

対立する概念!?

Works Applications

Executive Fellow

小松 宏行

技術者の会話は対立する概念を軸に 繰り広げられている

- Open System と Microsoft
- Emacs と vi
- RISC と CISC
- 分散と集中
- 疎結合と密結合
- ギークとスーツ
- Enterprise と Consumer

で、まず分散と統合

- 並列コンピューティングの歴史は古い
- Von Neumann アーキテクチャと分散の困難
- しかし、多くの人で使いたい→Time Sharing
- ネットワークとPC
- P2P
- 技術者は分散を好み、利用者は統合を好む
- 分散と統合の裏にあるのは管理

Client と Server

- なにが Client で Server か？
- X11
- Client/Server 型の Programming
- ≡ Application/Database

Internet と多層アーキテクチャ

- Multi-tier architecture
- ≡ 入出力—Application—Database Model
- PC は 入出力のみ！？
- Server side から Client への引き戻し

クラウドを軸に...

- 情報空間が統合されていると全員にとってのメリットが大きい
- どのように情報空間を統合するか？
 - 一社独占型
 - Semantic web
 - ID

Semantic Web を軸に...

- 人為的な tag 付け
- 対概念としての Data Mining
- Hybrid は当然可能
- 対立の危険性は Data Mining の誤解釈か？

Programming style への影響？

- Data Oriented
 - RDB vs Object
 - RDB のメリットって？
- Cache Server
- Web, Application, Cache, Database

番外編(1)パフォーマンスと可読性

- どっちをとるか？

番外編(2)フリーランスと 会社員プログラマー

- どっちがおいしいのか？

番外編(3)顧客と研究開発

- 正しいのはどちらなのか？