A RESTful, Distributed and Enhanced ICN System for IoT

Lijun Dong & G.Q Wang

Huawei Technologies

Outline

- **Benefits for IoT with ICN**
- **RESTful system of ICN for IoT**
  - RESTful method
  - Resource naming and naming translation
- **Distributed management and routing of large amount of IoT resources**
  - Distributed resource directory
  - Reasonable FIB size
- **Enhancement of in-network capabilities of ICN for emerging IoT scenarios**
  - Context-awareness resource acquisition
  - Event-driven notification
  - More…

How ICN Benefits IoT

- Can ICN benefits IoT use cases and how?
  - https://tools.ietf.org/pdf/draft-zhang-icnrg-icniot-requirements-01.pdf
  - Distributed caching and processing
    - Avoiding unnecessary transmissions with IoT devices to retrieve and distribute IoT data to multiple places is important, hence processing and storing such content in the network can save wireless bandwidth and battery power.
    - Applications for IoT networks requiring shorter delays can benefit from local caches and services to reduce delays between content request and delivery.
  - Decoupling between sender and receiver
    - IoT devices may be mobile and face intermittent network connectivity. When specific data is requested, such data can often be delivered by ICN without any consistent direct connectivity between devices.

RESTful Method and Resource Naming

- Methods in RESTful protocols (CoAP): GET, POST, PUT, DELETE, OBSERVE (Subscription).
  - Mapping from the ICN messages for the RESTful protocols is needed.

- Resource Identifier in RESTful architecture is URI, while ICN architectures have different kinds of naming and resolution mechanisms. URI can be one of the naming types in ICN.
  - Should the RESTful architecture be able to support other kinds of naming, how to do the conversion?
    - There can exist name translation overlay to translate between the globally unique URI and the name in the local domain.
  - Or should ICN architecture only adopt URI as one primary naming system?

Is ICN Router Good Candidate for Distributed Resource Directory?

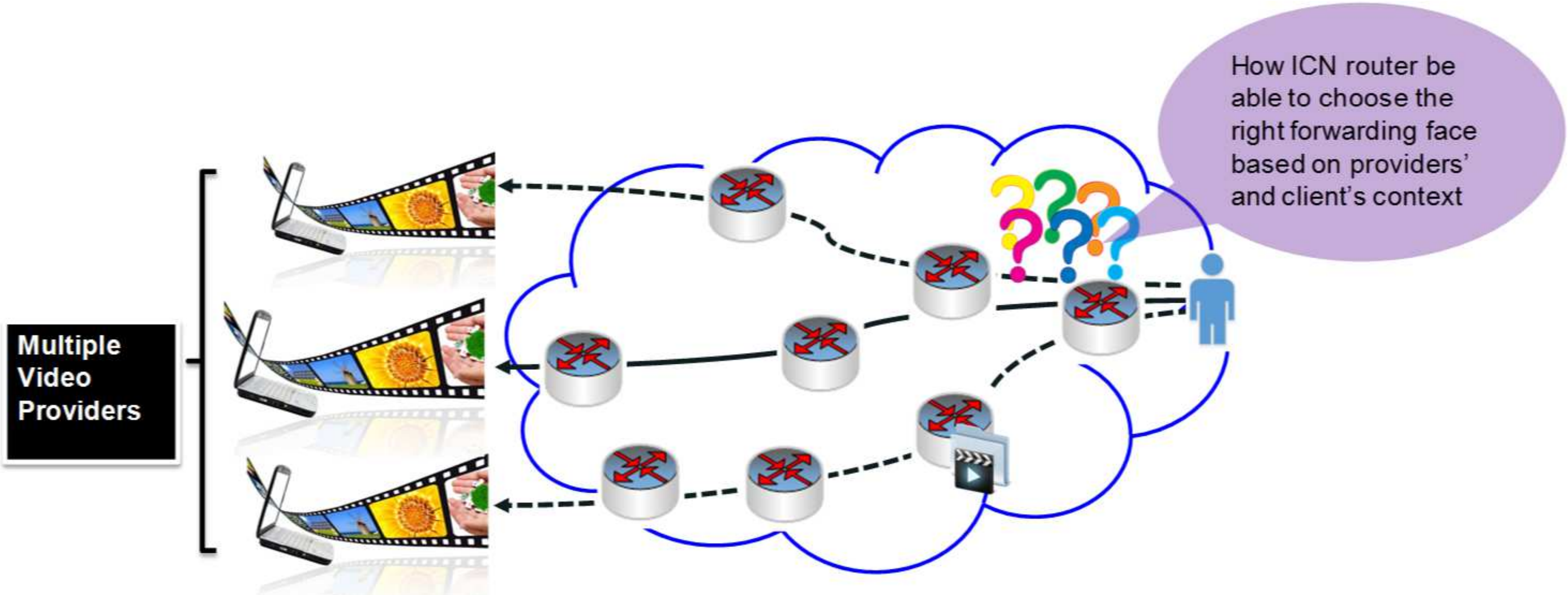
- The IETF Constrained RESTful Environments (CoRE) defines Link Format, which describes the attributes about the resources hosted by endpoints (EP) as well as possible further link relations.
- CoRE also defines Resource Directory (RD), which provides a set of REST interface for EPs to register and maintain sets of RD entries, and for clients to look up resources.
- The lookup interface allows the client to specify the attributes such as endpoint name ('ep"), domain ("d"), resource type ("rt"), endpoint type ("ep"), and link attribute parameters (e.g. "lf").
- Currently RD is centralized, which has the issues such as limited scalability and reachability, large latency, single point failure. Thus, a distributed RD architecture is needed.
  - ICN’s distributed and in-network routing characteristics make it a favorable architecture to implement de-centralized IoT resource registration and lookup interfaces in the ICN routers.

How to Make FIB Size Reasonable for IoT Resources?

- The existing ICN architectures, such as NDN suffer from the explosive number of routing entries based on content/resource. It is not desirable to maintain routing entry for each IoT resource.
  - It is a challenging issue on how to build routing entries such that the in-network IoT resource discovery and forwarding can be facilitated but still keeping the number of routing entries manageable.
  - The lookup interface designated in Resource Directory (RD) gives some key attributes of resources for clients to specify.
    - Can the FIB be managed to have less entries by only maintaining the key attributes of resources, such as endpoint name ('ep"), domain ("d"), resource type ("rt"), endpoint type ("ep"), and link attribute parameters (e.g. "lf").

How to Better Support the IoT Scenarios?

- IoT use cases require more sophisticated in-network capabilities of ICN routers and how ICN protocols and messages are going to support them.
  - Example 1: Context aware IoT data acquisition



Another Example

- Example 2: Conditional event subscription and notification