資料2

レイテンシコアの高度化・高効率化による将来のHPCIシステムに関する調査研究

Feasibility study on advanced and efficient latency core-based architecture for future HPCI R&D

石川　裕　　東京大学情報基盤センター
平木　敬　　東京大学情報理工学系研究科
青柳　睦　　九州大学情報基盤研究開発センター
新庄 直樹　富士通
飯田 恒雄　日立製作所
中村 祐一　日本電気

レイテンシコアの高度化・高効率化・・・・

2012//08/10

FS (Feasibility Study)概要

2018年頃設置可能並列システムを汎用型プロセッサからのアプローチでFS
アプリケーション、システムソフトウェア、アーキテクチャのco-design
システムソフトウェアスタック共通化 (From PC cluster to high-end machines)

ターゲットアプリケーション群

電力：20MW～30MW
設置面積：2000m²

ポータビリティ | スケーラビリティ | メモリ階層と性能 | 低消費電力機構 | 耐故障性

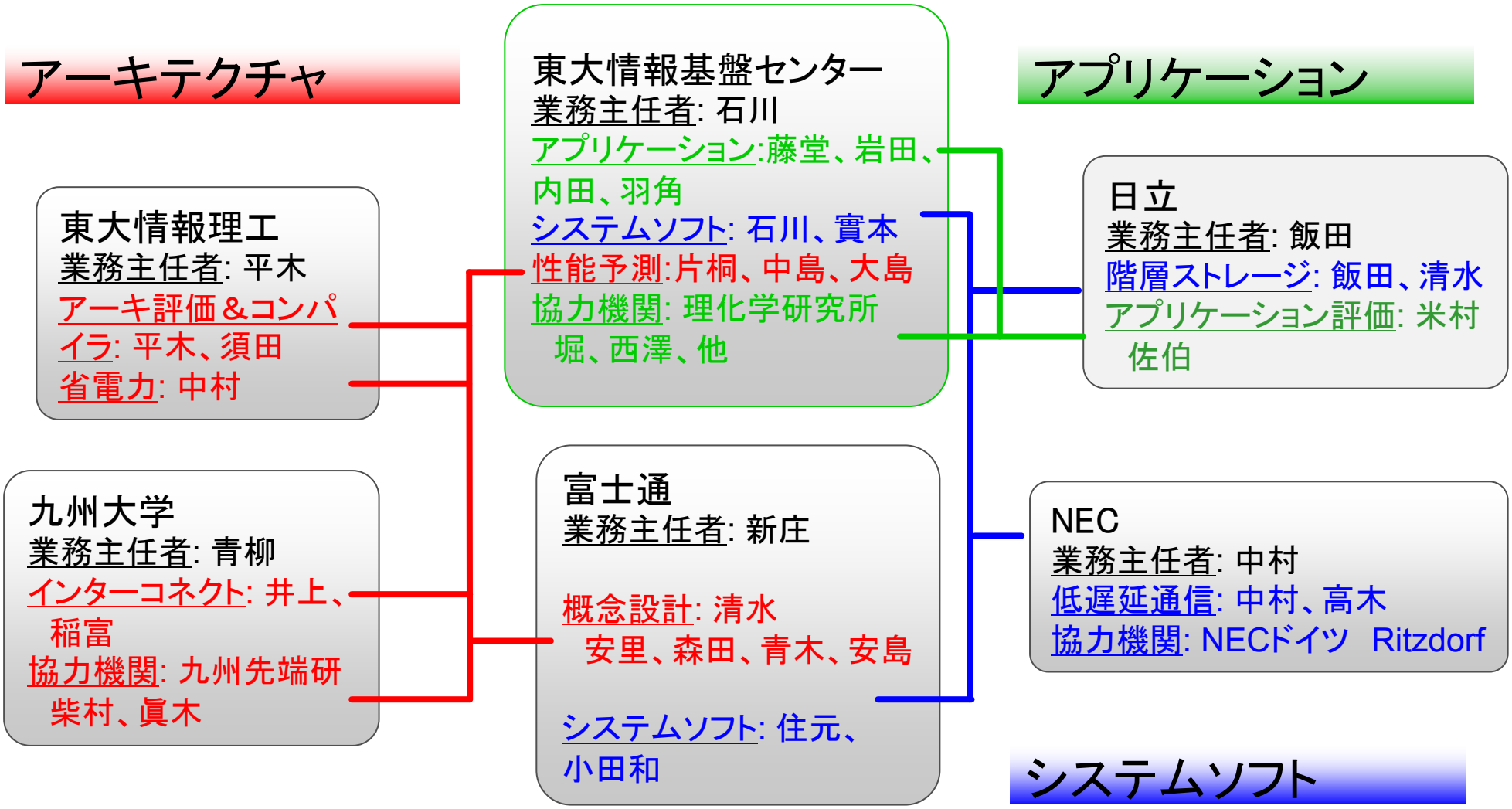
並列マシンアーキテクチャ
CPUアーキテクチャ、ノードアーキテクチャ、インターコネクトアーキテクチャ

階層ストレージアーキテクチャ
記憶デバイス、ネットワーク、ファイルシステム

システムソフトウェア
オペレーティングシステム、通信ライブラリ、ファイルI/Oライブラリ、数値計算ライブラリ

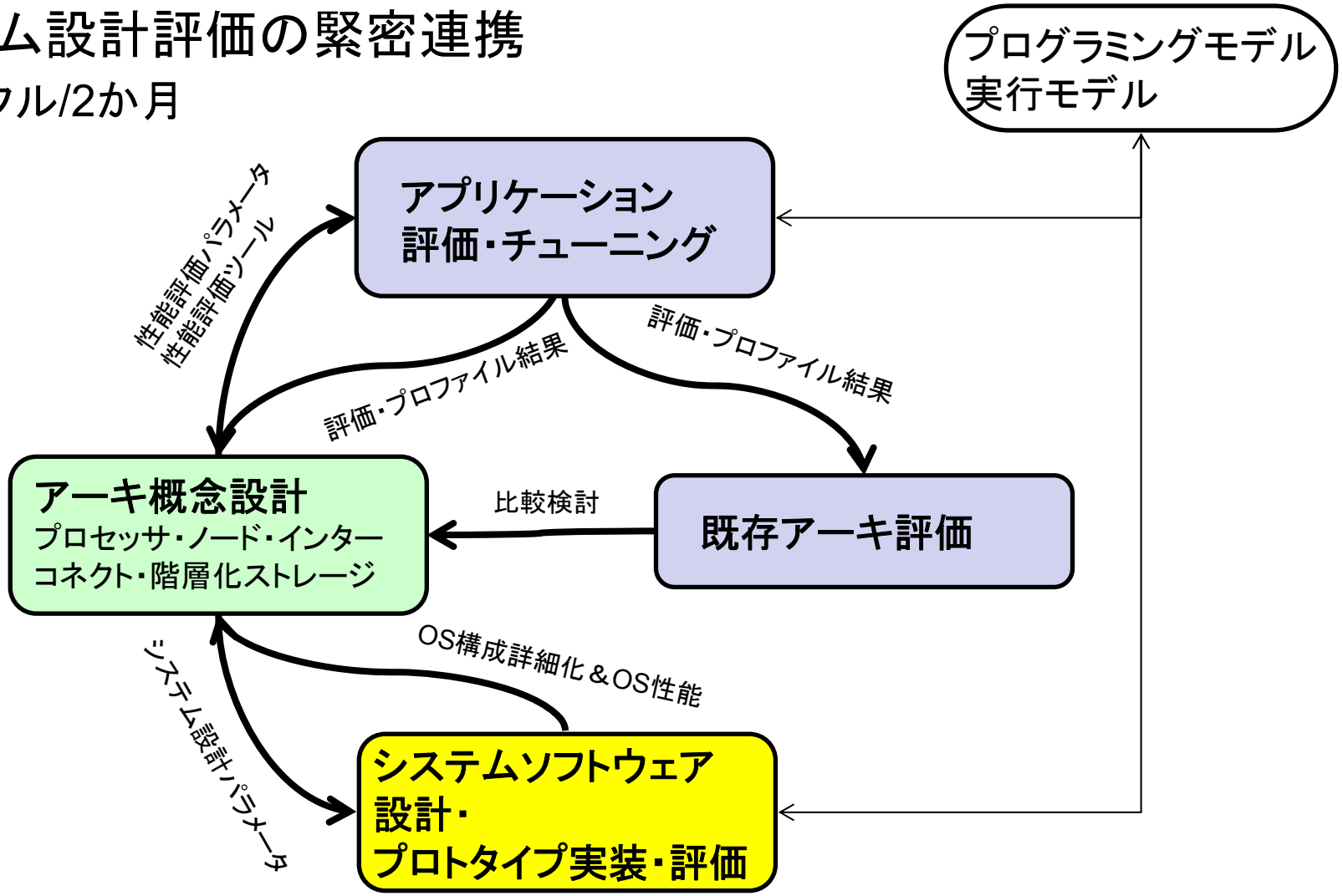
- プロジェクトの総合的推進及びシステムソフトウェア及びアプリケーション性能予測に関する検討
  （東京大学情報基盤センター・大気海洋研・物性研・工学系研究科、協力機関：理研AICS）
- アーキテクチャ評価およびコンパイラ技術と省電力機構の検討　（東京大学情報理工学系研究科）
- インターコネクト性能推定環境の検討
  （九州大学情報基盤研究開発センター、協力機関：九州先端科学研究所）
- アーキテクチャ概念設計およびシステムソフトウェアに関する検討 （富士通）
- 階層ストレージとコモディティ向けシステムソフトウェアに関する検討 （日立製作所）
- 低遅延通信機構の検討 （日本電気、協力機関：NECドイツ）

体制


アーキテクチャ
東大情報理工
業務主任者: 平木
アーキ評価＆コンパイラ: 平木、須田
省電力: 中村
九州大学
業務主任者: 青柳
インターコネクト: 井上、稲富
協力機関: 九州先端研 柴村、眞木
東大情報基盤センター
業務主任者: 石川
アプリケーション:藤堂、岩田、内田、羽角
システムソフト: 石川、實本
性能予測:片桐、中島、大島
協力機関: 理化学研究所 堀、西澤、他
富士通
業務主任者: 新庄
概念設計: 清水 安里、森田、青木、安島
システムソフト: 住元、小田和
アプリケーション
日立
業務主任者: 飯田
階層ストレージ: 飯田、清水
アプリケーション評価: 米村 佐伯
NEC
業務主任者: 中村
低遅延通信: 中村、高木
協力機関: NECドイツ Ritzdorf
システムソフト

Co-designの進め方

- アーキテクチャとアプリ評価チューニングおよびシステム設計評価の緊密連携
  - 1サイクル/2か月



FS概要：進め方

アプリケーション
東京大学情報基盤
ALPS, RSDFT,
NICAM, COCO

最適化手法

アーキテクチャモデル
- ノード単体性能パラメータ
- 通信性能パラメータ
- ストレージ性能パラメータ

システムソフトウェアモデル
- 基本数値計算ライブラリ性能パラメータ
- MPI通信性能パラメータ
- ノード・ローカル・グローバルファイルシステム性能パラメータ

CPUプロファイル
通信プロファイル
I/Oプロファイル

FX-10, 京

アプリケーション
実行性能モデル

実行シナリオ群

性能予測
- ノード単体
- 並列性能
- 並列ファイルI/O性能
- チェックポイント性能
- 階層データ移動性能

他マシンの比較：86系、Power系

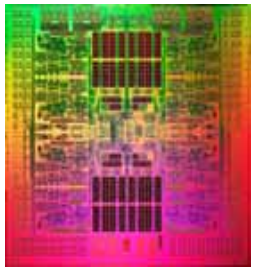
注：現在京のアプリにないデータインテンシブ系に関しては別対応

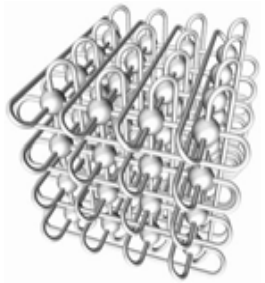
- ✓ システムソフトウェアはProof of concept実装も行う
- ✓ **平成25年度アプリチームとの連携でより広いアプリケーションを使用して評価**

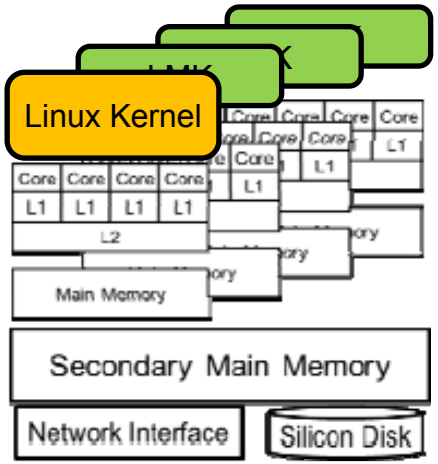
「今後のHPCI技術開発に関する報告書」を尊重し、京およびFX10におけるアプリケーション並列性能およびI/O性能、耐故障性および運用・保守の観点で課題を精査、概念設計に反映

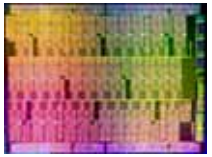
FS概要：特徴と公開成果物

- 汎用型プロセッサからのアプローチ
  - 京コンピュータ、FX10CPUを基に、メニィコア化、SIMD化、通信機構の高度化を検討
  - ハードウェア概念設計時からシステムソフトウェアスタックの設計開発の一部を先行し、コモディティベースクラスタでも利用できるよう設計。国際協力。システムソフトウェアスタックのガラパゴス化を回避
    - ヘテロOSカーネル
    - 低遅延通信ライブラリとMPI
    - スケーラブルファイルI/Oと階層化ストレージ
- 運用を考えたシステム設計
  - ユーザアプリケーションの利用シナリオ
    - 課題解決に必要とされる資源量と必要時間
      - 大規模、中規模、小規模実行
  - 京コンピュータ、FX10@東大における運用経験
    - スケジューリング、耐故障性
- 開発計画、必要経費（開発経費、設置経費）、保守・運用経費の見積もり

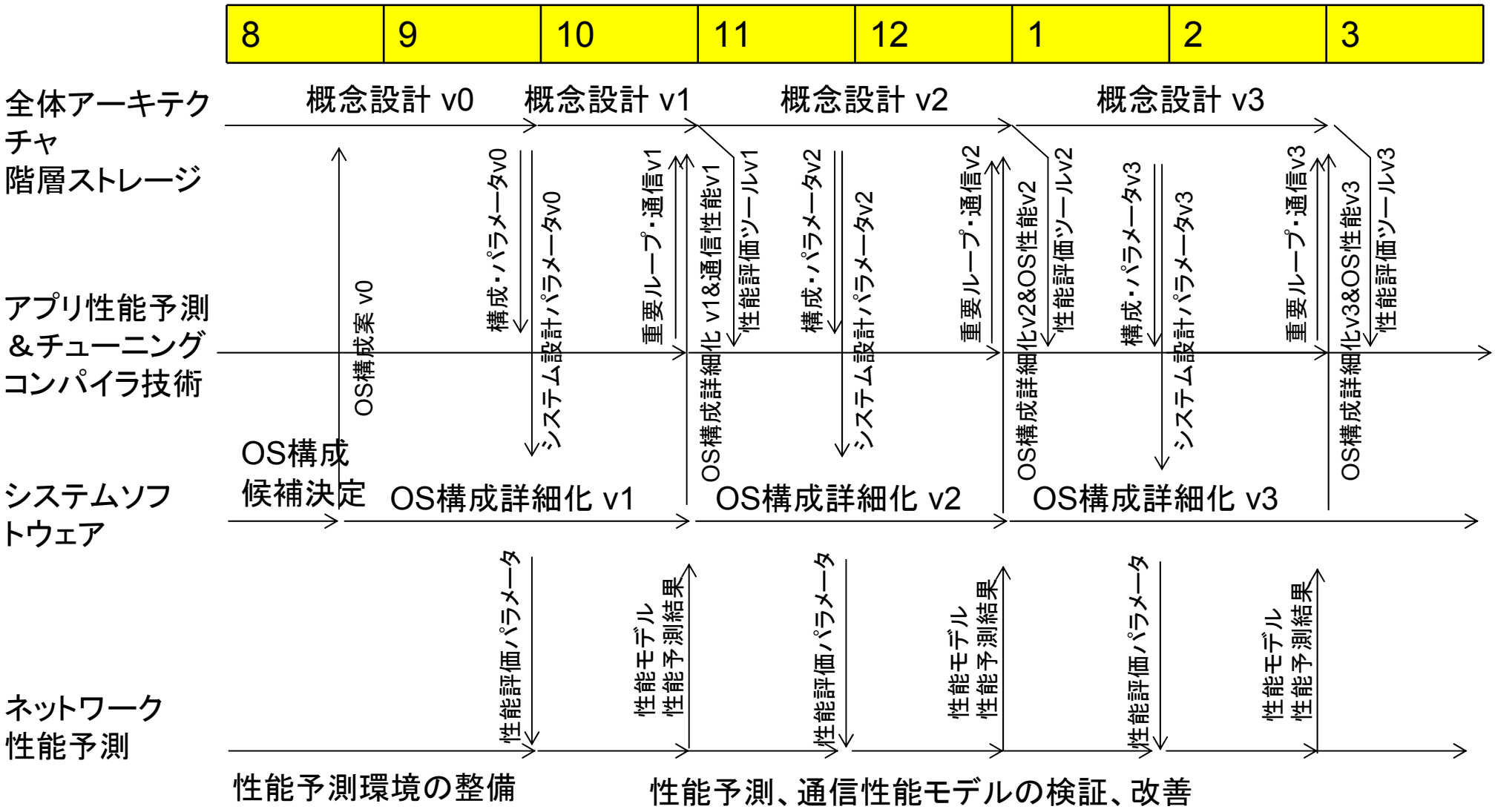






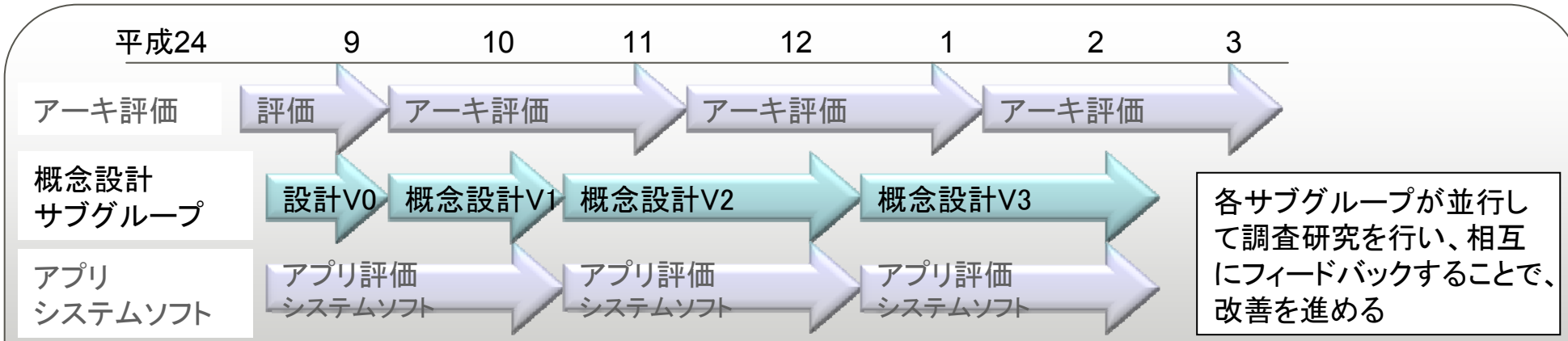


H24年度全体スケジュール



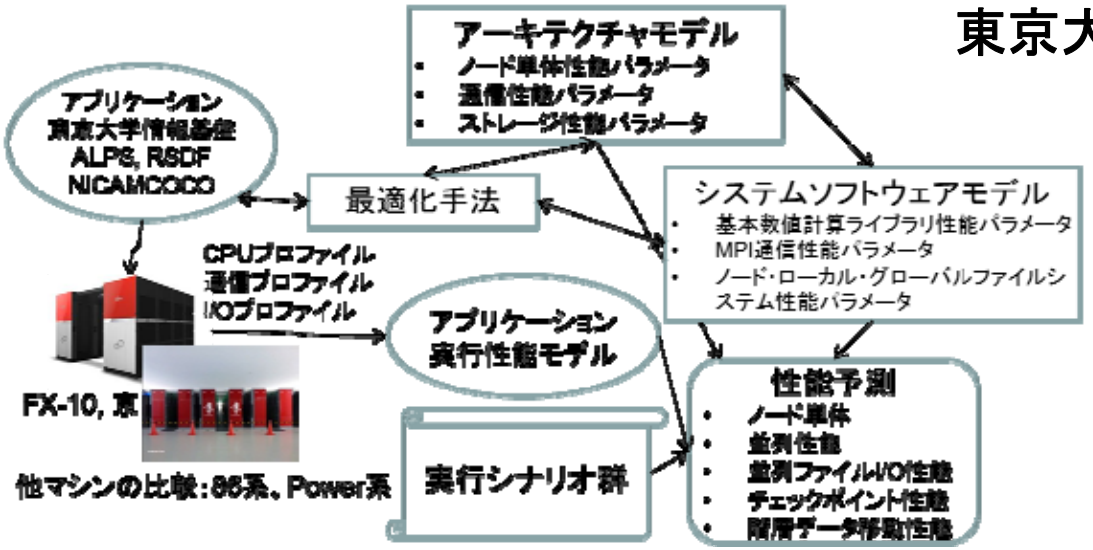
アーキテクチャ概念設計の調査研究の進め方

- 「今後のHPCI技術開発に関する報告書」（白書）の汎用型スペックとアプリ要求スペック
  - 白書の汎用（従来）型のスペック
    - 設置面積　2,000m^2
    - 消費電力　20~30MW
    - 総演算性能　200～400 PFlops
    - 総メモリ帯域　20～40 PB/s
    - 総メモリ容量　20～40 PB
    - インタコネクト　16 GB/s × 8/node (4D-Torus)
      or 32 GB/s × 64 port/32 nodes (Dragonfly)
  - 白書のアプリケーション要求スペック
    - 総演算性能　800～2500PFlops
- 調査研究の進め方
  - 技術動向の予測に基づき、複数のアーキテクチャ設計方針を策定
  - 白書のアプリケーション要求性能に近づけるために、新技術開発を概念設計に加味



- 概念設計の出力：　構成・パラメータ、性能推定ツール
- 概念設計への入力：　重要ループ、重要通信パラメータ、アーキ評価

アーキテクチャ評価およびコンパイラ技術と省電力機構の進め方



東京大学情報理工学系研究科

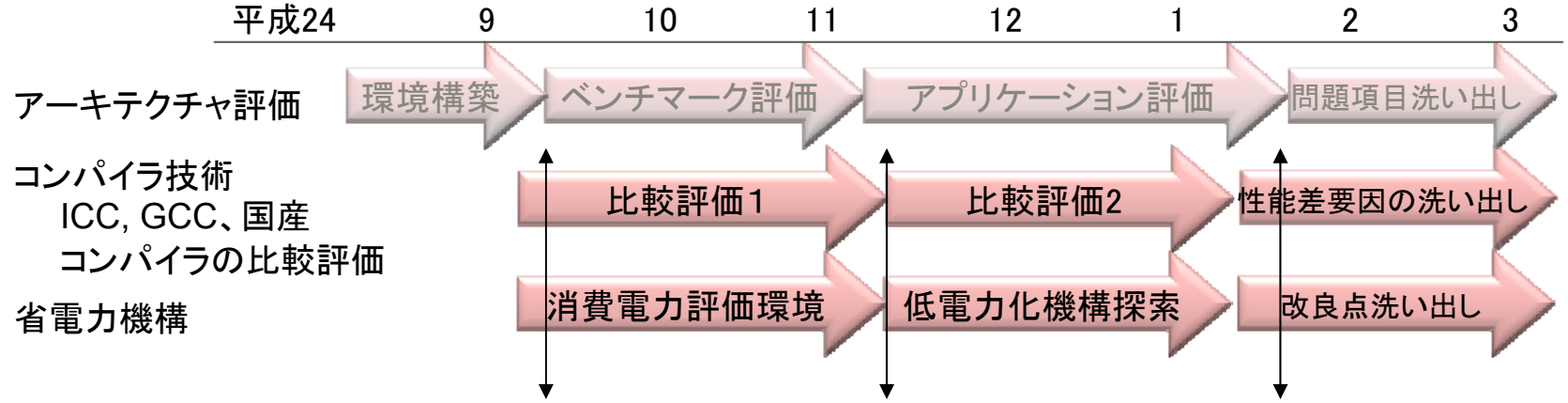
アーキテクチャ評価およびコンパイラ技術

- 評価環境の構築：FX10,IvyBridge
  環境整備、ソフトウェアの開発
- アプリケーションによる評価：
  標準的ベンチマークソフトと第一原理
  計算等アプリケーションによる評価

省電力機構

- インターコネクト部の消費電力分析と
  省電力機構の検討

スケジュールとインターラクション



アプリ・アーキとのインターラクション

予備評価結果FB
評価アプリ導入

ベンチマーク結果
アーキテクチャ原案
システム性能差の結果

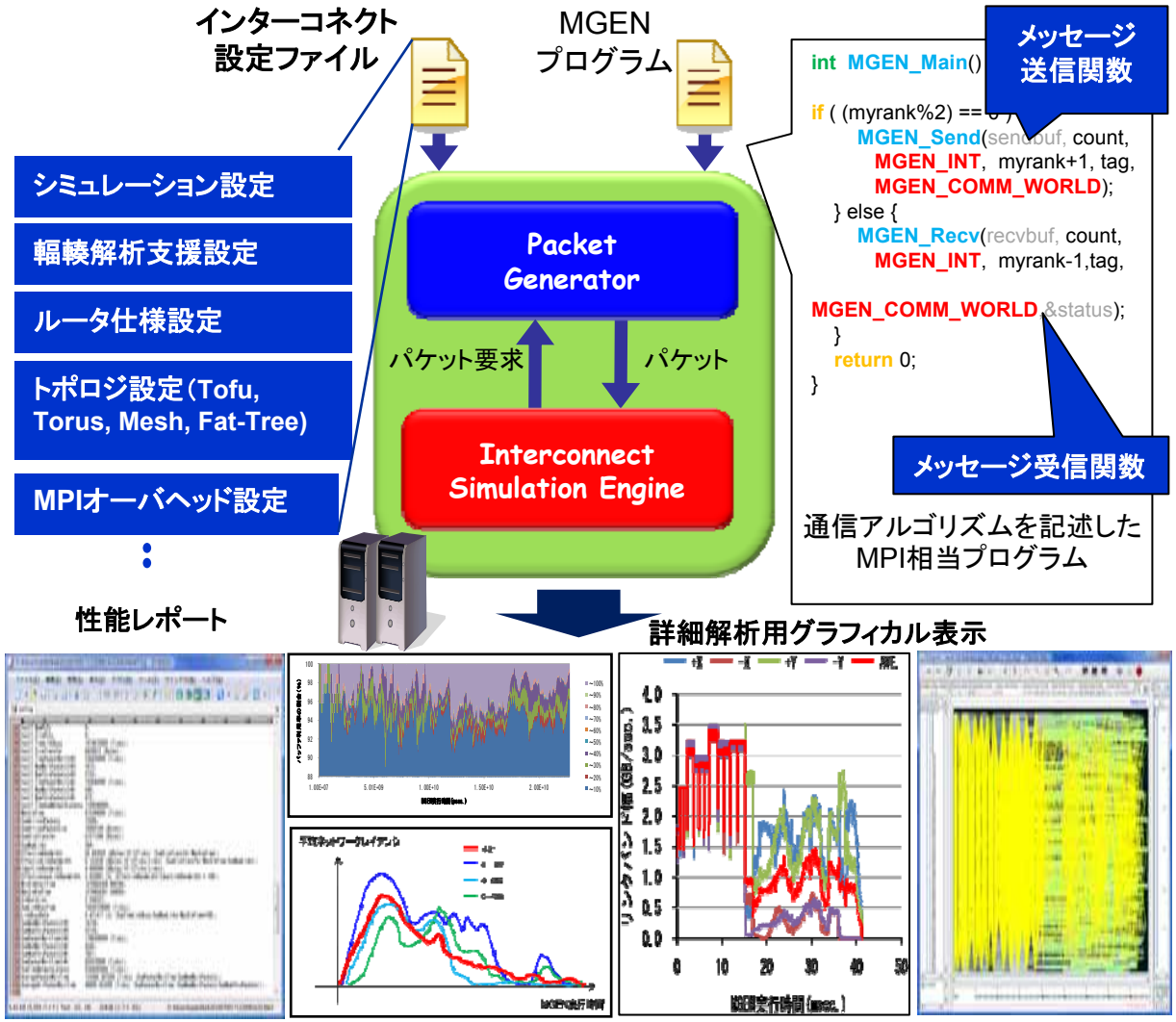
アプリケーション評価結果
概念設計V2,3情報
システム性能差の結果
低電力化効果の結果

注：スペース上の問題でインターラクションは一部記載していない

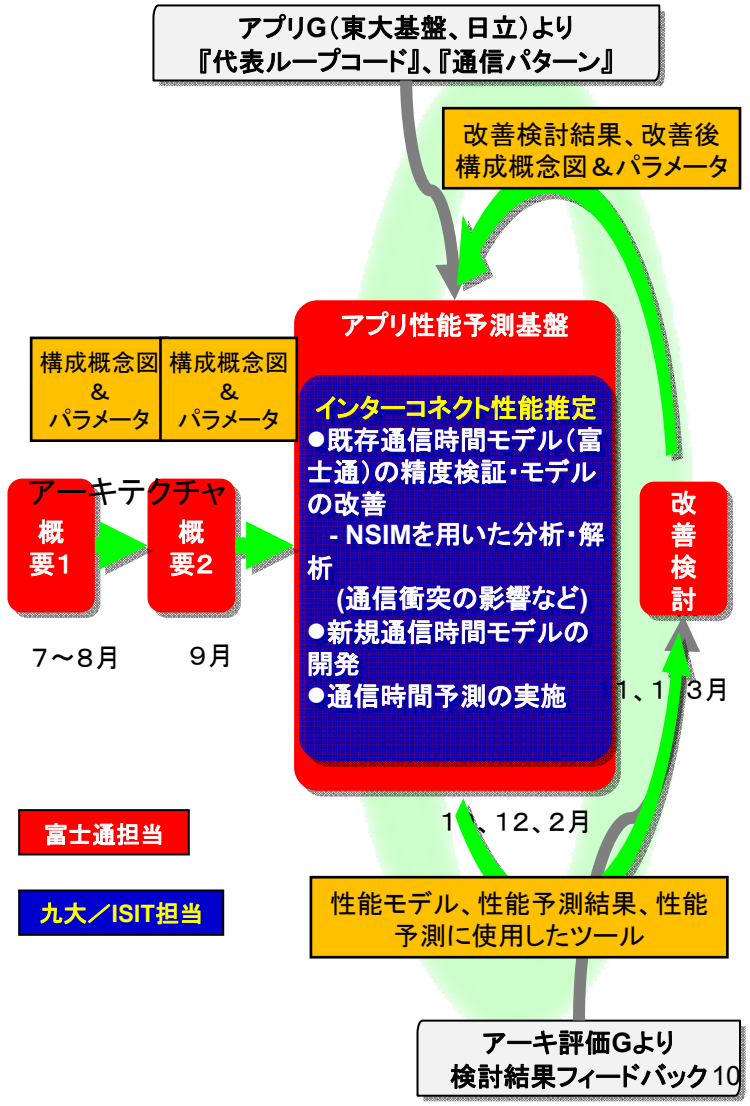
インターコネクト性能評価に関する検討の進め方

九大・ISIT

■ 性能予測手法



■ スケジュール（進め方）



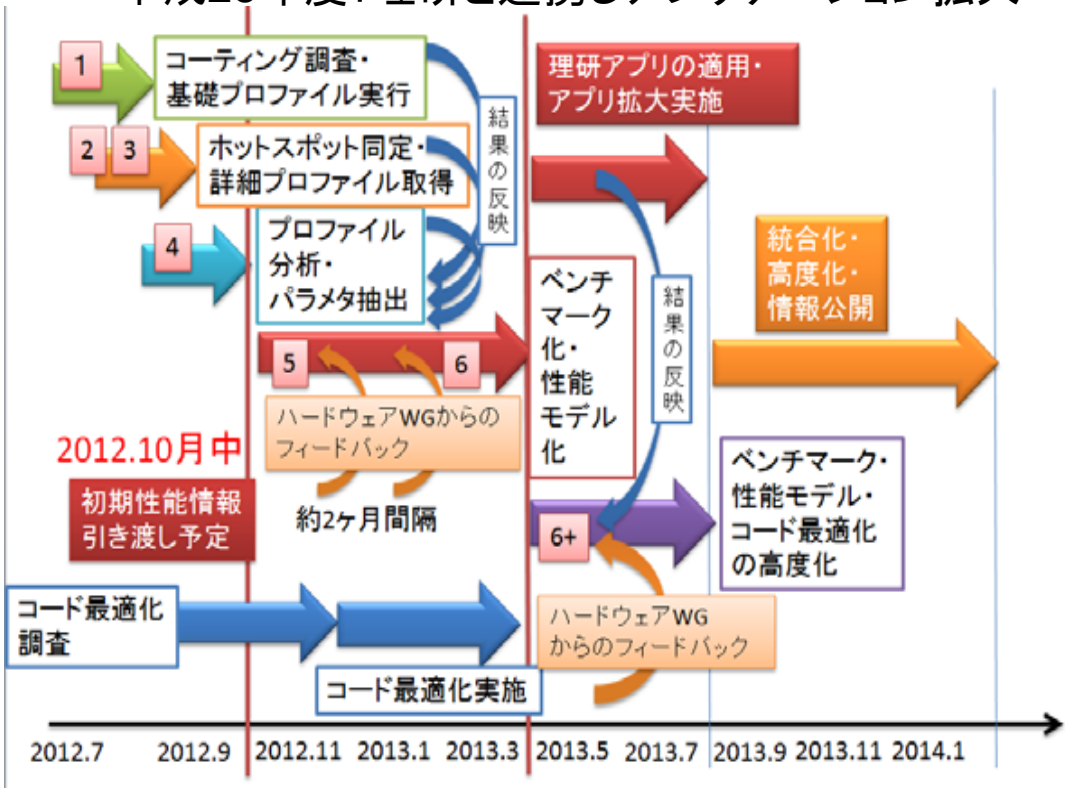
アプリケーション性能予測に関する検討の進め方

性能予測手法

1. **ホットスポット同定**：基本プロファイラで複数の**ホットスポット**（ループレベル）を同定、全体性能の近似
2. **カーネル分離**：（目視により）計算部分、通信部分、I/O部分の分離
   - 計算部分：演算カーネル
   - 通信部分：通信カーネル
   - I/O部分：I/Oカーネル
3. **通信パターン確認**
4. **詳細プロファイルと分析**：詳細プロファイラを用い、ホットスポットごとにハードウェア性能情報を取得し分析
   - 演算カーネルの演算効率／命令発行量／キャッシュ利用効率
   - 通信カーネルの通信回数／量／通信待ち時間
   - I/Oカーネルのデータ読み書き量／頻度
5. **ベンチマーク化**：ホットスポットのみで動作するようにコードを再構成
   - マシン特化の書き方、および、汎用的な書き方、の2種を区別
   - 演算カーネル、通信カーネル、I/Oカーネル分類
6. **詳細モデル化**：ハードウェア因子を用いた数式による実行時間の近似

スケジュール予定

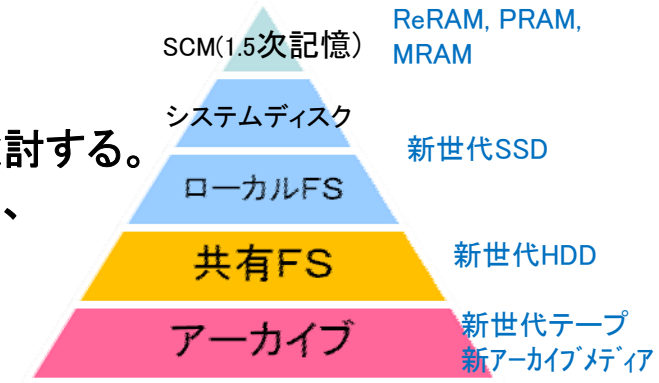
- 平成24年度：4種アプリの性能プロファイリングを行い、ハードウェア設計に反映
  - エクサ環境で想定される実行モデルをアプリケーションごとに作成
  - 2か月間隔でハードウェア設計をフィードバック
  - コード最適化も同時に行い適切な性能プロファイリングを実現
- 平成25年度：理研と連携しアプリケーション拡大



階層ストレージに関する検討の進め方

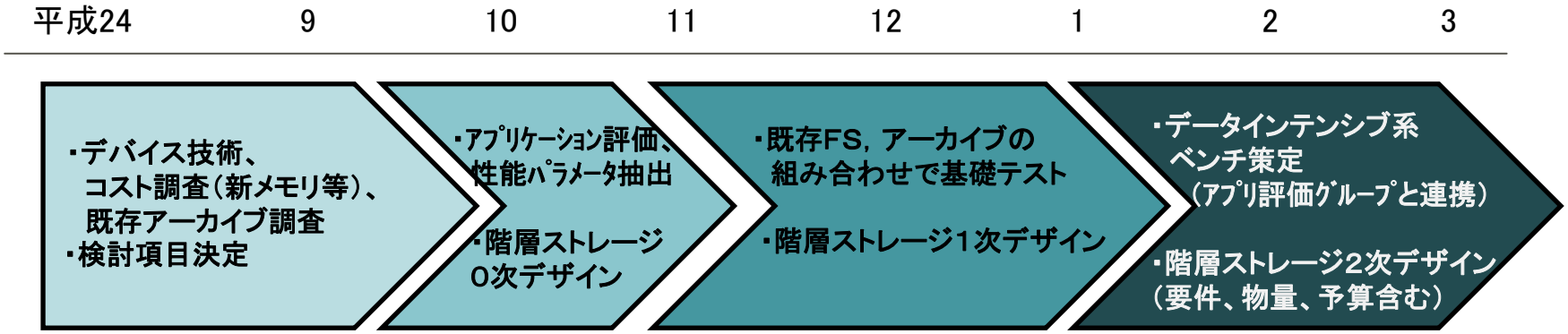
【目的】

・計算ノードローカルストレージ(システムディスク)、
計算ノードローカルストレージ(ユーザ、ステージング、チェックポイント)、
グローバルストレージ(共有ファイルシステム)、
アーカイブをトランスペアレントに扱えるファイルI/O機構(階層間API)を検討する。
・2018年に利用可能な、新メモリ、新アーカイブメディアの技術動向も調査し、
ストレージ階層への取り込みを検討する。



【調査研究の進め方】

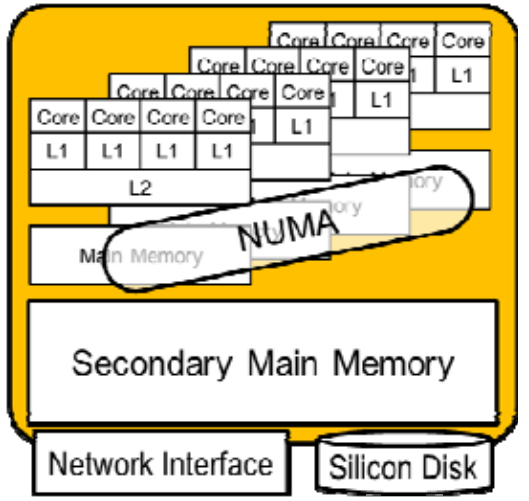
・「京」、FX10の現状アプリからストレージ要求パラメータを取得、将来性能導出
・ゲノム系アプリからデータインテンシブ系ベンチマークを策定(ランダムI/O系?)
・トレンド・動向から利用可能なストレージパラメータを抽出
・ローカル・グローバル・アーカイブ階層の性能設計、機能設計



システムソフトウェアに関する検討の進め方

- OS設計に必要なハードウェア構成を決めて、proof of concept実装を行いアーキテクチャ概念設計に反映
- 下表の検討すべき性能阻害要因に基づくシステムソフトウェア構成法および新OS機構の設計・試作・評価
- システムソフトウェア研究開発の国際協力（国際標準を含む）実現に向けて関係機関と調整

ノードアーキテクチャの一例



性能阻害要因

Core Capability	O(100) Core / Node	O(1M+) Node, O(100M+) Core
• Architectural support for scalability • Programming Model & Execution Model		
• Reducing cache pollution • Localizing data • Managing memory hierarchy • Utilizing SIMD resource	• Reducing memory contention • Reducing data movement among cores • Providing fast communication • Parallelizing functions achieving less data movement • Sharing hardware resources	• Reducing information whose size grows by # of cores & nodes, e.g., comm. connection. • Supporting strong scale in terms of communication • Handling huge number of files, access and store • Providing fault resilience

OS構成の一例