

Will Decentralization Drive Event-Based Architectures?



Rohit Khare

Founder & CTO, KnowNow

Doctoral Student, UCI

ISR Research Forum

18 June 2002

A Tale of Three Markets

How do multiple traders set prices?

- NYSE: one human, a 'specialist', aggregates all data
- NASDAQ: many 'dealers' quote their own prices
- FOREX: every trade is settled pairwise (unregulated)

There is a spectrum here:

- Centralized: single immediate arbiter in space-time
- Distributed: multiple arbiters with timeouts
- Decentralized: no arbiters at all

Are There Any Decentralized Systems Yet?

- Current architectures for Internet-scale software *assume consensus is feasible* -- RPC, Dist. Objects, &c
- However, consensus is not feasible on an asynchronous network with even one failure
- The continuing evolution of decentralized computer architectures points in that direction
 - *A continuum of embedded systems from 1 meter to 10^9 meters apart*

I claim that we do not yet have effective software architectures for coordination without consensus

Events are Well-Suited to Decentralization

I only have three clues to share from my investigation so far:

- **Physics: one-way, best-effort reflects real limits**
 - ┆ *All the other sorts of Message-Oriented Middleware (MOM) are effectively end-to-end protocols, just as TCP is run on top of IP*
- **Protocols: concrete contracts across organizations**
 - ┆ *The useful part of Web Services is its black-box abstraction of software as nodes on a network, allowing us to model interaction as application-layer messages*
- **Proxies: dynamic extensibility for many agencies**
 - ┆ *Separate event router proxies enable 1st parties and 3rd parties to add ilties such as security, reliability, and interoperability without modifying services.*