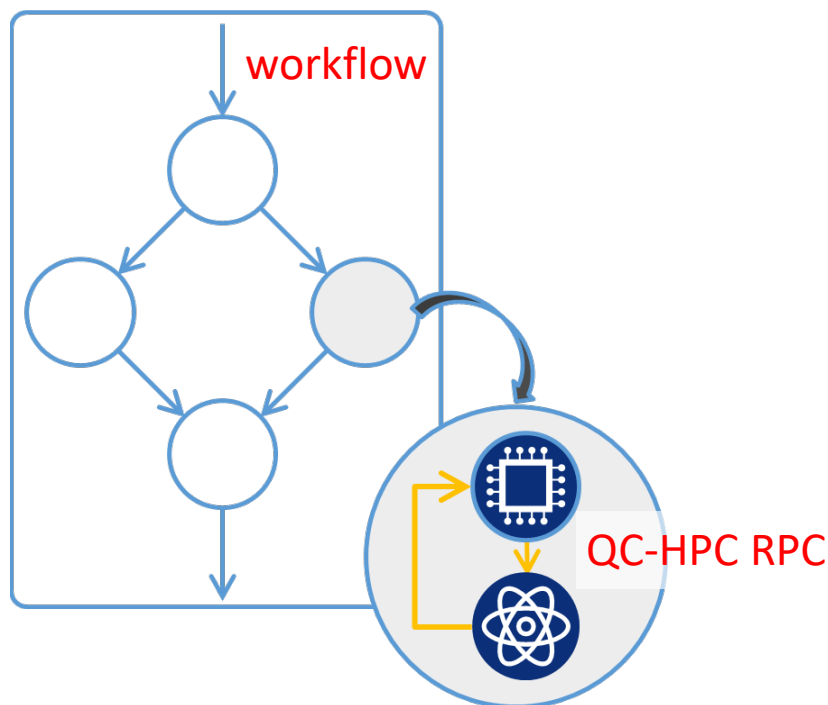
目標： 量子・スパコン連携プラットフォームを利用した量子・HPC連携システムソフトウェアを開発し、その有効性について実証する。ポスト5Gのサービスとして展開。

- **量子・HPC連携_基盤システムソフトウェア**は、スパコンの高性能アプリケーションから遠隔手続き呼び出しシステムによりシームレスに量子コンピュータの利用をサポート。連携スケジューラにより効率的に計算資源を利用。
- **実際の量子・HPC連携_基盤システムソフトウェアで有効性を実証するためには実際に最高性能スパコンと最先端の量子コンピュータが必要。**2種類の量子コンピュータ（超伝導型（神戸）、イオントラップ型（和光））を整備し、理研および東京大学、大阪大学のスパコンと結び、実際のスパコン・量子コンピュータを連携させるプラットフォームを構築・運用する。
- モジュール型量子ソフトウェアライブラリを整備、量子コンピュータの特性に合わせたエラー緩和技術、回路最適化技術を研究開発。
- 事業展開を見据えて、クラウドから**量子・HPC連携_基盤システムソフトウェア**の利用を支援するPaaS(Platform as Service)ソフトウェアを開発。



特性の異なる2種類の量子コンピュータを整備(2025年2Qに整備)、理研及び東京大学、大阪大学のスパコンから利用可能な量子スパコン連携プラットフォームを構築し、**実際の量子・HPCアプリケーションで有効性を実証。**



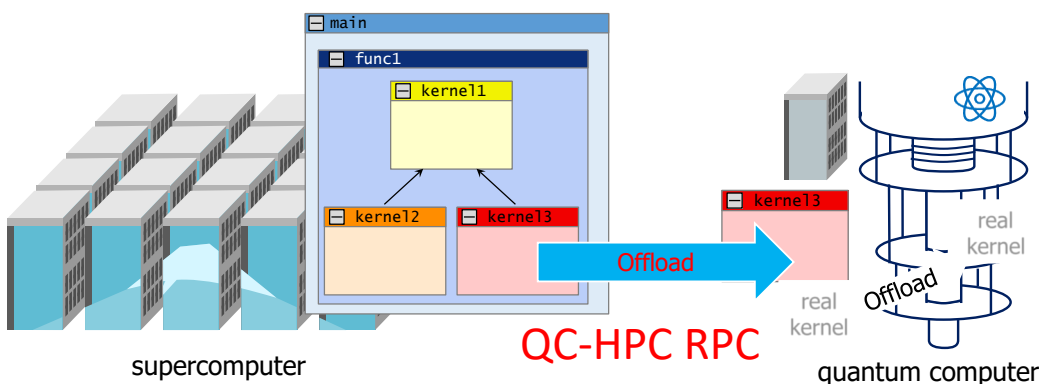


● ワークフロー実行モデル

- 依存関係グラフにジョブ（タスク）を記述し、ワークフロー実行ツールを用いて実行する。
- 各ジョブは独立したプログラムの実行であり、**MPIを利用した複数ノードによる並列プログラム**の場合もある。各ジョブは、先行ジョブの出力を入力とするプログラムである。
- ジョブの中には、RPC APIによるQCへのオフロードを含むプログラムもあり、これをQCジョブと呼ぶ。

● （非同期）RPC（リモートプロシージャコール）によるQC計算のオフロードを含むプログラミングモデル

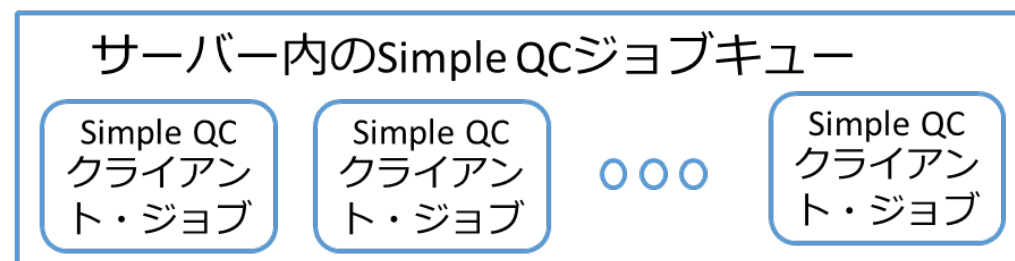
- RPC APIによるQC計算のオフロード。
- QCジョブによって実行されるプログラムは、このプログラミングモデルによって記述される。
- RPCは非同期に実行することもできる。この場合、ホスト側での実行と重なる可能性がある。



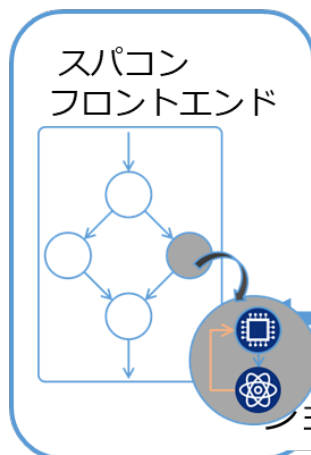
- 1. HPCプログラムの一部分として、量子プログラムを利用できるようにする。実行においては、量子コンピュータだけでなく、スパコンなどHPCシステムの効率的な利用を可能とする。
 - RPC（リモートプロシージャコール）プログラミングモデルの使用
 - HPCスケジューラ（ジョブ管理システム）とQC-HPC RPCリクエスト・スケジューラとの協調スケジューリング
 - HPCスケジューラによるQC-HPCジョブ数の制限（1ジョブまで？）
 - QC-HPCハイブリッドジョブからのRPCリクエストは高優先で実行されるべき
- 2. HPCプログラムだけでなく、これまでのqiskitやTKETなどのpythonのフレームワークで書かれたプログラムなど、通常のサーバーのシングルノードで動作するプログラムの実行を可能とし、これまでの量子コンピュータの利用環境と互換性のあるものとする。
- 3. 複数の量子コンピュータを複数のHPCシステムおよびサーバー環境から利用できるようにし、我々のユーザー管理のポリシーで運用できるようにする。ユーザーの認証、アカウントिंग等についてもサポートする。
 - QC-HPC RPC scheduler (front-end scheduler) and QC-vender's backend scheduler
 - QC-HPC RPC scheduler takes core of user authentication/accounting

HPCスケジューラ（ジョブ管理システム）と QC-HPC RPCリクエスト・スケジューラとの協調スケジューリング

単一ノード（サーバー）上で実行されるSimpleクライアント



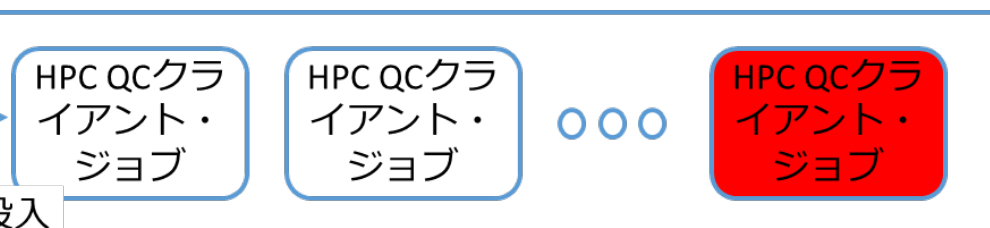
対話型のQCジョブまたは実時間QCジョブ



ワークフローは
フロントエンド
で実行される

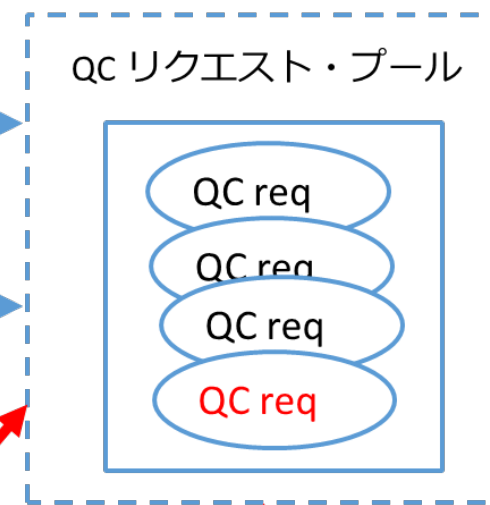
HPC/スパコン・クライアント

HPC/スパコン内のHPC QCジョブキュー



待ち時間を少なくするために一つだけの HPC QC クライアント・ジョブが高優先度のQCリクエストで実行される。

QC リクエスト・スケジューラ
(フロントエンド・スケジューラ)



RPCによる低優先度
QCリクエスト

RPCによる中優先度
QCリクエスト

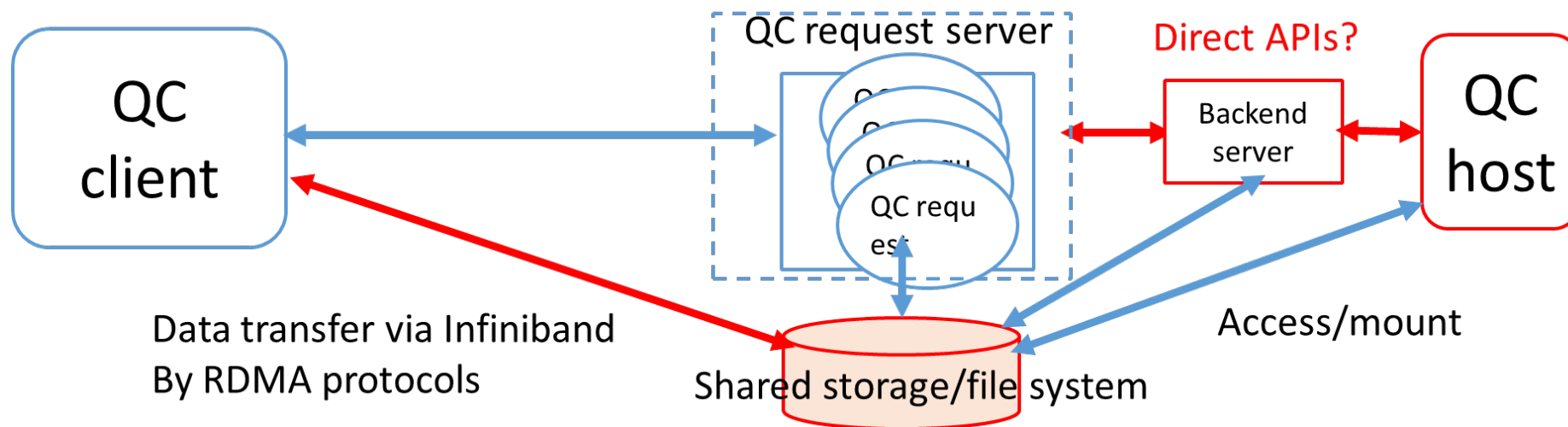
RPCによる高優先度
QCリクエスト

REST API or Direct APIs

QC バックエンド・
スケジューラ

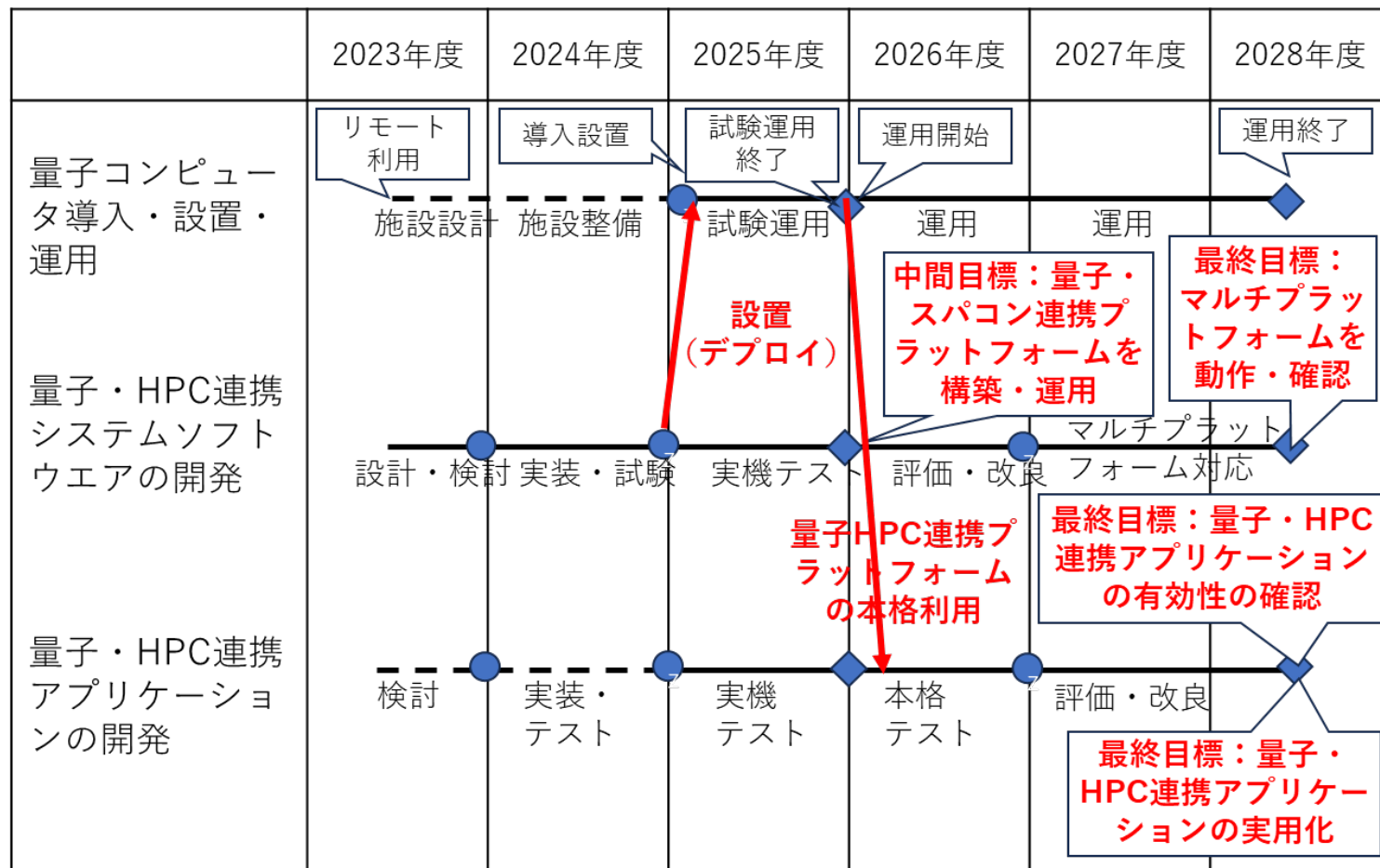


- 量子ビットのエラー修正の処理など高速な計算が要請される処理にスパコンが期待されているが、現在のスパコンは高スループットな処理が得意であるが、実時間処理には向いていない。⇒ **むしろ、ここには小規模並列GPUシステムが適している。**
- 本プロジェクトで整備するスパコンと量子コンピュータの高速ネットワークは、高速データ転送に用いることを検討している。



JHPC-quantumプロジェクトのマイルストーン

- 2023年11月、プロジェクト開始
- 2025年2Q、実機設置完了
- 2026年1Q、量子・スパコン連携プラットフォーム運用開始
- (2026年 2 Q: ステージゲート)
- 後半は、量子・HPC連携アプリの有効性実証、実用化へ



- **量子コンピュータ（超伝導型）では、量子ビット数が100を超えた。**
 - すでにシミュレータ（ステートベクトル法）では、シミュレートできる範囲を大きくこえている。
 - 量子コンピュータの技術は、エンジニアリングのフェーズにはいっているといえる。量子ビット数は年々増えていくことが期待される（量子の「ムーアの法則」はくるか？）
- **しかし、100量子ビットは実用にはまだまだ足りない！**
 - 忠実度は次第に改善されているが、依然、NISQ（ノイズあり）
 - さらに、FTQC（ノイズなし）が実用化されるにはまだ時間がかかる（10年？ 2050年？）
 - 企業ニーズが高い（と思われる）量子アニーラやQAOAアルゴリズムは、有望なアプリ領域であるが、できたとしても、スパコン等で行われる近似解よりもどのくらいいいかは明らかではない。（量子機械学習でも！）
- **量子コンピュータは、スパコンからはアクセラレーター（演算加速装置）の一つ**
 - （GPUのように）計算科学アプリの一部分を量子コンピュータにオフロードする使い方。（まったく別のアプリを開発するわけではない） **これからは、量子デバイスと呼ぼう！**
- **いろいろな問題がある量子コンピュータであるが、進歩していくのは確実。現在の規模ではまだまだ実用アプリは限定されるが、近い将来を見据えて、利用領域を開拓して準備しておくことは重要！！！！**