Evolving a Best-of-Breed Framework for IoT

Greg Burns - Chief IoT Software Technologist

IoTivity ⟷ ALLJOYN

Same Goals:

- Standardized wire protocol
- Standardized schema definition
- Standardized data models
- Transport and OS independent
- Collaborative development
- Open source and freely available
- Proximal discovery

Feature Comparison

IoTivity

- RESTful + Notify
- JSON payload
  - CBOR serialization
- Transport
  - UDP (IPv4 and IPv6)
  - UDP multicast
- CoAP service discovery
- Schema definition:
  - JSON (data)
  - RAML (interaction)

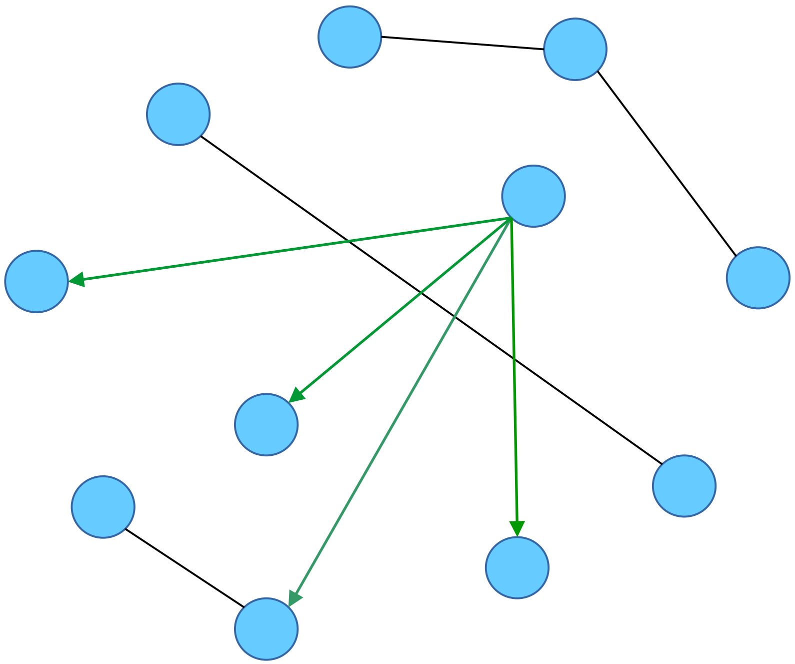
ALLJOYN

- RMI + Pub/Sub
- Binary payload
  - DBUS serialization
- Transport
  - UDP and TCP (currently IPv4)
  - Serial, and BTLE (experimental)
- MDNS service discovery
- Schema definition:
  - XML

Network Topology

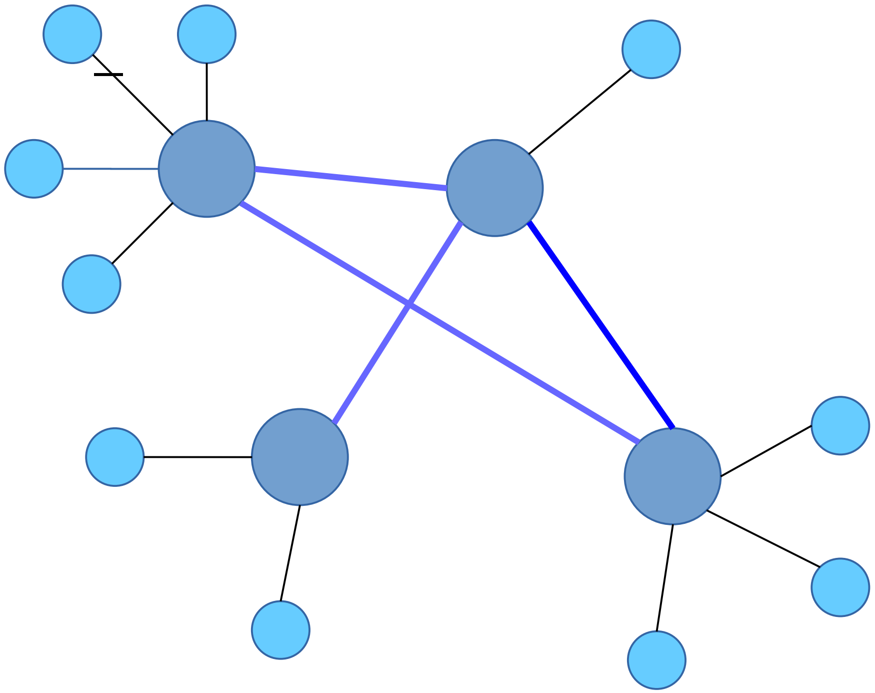
IoTivity

Point-to-point
Point-to-multipoint



ALLJOYN

Mesh of stars



Security

IoTivity

- DTLS link-layer
  - ECC, AES, X509
- ACL permissions
  - Resource, interface, wild-card
  - CRUDN
- SubjectId, RoleId

ALLJOYN

- Application-layer
  - ECC, AES, X509
- ACL + Capability
  - Interface, object, property, method
  - Provide, Observe, Modify
- Membership certs

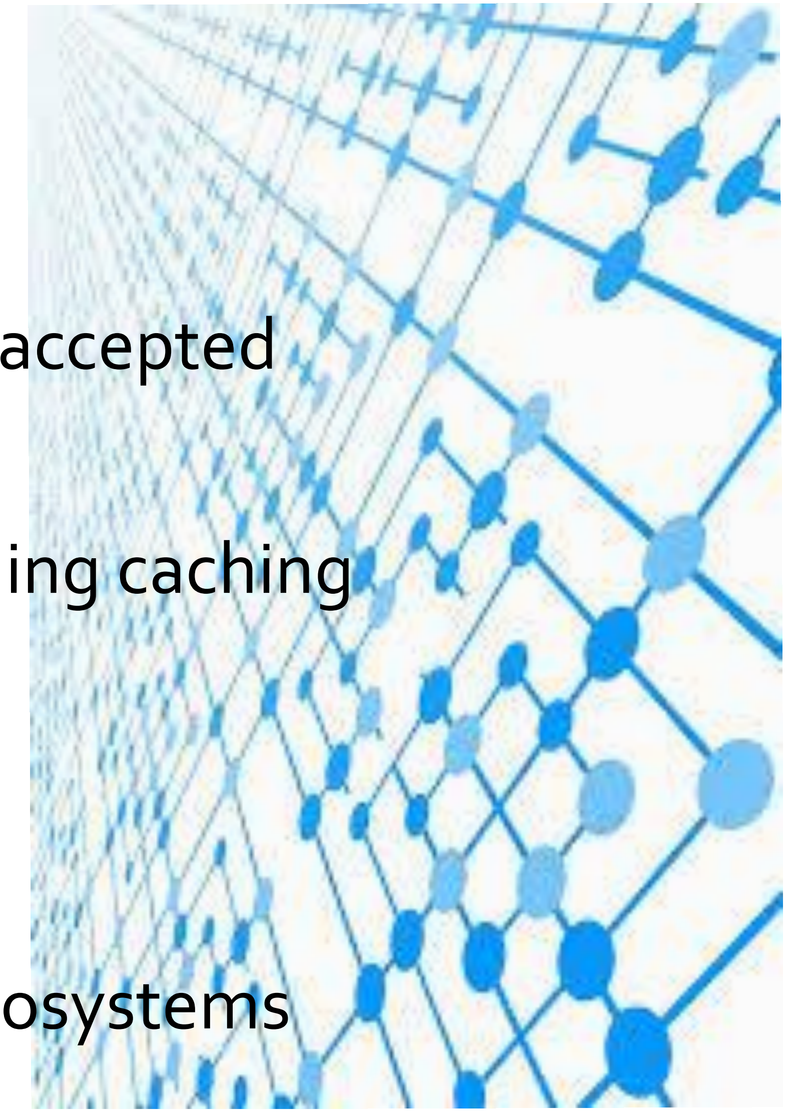
How The Concepts Map

IoTivity	ALLJOYN
URI	Bus-name + Object-path
Resource	Object
Resource type	Interface
Interface	N/A
N/A	Introspection
Collection	Child object
Link	Object path (local only)
Observe	Add-match

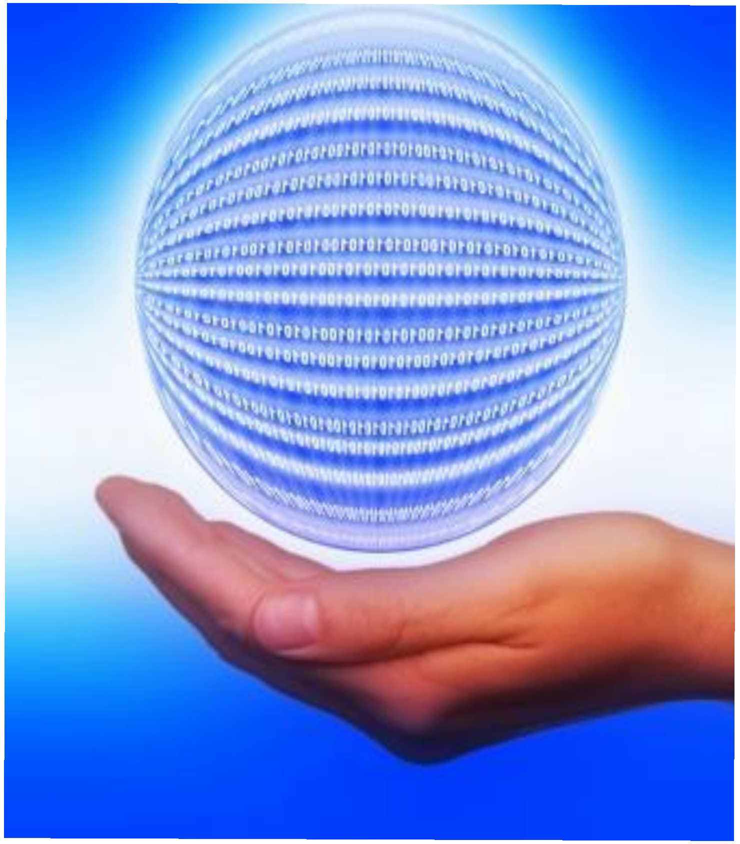
Putting It All Together

Programming Model

- RESTful
  - It works
  - Is broadly adopted and widely accepted
- Publish/Subscribe
  - Extension of “observe” leveraging caching
  - Support “sleepy” nodes
- Operational mapping
  - Can be achieved by POST
  - Facilitates bridging to other ecosystems



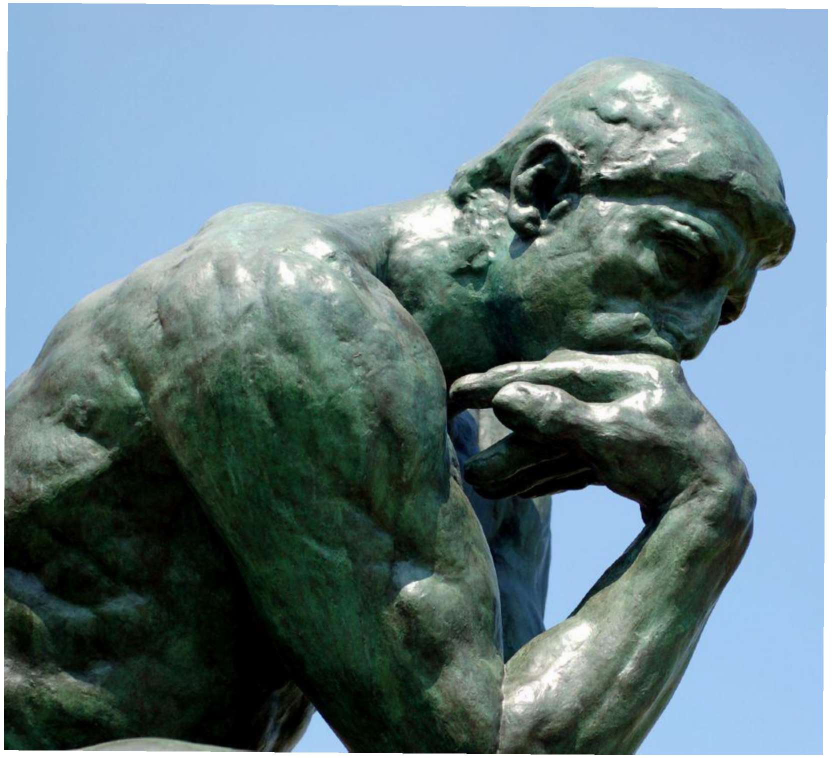
Extended Schema & Hi-Fi Serialization



- CBOR
  - Compact, expressive, extensible
  - Supports streaming
- Schema support for data types
  - Need unambiguous serialization
  - Multiple numeric types
  - Structures, typed arrays, etc
- Supplemental meta-data
  - Field names, enumeration values for code generation
  - Human readable strings for if-this-then-that type rules

Support For Introspection

- Resource 'knows' its own data model
  - Always available
  - Always correct
  - Returns it on request
- Multiple uses:
  - Code generation
  - Run-time data validation
  - Access rights checking
  - Test case generation
  - User interaction
- Needs Evolved Schema Definition
  - JSON schema definition not expressive enough



Extend Security End-to-End

- Application level authentication & encryption
  - Link-layer for additional privacy if required
  - Full transport independence
  - Endpoints can use different transports
  - Trust relationship extends over bridges and gateways
- Must support sleepy nodes and store-and-forward
  - Content stored/cached encrypted and authenticated
- Should leverage existing standards work
  - JOSE (JSON Object Signing and Encyption)
  - COSE (CBOR Object Signing and Encryption)

More Flexible Access Control

- Resource and Interface level ACLs
  - Interface level granularity is very flexible
  - ACLs stored on server device or in security manager
  - CRUDN captures access modes
  - Perform access rights checks in the framework
- New capability-based access rights
  - Client device presents signed manifest
  - Takes storage/lookup burden off server device
  - Facilitates delegation, membership, etc

Add Middleware Functionality

- Current IoTivity APIs
  - Relatively low-level
  - Application developer is responsible for data validation
- Middleware functions:
  - Validate all incoming requests against local data model
  - Validate all outgoing requests against remote data model
  - Automate access rights checking on incoming requests

Q & A