

分散システム基礎と クラウドでの活用

第7回：まとめ

国立情報学研究所

石川 冬樹

f-ishikawa@nii.ac.jp



本講義の位置づけ(第1回より)

- 分散システムの難しさ(重要さは言わずもがな)
 - 相互運用性からセキュリティまで, 多様な側面
 - それらの間のトレードオフ(特に, 複製管理における性能, 一貫性, および耐故障性・可用性)
- ➡ これまで累積された知見の活用に向けて
 - 教科書的な基礎知識・技術の習得
 - 様々な種類の「一貫性」など, 達成する性質の, 厳密な定義および実現のための原則や方法
 - クラウドの構築・利用などにおけるそれらの役割や活用方法に関する議論



既存の基礎知識・技術

- 数十年にわたり盛んに取り組まれており、また実用されているため、スタート地点とできる
 - パターン(カタログとして整理はされていない?)
 - プロトコル
- 理論上は簡単に見えることでも、十分難しい
- ミドルウェア・フレームワークの頑張りどころ
 - 例: JGroups
 - 最近クラウドのためにも多数実装
 - 当然のように一貫性などの説明文句や、選択肢が出てくるため、理解が必要



最近の状況(第1回より)

- クラウドでは、スケーラビリティ・性能と可用性を重視し、一貫性(整合性)が弱いとも言われる
 - 必要以上に一貫性を保証することはない、という議論で筋が通っている(要求によっては)
 - 何かと議論されているようだが、クラウドの利用(や構築)にあたって、「一貫性(整合性)」などを、**受講前より正確に理解している？議論できる？**
 - 例：「これからはACIDではなくBASEだ！」
 - 例：「CAP定理を踏まえての当然の妥協だ！」



本講義(担当講師)のスタンス(第1回より)

- 「基礎(理論や技術)」を学んでみましょう
 - 既存の知見(特に設計)を再利用して解決できるに越したことはない
 - 技術者として, 正確な定義や分析, 議論ができるよう体験すべき(+ 常識として知っておくべき?)
- その実用的な活用是非や方法を議論しましょう
 - たいてい, 現場にはオーバースペック?
 - ときどき, 重要視する観点が現場と異なる?
(が, 「良さ」の基準や理想のあり方を示すよい道しるべになる, よいスタート地点・たたき台になる)



扱わなかったこと(1)

- 広い話題の中で一貫性, 同期, 耐故障性に対する基本的な考え方に集中
 - セキュリティ
 - すでに広く使われているシステムに関する具体的な知識・技術
 - 分散ファイル, 分散オブジェクト, Webクライアント・サーバなど
 - より特定の領域に踏み込んだ, 具体的で再利用しやすい知識・技術



扱わなかったこと(2)

- あくまで基礎理論の中の，直観的な考え方
 - 実際に証明する，といったことはしていない
 - 非常に微妙にうまく成り立つような，理論上の限界を追い求めるような仕組みには触れていない
 - 理解，実装，デバッグ，確信できない
 - 安全が確保できる荒いが単純な方法が現実的
 - 「基本 * * 試験」とかに出るような概念までは戻っていない(1台のバックアップの運用法など)
 - 実際の開発・運用で考えなければならないことはたくさん



扱わなかったこと(3)

■ 真のトレードオフの追求

- 性能差の実感や, 検討的な調査と議論の流れまではやっていない
(「実線演習」講義では簡単にR/Wした?)

- 理論上ありがちな議論については, 語句簡単なものを課題としている

■ 大規模分散については「伝聞」

- Amazon, Google, Facebook, Twitterなどの中の人たちにしか見えていない世界はあるはず
(見る必要があるかさておき)



ソフトウェア工学との関連

- モデル検査やそのための形式言語は、最も関連性がある
 - 様々な「可能性」や「タイミング」を、網羅的に、見合うなら十分厳密に検討しようという考え方
(本講義のように、「争点絞って人手レビュー」
だとしても)
 - ただし、たいていの場合、「必要な仮定」もしくは「例外的に満たせない状況」が存在する
(モデル検査器としては「成り立ちませんでした・判例があります」)



ソフトウェア工学との関連

- このため，仮定や結果の妥当性確認・評価の方が非常に重要
 - 滅多に起きないことを，定性的に説明，または実証できる？
 - 実証試験とその統計分析
 - (それらを踏まえた)確率シミュレーション，確率モデル検査など
- アプリケーションの性質上，必要性や優先度が低いと言える？



ソフトウェア工学との関連

- 明らかに, ドメイン固有, または特定の(非機能)要求固有のパターンが多く積み上げられている
 - オブジェクト指向のデザインパターンほど, 系統的に整理した形ではあまり見ないかもしれない(カタログ化, フォースの明記などはなく, 本講義程度の列挙, パターンとは呼ばない?)
- 自分たちで作って使うのは本当に一苦労
- 提供されているもの(の無数のオプション)を理解して使うだけで一苦労



評価

- 出席（講義内にめいっぱい考えましたか？）
 - 欠席の場合には，講義内容や講義内演習に関するレポートまたは講師とのやりとり
- レポート課題1つ
 - 前回の範囲でできる古典的・教科書的な問題
「特定の知識が要らない，比較的簡単なものをじっくり考える機会をとってみよう」
なので，講義関係なくおそらくできる
(10/22 ✕ 切)