XMPP-Grid for SACM Information Transport

XMPP Protocol Extensions for Use in SACM Information Transport

http://tools.ietf.org/html/draft-salowey-sacm-xmpp-grid-00

Syam Appala, Nancy Cam Winget

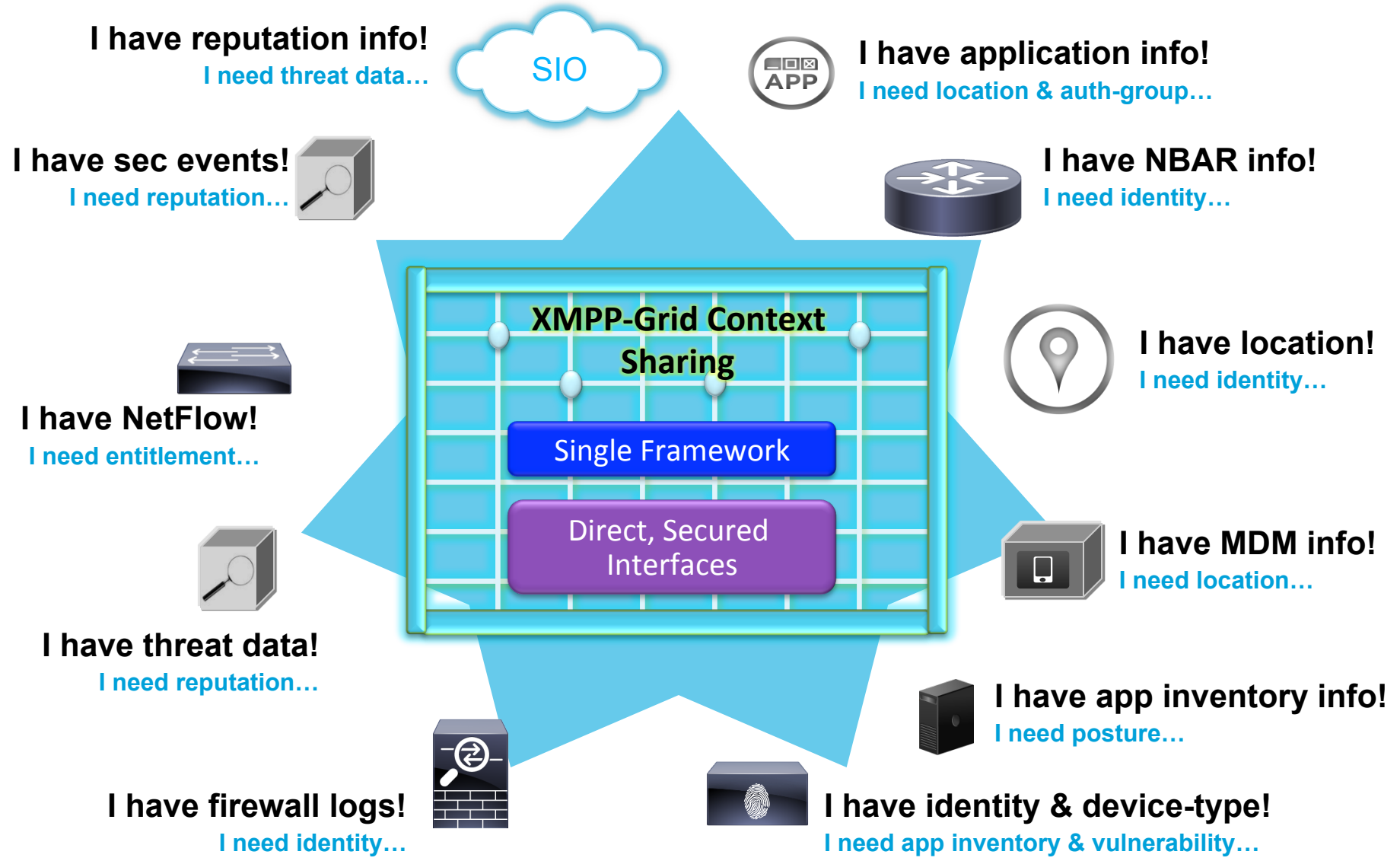
22 July 2014

Agenda

- XMPP-Grid Use-Case
- Design Considerations
- What is XMPP-Grid
- XMPP as XMPP-Grid Transport
- XMPP-Grid Controller & Control, Data Flow Segregations
- Client Authentication & Authorization
- XMPP-Grid Protocol
- Topics & Subtopics with message filters
- IF-MAP with XMPP-Grid

XMPP-Grid

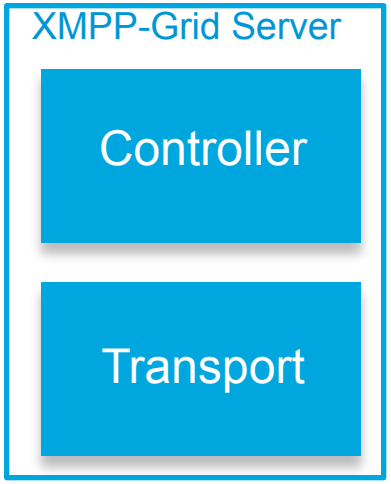
Enabling the Potential of Network-Wide Information Sharing



XMPP-Grid addresses …

- Visibility into *“who is connecting”, “who is accessing what”*
- Centralized, policy-based authorization – *“who can do what”*
- Secure, bidirectional connectivity
- Mutual certs-based authentication
- Flexible consumption APIs – real-time, on-demand, bulk transfer
- Client contextual needs support through semantic, syntactic filtering
- Ability for peers to negotiate out-of-band, secure p2p connection
- Standardize schemas & information models through XML
- Scalable to thousands of nodes
- Platform agnostic

XMPP-Grid Controller Design Tenets



- **Policy-based Authorization**
  - Centralized control for authorization and client management
  - Facilitates secure communication between authorized clients
- **Scalable**
  - Architecture scales to thousands of clients/nodes
  - Provide resilient, high availability support
- **Agile**
  - Enable many different uses across the communication fabric i.e. context, policies …
  - Should be platform agnostic (C/C++, Python, Java …)
  - Negotiation for type of data plane communication & APIs
- **Lightweight Client**
  - Enable adoption through small footprint & intuitive APIs
- **Standards**
  - Enable adoption through standardization of schemas & information models

XMPP-Grid Infrastructure Design Tenets

- **Scalable**
  - Architecture scales to 100K – 1M of nodes/clients
  - Provide resilient, high availability support
- **Reliable**
  - Provide message delivery guarantee
- **Flexible**
  - Support semantic & syntactic filtering to serve contextual needs
  - Support information time sensitivity needs
- **Standards**
  - Enable adoption through standardization of schemas & information models

Why XMPP for XMPP-Grid Transport?

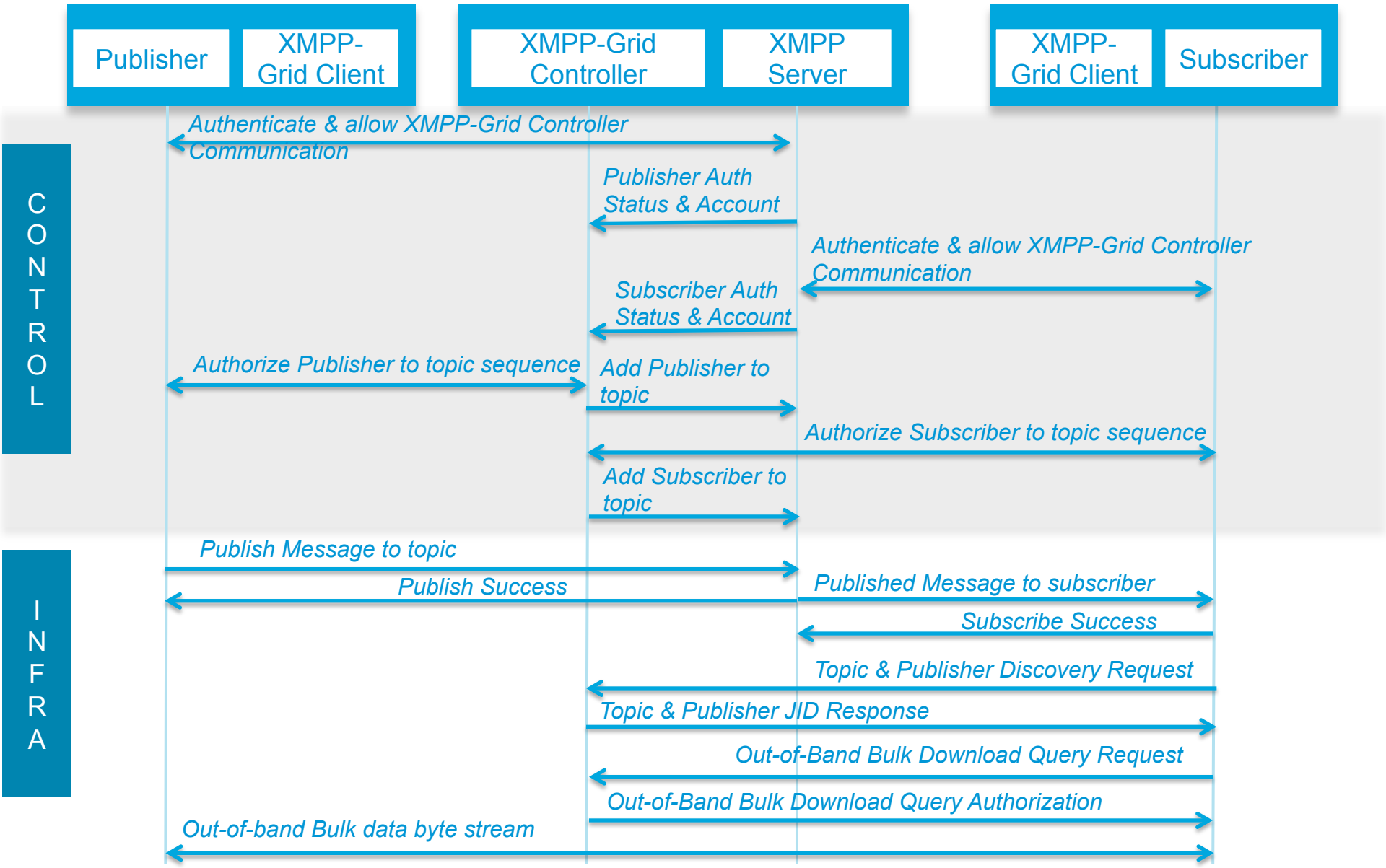
- **Open** – standards-based, decentralized (no single point of failure) and federated architecture
- **Real-time eventing** – using publish, subscribe notifications
- **Security** – Domain segregations; federation support; strong security via SASL and TLS
- **Flexibility** – Custom functionality can be built on top of XMPP; Easily extensible
- **Bi-directional** - avoids firewall tunneling
- **Scalable** – supports cluster mode deployment and message routing
- **Peer-to-peer** – directed queries and OOB file transfer support
- + Presence, service and device capability discovery …

XMPP-Grid Controller

- Plugs-in as external component to the XMPP server

- Responsible for –
  - Account approvals of XMPP-Grid clients
  - Authorization of client actions – subscribe, publish, query, bulk download
  - Topic (information channel with publishers and subscribers sharing a well defined publisher data model) setup with subscription list
  - Maintains directory of topics & topic subscriptions
  - Communicates with other XMPP-Grid controller in cluster for HA
  - Offers interfaces & statistics for management of clients & topics

XMPP-Grid Control & Data Flow



XMPP-Grid Client Authentication

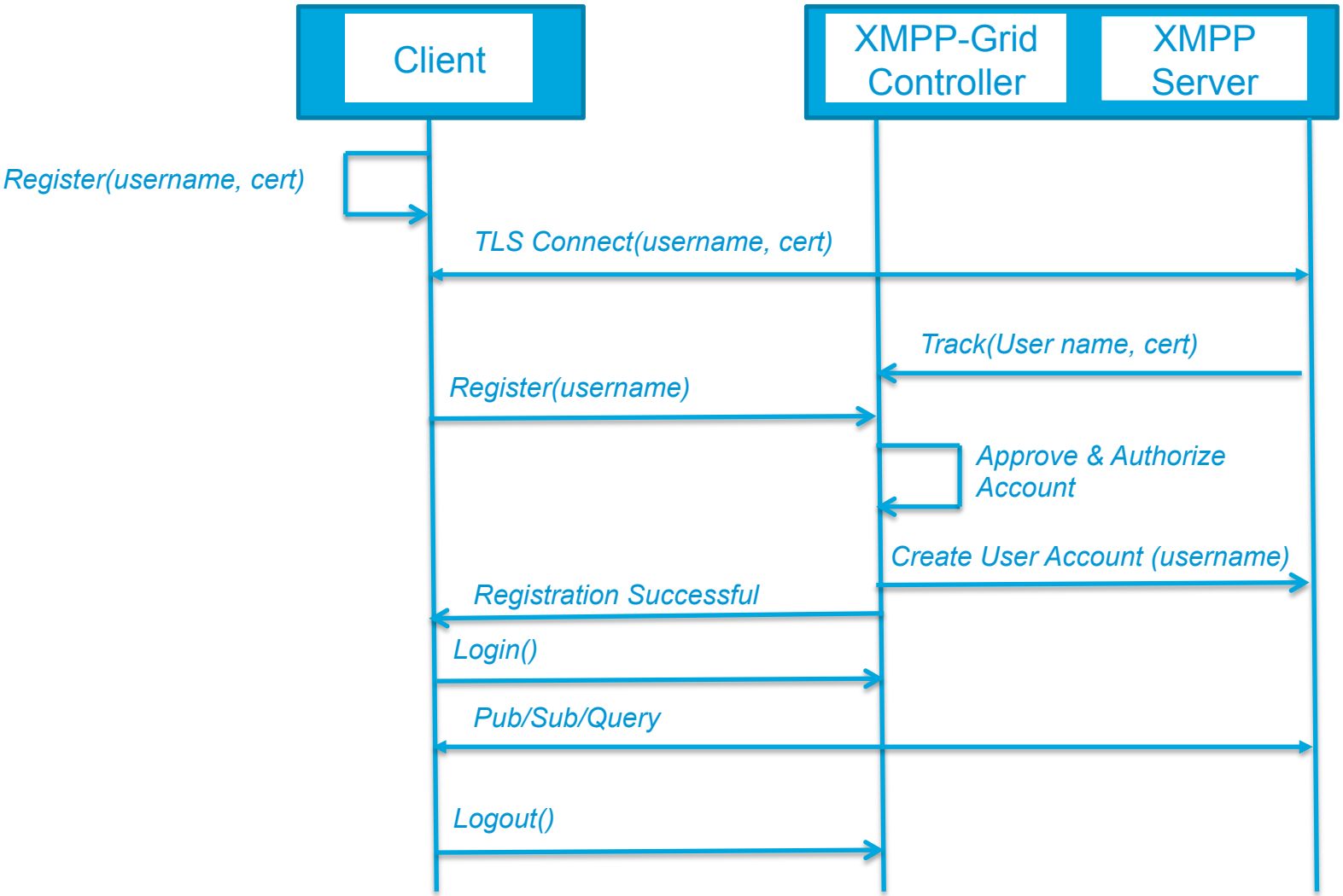
- Each XMPP-Grid client will go through the phases of authentication, registration and authorized access
- Certs-based mutual authentication between client and server using X. 509 certificates
- Mutual authentication and tunnel establishment through XMPP “SASL External”
- If client certificate passes validation client registration requests are relayed only to XMPP-Grid controller for account approval
- If client certificate does not pass validation, the connection is terminated with XMPP standards-based error messages

XMPP-Grid Client Registration

- Auto registration
  - Clients with the right cert will have their accounts auto created after authentication
  - Clients can specify authorization group of interest
- Manual registration
  - Administrator has to approve/decline client accounts after their authentication
  - Administrator can assign authorization group to the client resulting in client logoff and logging back in for the group change to take effect

3 layer security model –
Mutual-cert based authentication + account approval + authorization group assignment with policy control

Client Registration



XMPP-Grid Client Authorization

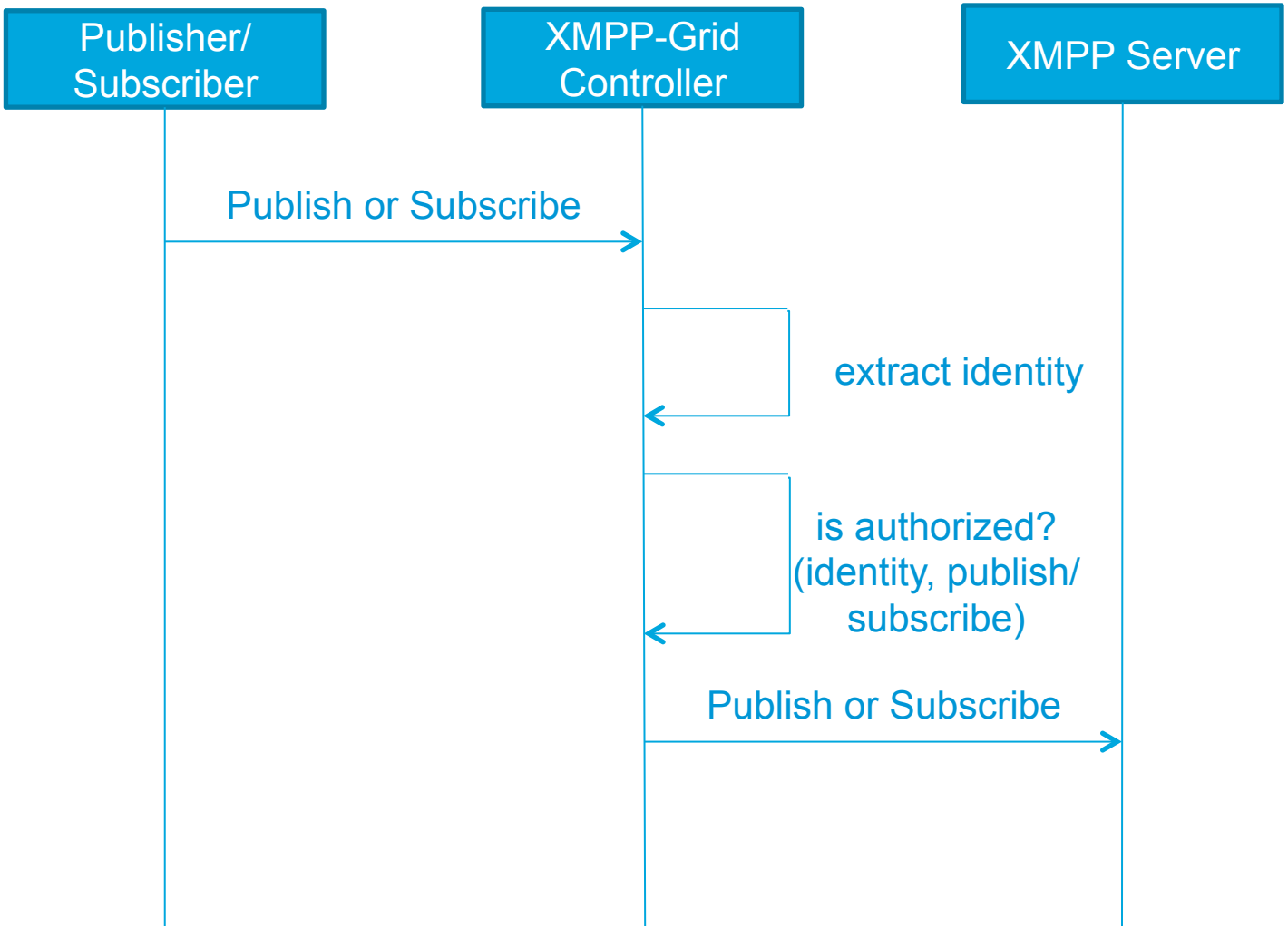
- Authorization policies can be based on attributes such as

  Authorization group, Topic name, client name, device type, operation …

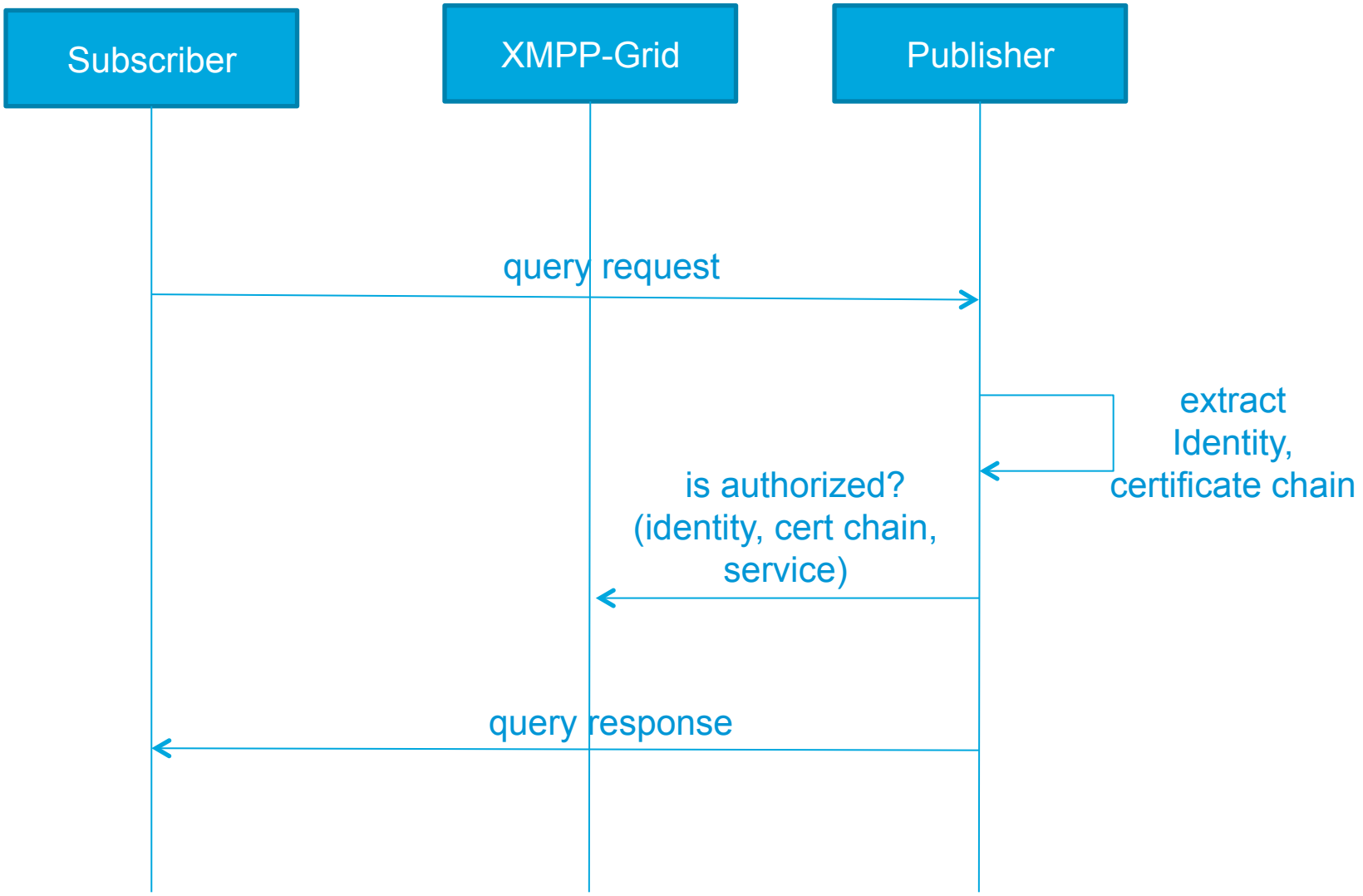
- Controller authorizes clients to publish or subscribe to a topic at “subscribe” time

- Publisher, when it receives a directed (peer-to-peer) or bulk download query from a subscriber, asks the controller for authorization using XMPP-Grid client identity

Publish/Subscribe Authorization



Directed/Bulk Query Authorization



XMPP-Grid Protocol

- Infrastructure protocol that enables client application to be agnostic to data plane protocol, XMPP
- Makes use of the XMPP transport and introduces an application layer protocol leveraging XML and XMPP extensions to define the protocol
- Provides interfaces for
  - Register, login, logout
  - Query to discover topics, capability provider discovery, directed peer-to-peer
  - Register as a publisher or subscriber to topic (information channel with publishers and subscribers sharing a well defined publisher data model)
- XMPP-Grid clients connect to the XMPP-Grid using the XMPP-Grid Protocol
- Capability providers extend the XMPP-Grid Protocol infrastructure model and define capability specific models, allowing a cleaner separation of infrastructure and capabilities that can run on XMPP-Grid

XMPP-Grid Protocol Example

// Capability Provider Discovery Request

```
<iq id="996IL-8" to="grid_controller.jabber" from="asa@syam-06.domain.com/syam-mac" type="get">
  <grid xmlns='gi' type='request'>
    <DiscoveryQuery xmlns='com.domain.gi.gcl.controller'>
      <find><param xsi:type="xs:string" xmlns:ns2="gi" xmlns:xs=" xmlns:xsi=">com.domain.ise.session.SessionQuery</param></find>
    </DiscoveryQuery>
  </grid>
</iq>
```

// Capability Provider Discovery Response

```
<iq from='grid_controller.jabber' id='996IL-8' to='asa@syam-06.domain.com/syam-mac' type='result' xmlns='jabber:client'>
  <grid type='response' xmlns='gi'>
    <DiscoveryQuery xmlns='com.domain.gi.gcl.controller'>
      <find xmlns=''><value xmlns:ns3='http://jaxb.dev.java.net/array' xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
xsi:type='ns3:stringArray'><item>ise@syam-06.domain.com/syam-mac</item></value></find>
    </DiscoveryQuery>
  </grid>
</iq>
```

XMPP-Grid Topics

- Capability provider publishes information with a defined schema on XMPP topic(s)
- Capability provider defines XML schema, topic version, available queries and notifications for each topic
- Capability provider publishes the messages to one or more XMPP topics depending on –
  - Mutually exclusive schemas – create one topic per schema
  - Same schema, but subscribers desire only a subset of attributes and values – XMPP-Grid creates subtopics and uses message filters to deliver filtered information
- Topics are discoverable on XMPP-Grid through XMPP-Grid protocol query

XMPP-Grid Subtopics & Message Filters

- Capability provider specifies semantic filters such as location, domain etc it supports for a given topic at subscribe time to the controller
- Subscribers discover the topics & supported message filters, and specify filters of interest to them to the controller
- Controller groups subscribers based on the expressed message filters, creates subtopics under the main topic and notifies the Publisher about the created subtopic
- Publisher publishes a message on the main topic and on the subtopics, after applying the message filter

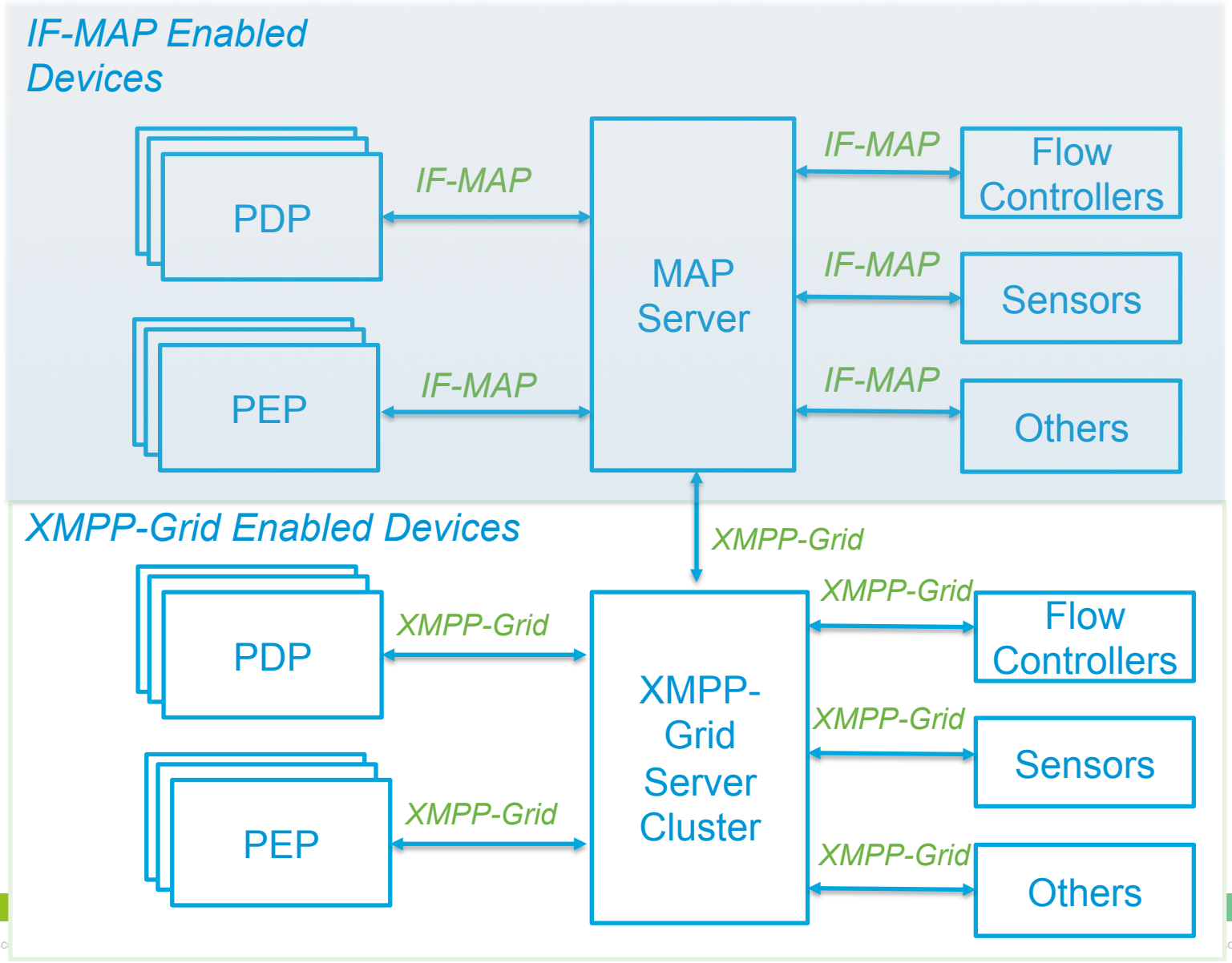
Subtopics & Message Filters

- Controller cleans up the subtopics if subscription list is 0, to avoid proliferation of subtopics
- Pub/Sub, directed and bulk query can be supported for subtopics also – it all depends on the capability provider
- Message filters can be applied on XMPP-Grid server side instead –instead of publishing on subtopic, capability provider publishes on main topic and XMPP-Grid Pub/Sub component can apply filter messages
  - Server-side message filters and specific message filter mechanisms such as XPATH are beyond the scope of this specification

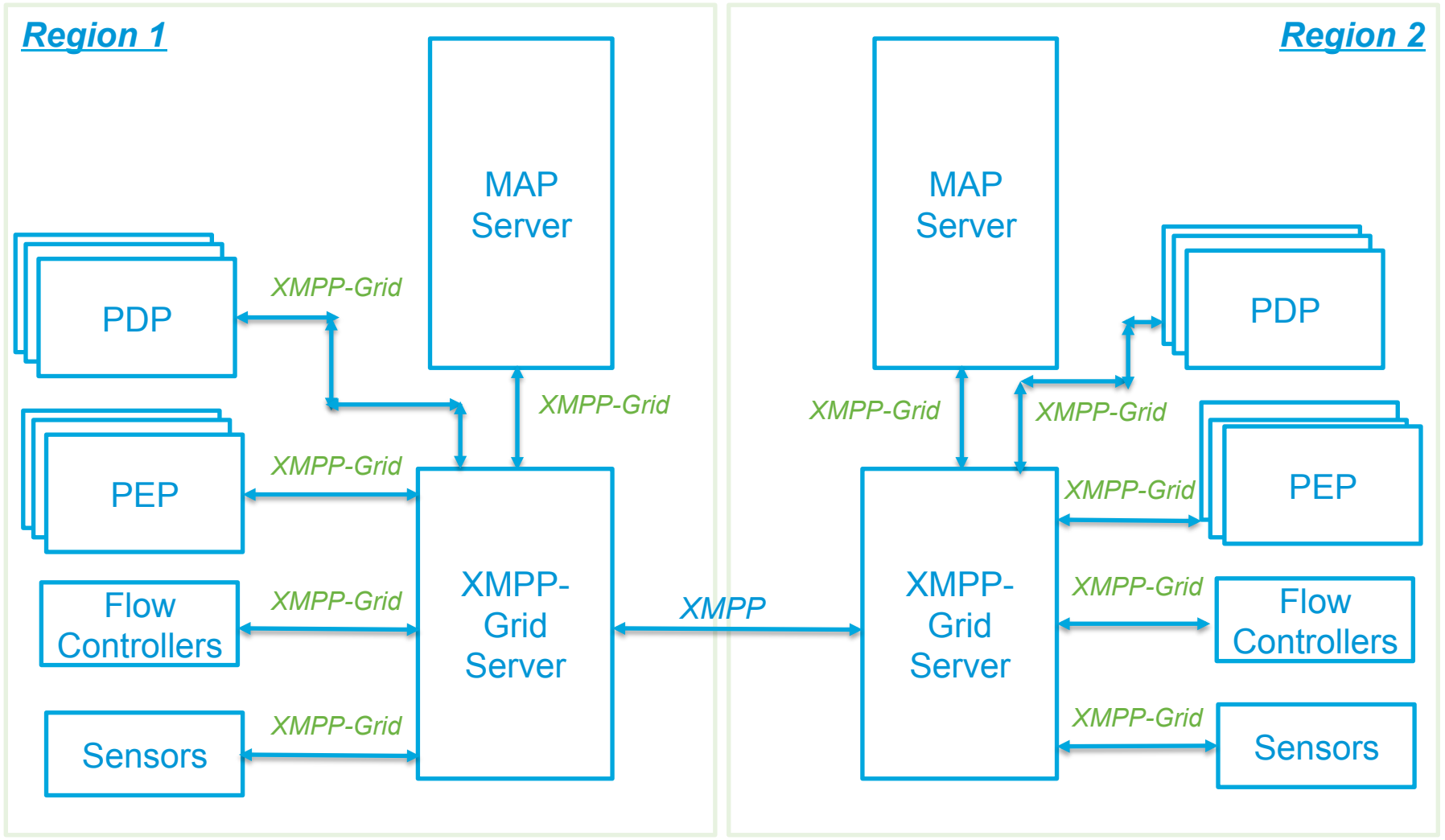
IF-MAP with XMPP-Grid

- XMPP-Grid to substitute the SOAP-based IF-MAP standard interface between the MAP server and other elements in the network
- IF-MAP data models for use-cases such as network security can be overlaid on XMPP-Grid transport to achieve model consistency for both IF-MAP enabled and XMPP-Grid enabled deployment scenarios
- MAP Server will be the participant in both the IF-MAP enabled network and the XMPP-Grid enabled network serving as aggregator and publisher of information
- MAP server can play the role of subscribers and/or publishers depending on the MAP graphs and the contextual metadata to be aggregated and/or published

MAP Server as Publisher/Subscriber on XMPP-Grid



MAP Server De-centralization with XMPP-Grid



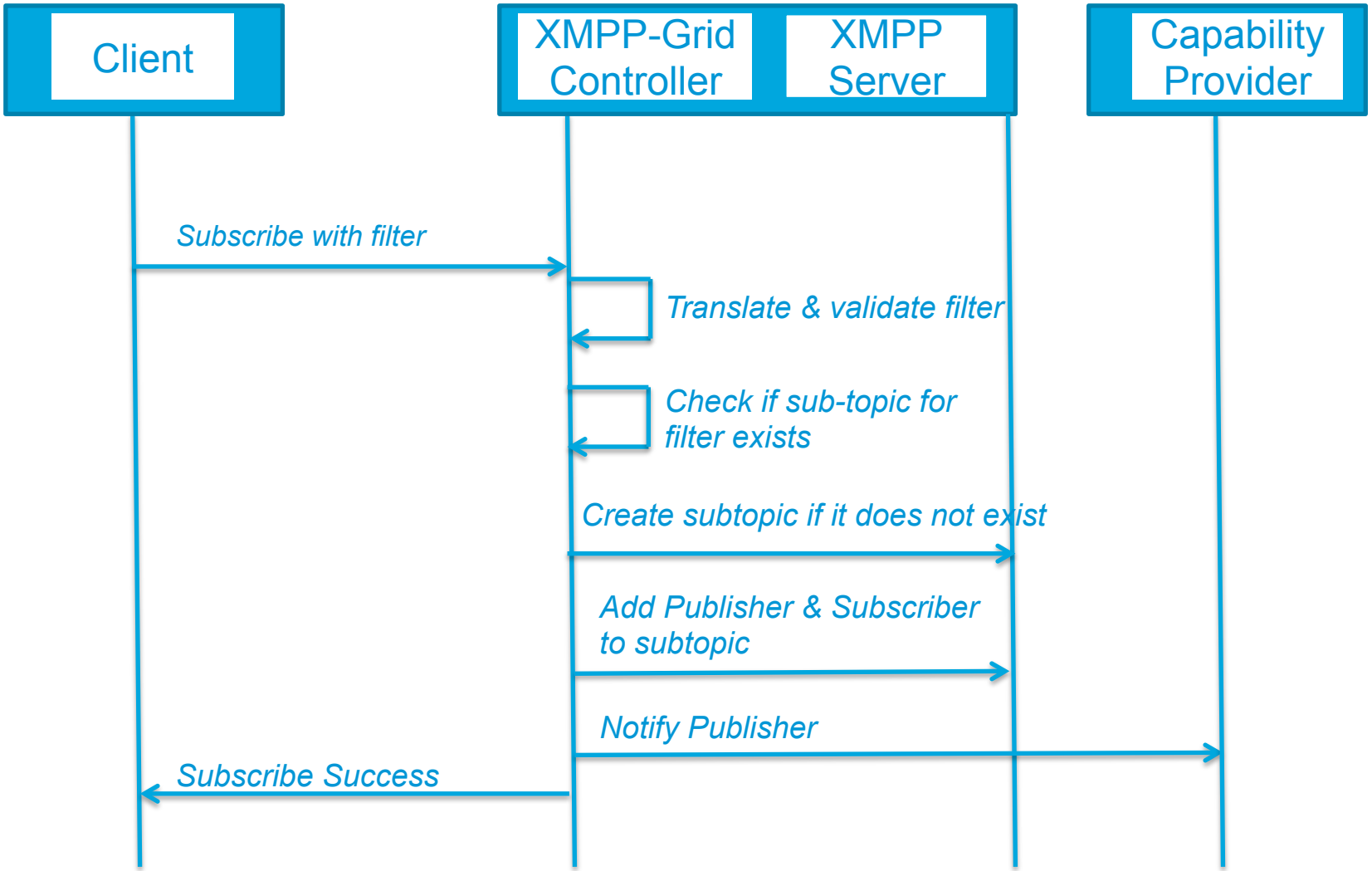
MAP Graph Subtopics & Message Filters

- MAP Server could publish the MAP graph attribute changes to interested subscribers

- Message filter criteria supported for subtopics could be based on
  - metadata types
  - metadata-identifier linkage attributes
  - metadata class
  - existing IF-MAP search criteria

Backup

Subtopic Creation Flow



Publish on Subtopics Flow