スライド集

[知働化研究会誌第2号] **採加働化研究会**

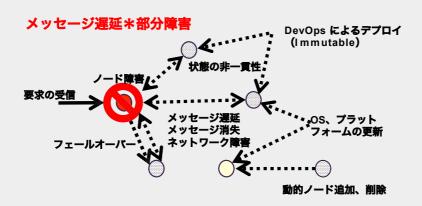
地球規模のクラウド競争の勝者を決める 技術とは何か

Microsoft Corporation Architect Masayoshi Hagiwara

アジェンダ

- クラウドにおける技術課題(おさらい)
- ●技術革新の方向性: これからの競争力
 - 要求の順序化を減らす
 - データ同期の緩和、協調を減らす
 - 正しさの基準
- 将来のクラスタ管理(私見)

クラウドにおける技術課題(おさらい)

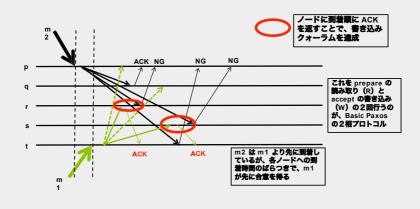


技術革新の方向性:これからの競争力

技術革新の方向性: これからの競争力

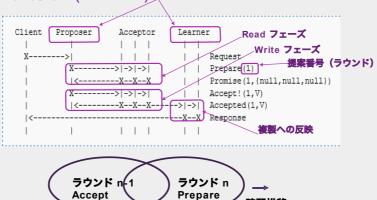
- これまで:
 - 分散並列処理: 負荷分散によるスケーラビリティの改善
 - ・並列によるスループットとレイテンシーの改善
 - 一括(バッチ)処理: スループットの改善
 - ・パイプライン(非同期)処理: レイテンシーの改善
- ・ これから:協調と同時実行制御の改善
 - ・要求の順序化を減らす: 全順序から半順序へ
 - ・データ同期化を減らす: 常時一貫から適宜一貫(同期の緩和、協調を減らす)へ

コミットの順序化: クォーラム



コミットの順序化: 耐障害性を持つプロトコル

• 要求の全順序化 (Paxos) 編製 (状態マシン)



時間推移

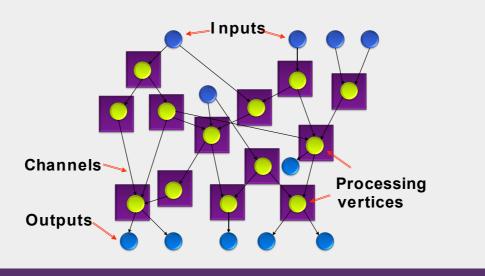
半順序で動く現実



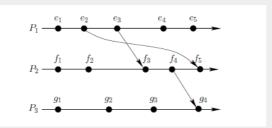
http://similar-image.com/photo/detail/876

**Executable Knowledge and Texture Laboratory

DAG (Directed Acyclic Graph)



Happened-before 関係



・ 同一メッセージの送受信: e3⟨f3

よって、e1⟨g4

では、e2 と f2 の順は?

同時実行の問題

・3つの同時実行トランザクション

T1: BEGIN A+=100, B-=100 END // 引出し、預金

T2: BEGIN C = A+B END // 残高合計

T3: BEGIN A=1.06*A, B=1.06*B // 利子計算

. .

同時実行の正しい順序を決める基準

どのスケジュールが正しいか

```
S1 T1: A+=100, B-=100
```

A=1.06*A, B=1.06*B

S2 T1: A+=100,

T3:

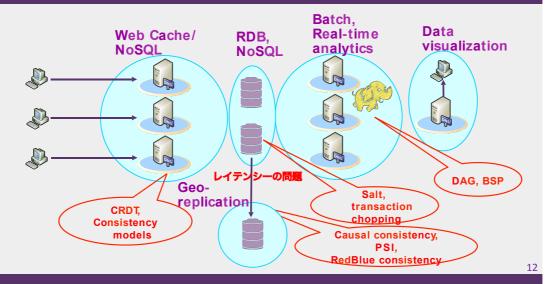
B-=100

T3: A=1.06*A B=1.06*B

T1: A+=100, B-=100
T2: C=A+B

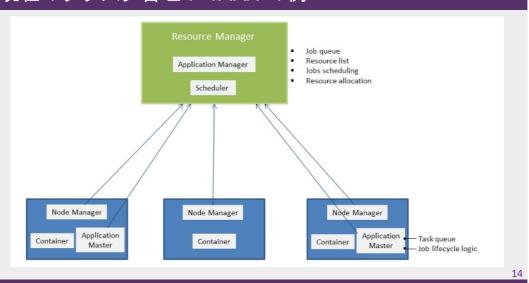
S4 T1: B-=100, A+=100 T2: C=A+B

データ同期の緩和、要求の半順序化



将来のクラスタ管理

現在のクラスタ管理: YARN の例



大規模クラスタ管理の現状

"Systems at a certain complexity start demonstrating emergent behavior, and it can be hard to know what to do with it"

Google's cloud chief Peter Magnusson

15

ネットワーク帯域の現状

「データセンター内部の200Gビット/秒や300Gビット/秒は、スパコンから見ればおもちゃみたいなもので、1000倍ぐらい違う。グラフが大規模な場合は処理時間の7割ぐらいがネットワークで通信している時間なんです。ネットワークの速度が1000分の1になってしまったら、処理の半分がネットワークだとすると、もういきなり処理速度が500分の1になってしまうんですね。だから(IDCでは)使いものにならないんです。」

東京工業大学 松岡聡教授

ワークロード毎の SLA を保証するクラスタ管理 Metro Network まとめ 地球規模のクラウド競争力 大規模クラスタ管理 ・予見可能なクラスタ管理 リソース管理、スケジュール管理 ・ネットワーク高速化 YARN, Mesos, Omega, Kubernetes これまで これから クラウド対応でスケーラビリ ティ、スループット、レイテ ンシーを改善 データベースと分散システムの融合 協調と同時実行 制御の改善 分散並列処理 要求の順序化を減 らす バッチ処理 ・データ同期の緩和 Pipeline 処理 分散システムの非同期性 _ 動的変更の許容 OSのみ必要な(ツールを保有している) ・OS、プラットフォームの更新 ・DevOps によるデプロイ クラウドにおける課題 FLP 定理、CAP 定理 ・ネットワーク遅延 Resilient, Scalable, Secure Architecture



本書に記載した情報は、本書各項目に関する発行日現在の Microso代 の見解を表明するちのです。 Microso代 は絶えず変化する市場に対応しなければならないため、ここに記載した情報に対していかなる 資 務を負うちのではなく、提示された情報の信意性については保証できません。 本書は情報提供のみを目的としています。 Microsoft は、明示的または暗示的を開わず、本書にいかなる保証も与えるものではあ りません。すべての当該著作権地を選守することはお客様の資務です。 Microsoft の書面による明確な許可なく、本書のMicrosoft は、明示的または暗示的を開わず、本書にいかなる保証も与えるものではあ りません。すべての当該著作権地を選すいるとはお客様の資務です。 Microsoft の書面にはる明確な許可なく、本書のMicrosoft は、体書の内容を分といっても、転替や機等システムへの格謝または排入を行うことは、とで が表式または手段(電子的、機械的、第写、レコーディング、その他)、および目的であっても禁じられています。これらは著作権保護された権刺を制限するものではありません。 Microsoft は、本書の内容を保 語する特許・特許出租書。高様、著作権、またはその他の知的財産権を保有する場合があります。 Microsoftは、ら書面によるライセンス契約が明確に供給される場合を除いて、本書の提供はこれらの特許、商 標、著作権、またはその他の知的財産へのライセンスを与えるものではありません。

○ 2015 Microsoft: Corboration All rights reserved.

Microsoft, Windows, Visual Studio は、Microsoft Corporation の米国およびその他の個における登録商標または商標です。その他、記載されている会社名および製品名は、一般に各社の商標です。

