

Hyper-Aether

AIネイティブ時代の 計算基盤を再定義するアーキテ クチャ

AIが実行構造そのものを組み替え、
推論・実行・保護を一体化。

柔軟性

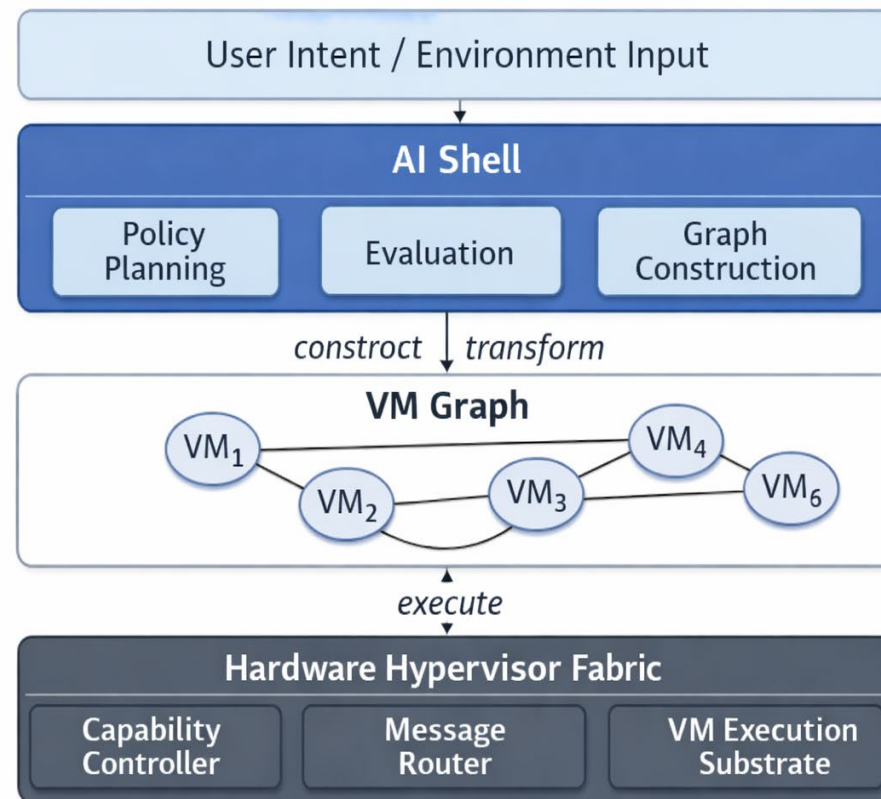
VMグラフを動
的に再構成

安全性

能力ベース保
護をハード実
装

統合性

AI制御と実行
基盤を接続



Virtual machines are the primary execution units; graph structure is synthesized by the AI Shell.

従来の「ソフトがハードの上で走る」発想から、
「AIが計算構造を生成する」発想へ。

AIワークロードは、固定的なソフト構成よりも動的な構造制御を必要としている。

従来の限界

- OS中心で制御層が肥大化
- 実行と推論が分離
- 構成変更が重い

Hyper-Aetherの答え

- VMを基本単位にする
- AI Shellが構造を組む
- ハード側で仮想化と保護を担う

期待できる価値

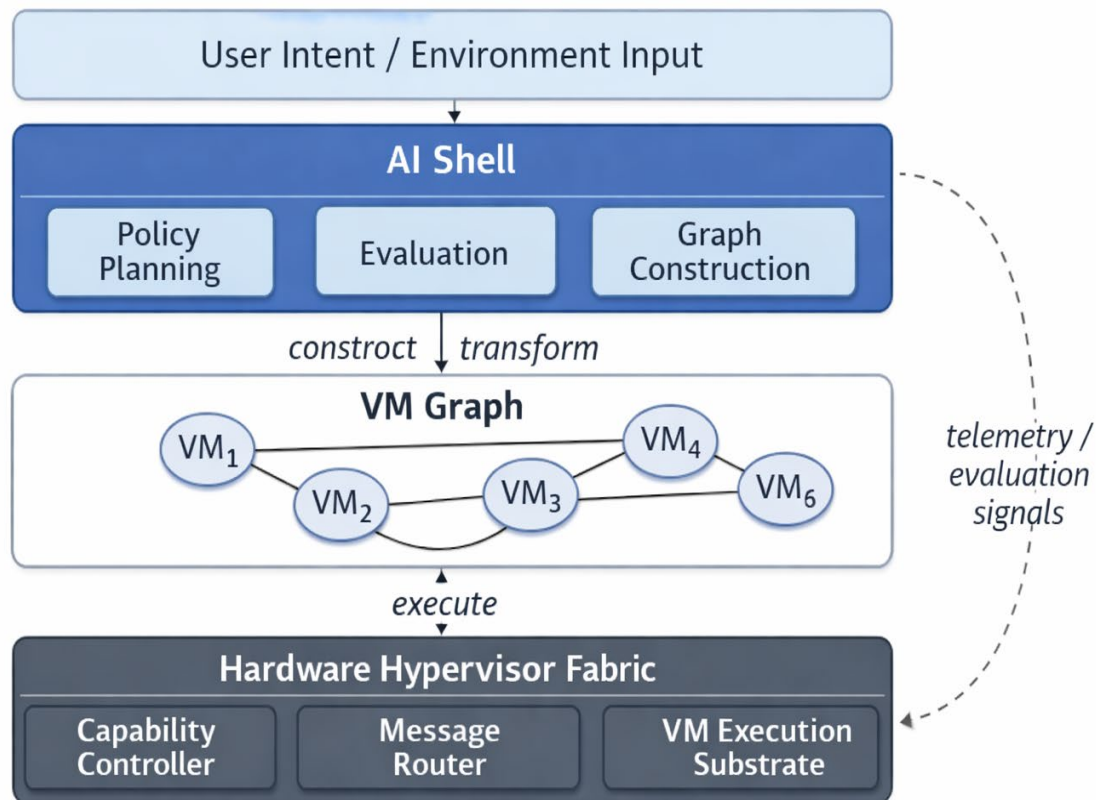
- 柔軟なオーケストレーション
- 安全な分散実行
- 意味評価を含む制御



固定的な実行環境から
“意味で最適化する計算基盤”へ

営業メッセージ

Hyper-Aetherは、AI時代の複雑な処理系に対して、実行構造をその場で組み替えられる“適応型コンピューティング基盤”を提示します。



Virtual machines are the primary execution units; graph structure is synthesized by the AI Shell.

1. 入力を受ける

ユーザー意図や環境情報を受け取り、どの構造が必要かを判断。

2. VMグラフを構築

必要な機能をVMとして配置し、通信経路を定義。

3. 実行しながら評価

実行結果やテレメトリをもとに、構造をさらに改善。

要するに、AIが“アプリ”ではなく“計算環境そのもの”をオーケストレーションする。



Hyper-Aetherでは、プログラムは“コードの列”だけではなく、VMのつながり方そのものも含む。

従来型

- 命令列を順番に実行
- 構造はほぼ固定
- OSが主に制御

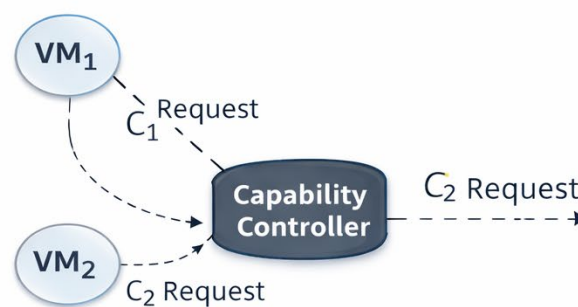


Hyper-Aether

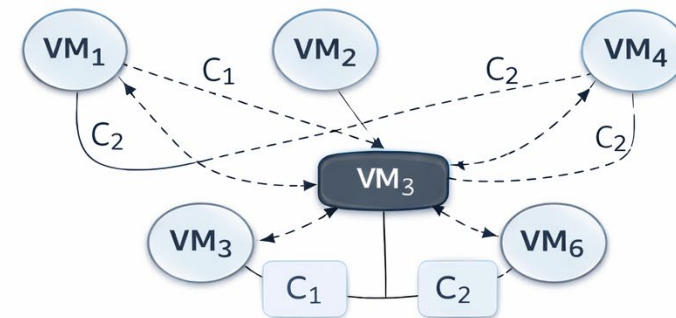
- グラフを書き換える
- 構造自体が実行状態
- AI Shellが変換方針を与える

ビジネス的な伝え方

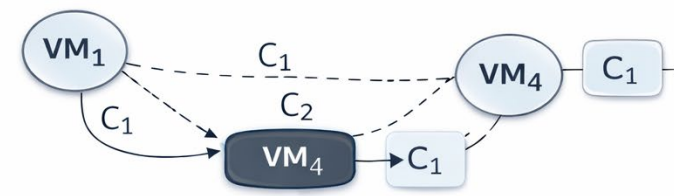
“処理を流す箱”ではなく、“目的に応じて形を変える基盤”。



(a) Capability Proxying



(b) Capability Routing Graph Construction



(c) Capability Routing

意味 = 構造 × 関係 × 価値

構造

どんなVMがあり、
どう配置されて
いるか

関係

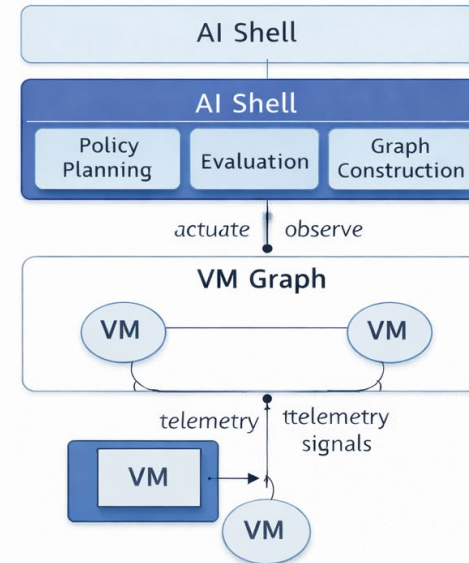
それらがどう接続
し、どう影響する
か

価値

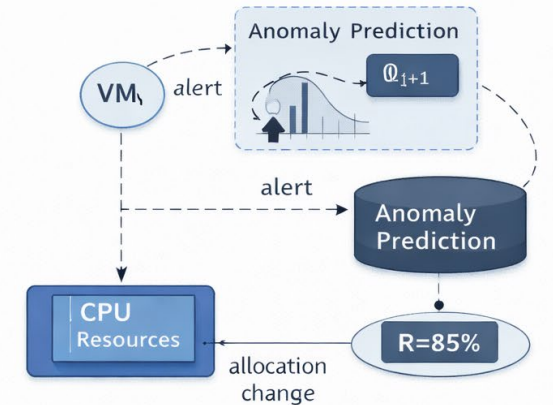
その構造に、どん
な意味や評価があ
るか

営業向けに言い換えると

Hyper-Aetherは、ただ速く実行するだけでなく、
“何が価値ある構造か”を評価しながら動ける。



(a) Control Loop



(b) Proactive Fault Detection

(c) Control Comput and likel Ressonse Allocation

Hyper-Aetherが刺さる相手

AIプラットフォーム設計者

複雑なワークロードを安全に分散させたい

次世代アーキテクチャ検討層

OS/仮想化/推論の分離に限界を感じている

研究開発・戦略部門

AGI/エージェント時代の基盤像を探している

伝えるべき一言
Hyper-Aetherは、
AIが計算構造を生成・評価・再構成することで、
従来のコンピューティングを
“意味駆動の基盤”へ進化させる提案です。

研究紹介にも、事業コンセプト説明にも使いやすい構成