Unity and Interoperability Among Decentralized Systems

Chris Gebhardt

The InfoCentral Project
https://infocentral.org
Users, Devices, I/O

Software Layer
(dynamic, largely declarative)

Global Information Graph
(immutable data entities replicated among networks and repositories)
Project Inspirations..

- Information-Centric Networking
- Semantic Web
- Pervasive Computing
- Programmable UIs
Project Philosophy

1. Decentralize Information
2. Let software and networks evolve around it
3. Same data, Compete on QoS
Unifying Architecture

Multi-disciplinary approach..

- Data Architecture
- Information Architecture
- Software Architecture
- Network Architecture

Prioritize clean separation of concerns..
InfoCentral Persistent Data Model

- Everything is immutable
- Reference only via cryptographic hash
- Payload encryption and signing
- Versioning and annotation
InfoCentral Standard Data Entity

- header
- payload blocks
  - self metadata
  - external metadata
  - content
- signatures block
Data Entities Hash-reference Other Entities

secure hash references

standard entity

header

payload blocks

self metadata

external metadata

content

signatures block

standard entity

header

payload blocks

self metadata

external metadata

content

signatures block

standard entity

header

payload blocks

self metadata

external metadata

content

signatures block

standard entity

header

payload blocks

self metadata

external metadata

content

signatures block
Networks Collect and Propagate References

Immutable Global Data (Independent, Stored Anywhere)

Mutable Local Data (Discovered, Disposable)
Known hash aliases for an entity:

- Entity Hash Value SHA2-256
- Entity Hash Value SHA3-512

HID = Hash ID
Network services compete on QoS

- Raw data entity storage
- Entity hash dereferencing
- Reference collection, propagation, and notification
App-Free Computing: No bounds

- **Applications**
  - Static, pre-designed, pre-packaged interactions
  - UI concerns baked in
  - User removed from backing data

- **Interaction Patterns**
  - Contracts for interaction over *open graph data*
  - Dynamic local-rendered UI – not pre-designed
  - User may work close to the data
Open Collaboration over Graph Data

My Restaurant

Menu Root

Menu Items

Comment

Rating

Menu Item

Interaction Pattern

Current Queue

Request

Approval

Customer Details

Customer

Answer

Round

Trivia Game

Interaction Pattern

* Arrows point in the direction of reference
No more fragile APIs..

Shared semantic data is the universal interface for composing systems.

All users and software operate over the immutable (append-only) data graph.
InfoCentral Core Components

Persistent Data Model

Data Entity

Data Entity

Data Entity

New Entity

DMF Components

Repository

Intermediate Indexes & Query Engines

Cryptography

Information Centric Networking

Entity Generation

selective ingestion

selective ingestion

Local UI Frameworks

Userspace Data Structures

Userspace Code

Interaction Patterns

Data Management Foundation

Information Environment
Early Applications

- Collaborative hypertext
- Document & media management
- Archival systems
- Scalable structured discourse
Q&A
Differences w/Semantic Web?

- **Same**: Most of abstract semantic graph model **but**..
- **Different data model**:
  - SW: named mutable documents (HTTP)
  - IC: hash-only immutable entities (agnostic)
- **Different network assumptions**:
  - SW: Host-based, Web APIs
    - ex. notification of doc update by push API
  - IC: Content-based, Direct interaction over graph
    - ex. PubSub notification of new references
Controversial Architectural Positions

- No human-meaningful naming of data
  - secure hash references only
  - no paths
- Considered harmful:
  - filesystems
  - embedded markup (use external annotations)