rebellions_

物理的な限界を超えるAI：ATOM™でCosmosを実現する。

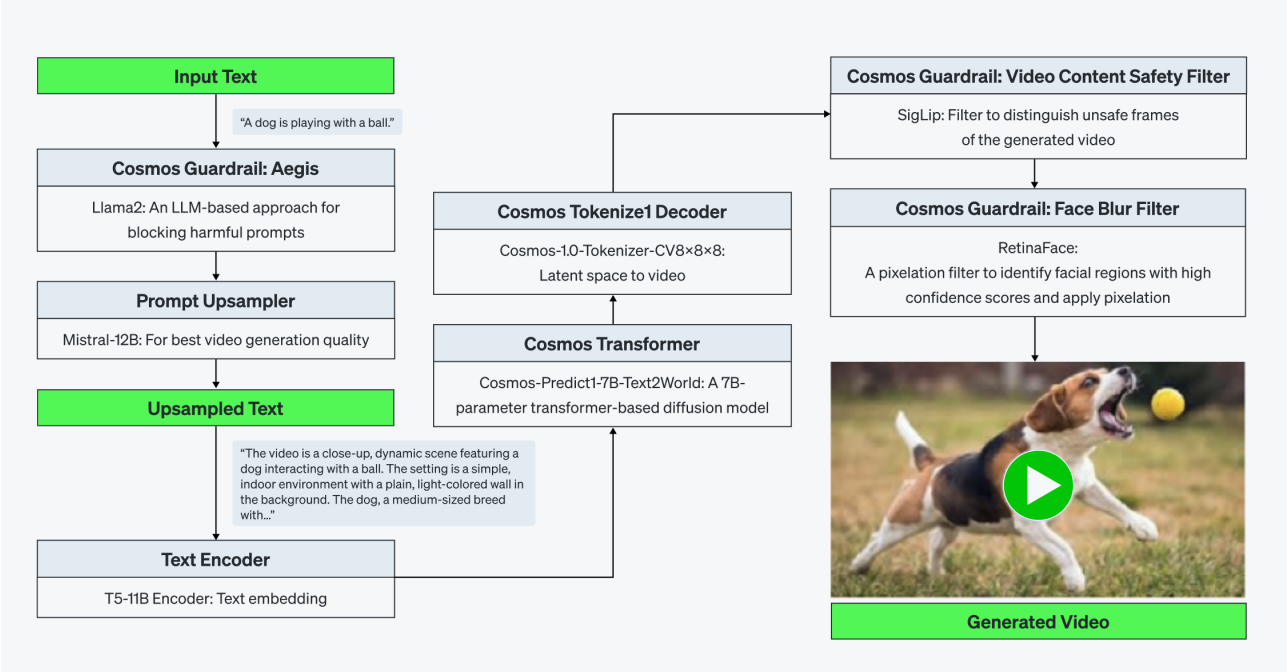
5月 19, 2025

ルチモーダルAIはインフラの境界を塗り替えています。もはやモデルは、一つの入力形式だけに限定されません。テキストや画像、動画など、さまざまな入力が同時に統合される世界―その最前線にCosmosがあります。

Cosmosはdiffusion基盤のText-to-Video生成モデルで、トークナイザー（tokenizer）、事前に学習されたワールドファウンデーションモデル、ガードレール（guardrail）システムなど5〜6個の独立したAIサブシステムで構成されています。各サブシステムはアーキテクチャ上の複雑性と演算特性を持っています。

これまでは、この大規模なワークロードを高性能なGPUプラットフォームだけが対応できました。これはGPUが最適だからではなく、他に代わりになるものがなかったからです。GPUは強力な汎用加速器ですが、 固定化されたソフトウェアスタックと硬直したメモリ階層に依存しています。そのため、新しいモデル構造への対応が遅く非効率的なうえ、時には過度な手作業が必要になります。

しかし、Cosmosのコアは単なるサイズではありません。この構造的な多様性と動的なシークエンスにあります。したがって、効率的な実行は単に演算速度によるものではなく、システム全体の適応性の問題といえます。これがまさにリベリオンが解決しようとした課題なのです。



[Figure 1. Cosmos-Predict1-7B + Cosmos Guardrailの図式]

新しいパラダイム：不可能を可能にする

リベリオンは商用NPUで初めて、Cosmos-Predict1-7Bをリアルタイムで駆動しました。これはただパフォーマンス改善に止まらず、 本質的に柔軟で拡張可能なスタックを基盤としたからこそ実現できたものです。そして、この統合は決して一時的なイベントではありません。

イノベーションの構造：フルスタック性能の結晶

CosmosがATOM™上で実行できた理由は、コンパイラからランタイム、シリコンまでスタックの全階層がアーキテクチャの多様性に対応できるように設計されたからです。他社が一つのモデルだけに最適化していたのに対し、リベリオンは多様なモデルを同時に受け入れられる高い柔軟性を備えました。この柔軟性は後から追加（add-on）するのではなく、すでに設計に組み込まれた構造的な属性です。

AIモデルは絶えず進化しているだけに、それに合わせてインフラも共に進化すべきです。GPUシステムは依然として強力ですが、レガシーツールチェーンと分離したソフトウェア階層、特定ベンダーへの依存（vendor lock-in）で柔軟な対応が難しい状況です。

そのためリベリオンは、システムレベルのアプローチ（System-level Approach）を採用しました。コンパイラ、ランタイム、ハードウェアを全て自ら開発することで、モデルがインフラに合わせるのではなく、インフラがモデルに合わせて進化するプラットフォームを構築しました。

この哲学こそがCosmos on ATOM™の実現につながり、さらなる拡張を可能にした中核的原理です。

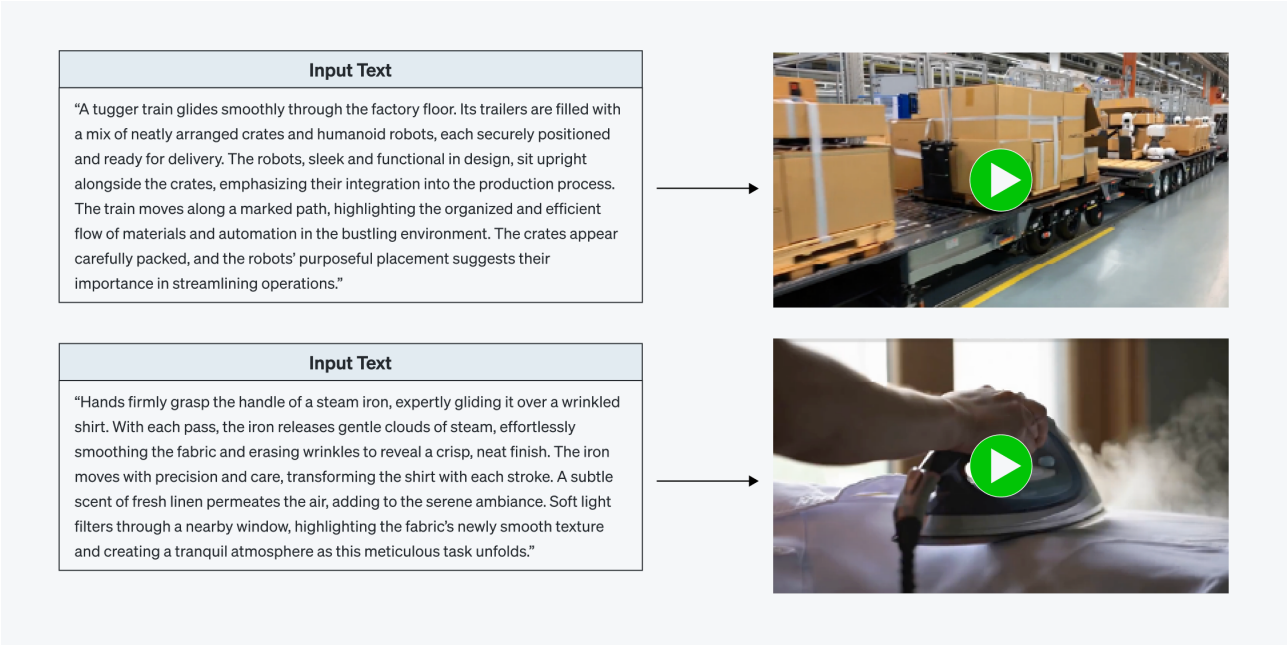
[Figure 2. リベリオンのフルスタック]

ソフトウェアのイノベーション：適応性向けのデザイン

- **モジュール型コンパイラスタック:** 適応性はモジュール性から始まります。リベリオンのコンパイラは200以上のモデルに対応する豊富な演算プリミティブライブラリーを備えており、トランスフォーマー（Transformer）、コンボリューション（畳み込み）、ディフュージョン（Diffusion）モデルを全て処理できるフロントエンド、さらにハードウェアの最適化コードを自動生成するバックエンドで構成されています。個別にチューニングをしなくても、モデルごとのカーネルを効率的に変換できます。
- **統合ランタイムのオーケストレーション:** モデルごとに実行パターンが異なります。リベリオンのランタイムはモデルの動作特性を分析し、メモリ・演算・カーネルの実行パスを動的に調整します。これにより、Cosmosのようにレイヤ別に負荷が変わるワークロードでも最大の稼働率を維持できます。また、Predictive DMAのスケジューリングを採用し、データを「必要になる直前」に予めロードすることで、非同期・多段階のワークロードでも演算遅延を防ぐことができます。
- **拡張型の通信インフラ:** 拡張はもはや選択ではなく必須です。「Rebellions Scalable Design」は高帯域幅の通信階層とテンソル並列実行により、マルチカードの推論が可能です。これにより、Cosmosは複数のNPUにわたって分散実行され、レイテンシやボトルネックが起きずに動作できます。

ハードウェアのイノベーション：柔軟性向けに設計されたアーキテクチャ

- **柔軟な演算構造:** ハードウェアはモデルに合わせて動作しなければなりません。リベリオンはATOM™の演算コアを多様な演算子と動的な実行パスに幅広く対応できるように設計しました。これにより、Cosmosのさまざまな演算パターンをカーネルの再作成なしでもそのまま実行できます。
- **メモリー演算の同期化:** 現代のAIモデルは線形的に演算しません。リベリオンはランタイムの動作を追跡し、メモリフローをリアルタイムで調整するメモリー演算の調整ロジックを開発しました。これにより、Cosmosのような非定型・バースト性の演算段階を別途の特殊なエンジニアリングなしでも処理できます。
- **高帯域幅のインターコネクト基盤のマルチカードスケーリング:** NPU間の拡張のために、システムを最初から設計し直す必要はありません。リベリオンは独自に開発した高速インターコネクトIPを通じて、テンソル並列ワークロードでも低遅延・高処理量の通信を実現できる設計となっています。Cosmosのような複合構造のモデルに対してもボトルネックを生じさせることなく大規模スケールでの同時実行が可能です。



[Figure 3. プロンプトから映像に：ATOM™でのCosmos-Predict1-7B推論]

推論の結果を見る

結論：新しいAI時代の始まり

Cosmos on ATOM™は単なる技術デモではありません。「システムがモデルに適応すべきだ」はリベリオンのアーキテクチャへの哲学が実現された事例といえます。リベリオンはこれまで一つのモデルのためのチップは設計しませんでした。これから登場するあらゆるモデルに対応できるプラットフォームを設計してきました。モデルの複雑度が爆発的に増える時代―未来を切り拓くのは、迅速に適応し、柔軟に拡張し、高度な実行を実現できるインフラです。リベリオンは、まさにそのインフラを構築しています。Cosmosはその第一歩に過ぎません。