

Requirements for Subscription to YANG Datastores

[draft-ietf-i2rs-pub-sub-requirements-01](#)

NETCONF WG - IETF 92

Eric Voit, Alex Clemm, Alberto Gonzalez Prieto

evoit@cisco.com, alex@cisco.com, albertgo@cisco.com

March 24th 2015

YANG Pub/Sub IETF Drafts

Eric presents this now



Requirements for Subscription to YANG Datastores

draft-ietf-i2rs-pub-sub-requirements-01

(Intended to capture requirements across multiple WG, including futures)

Pub/Sub

Subscribing to datastore push updates

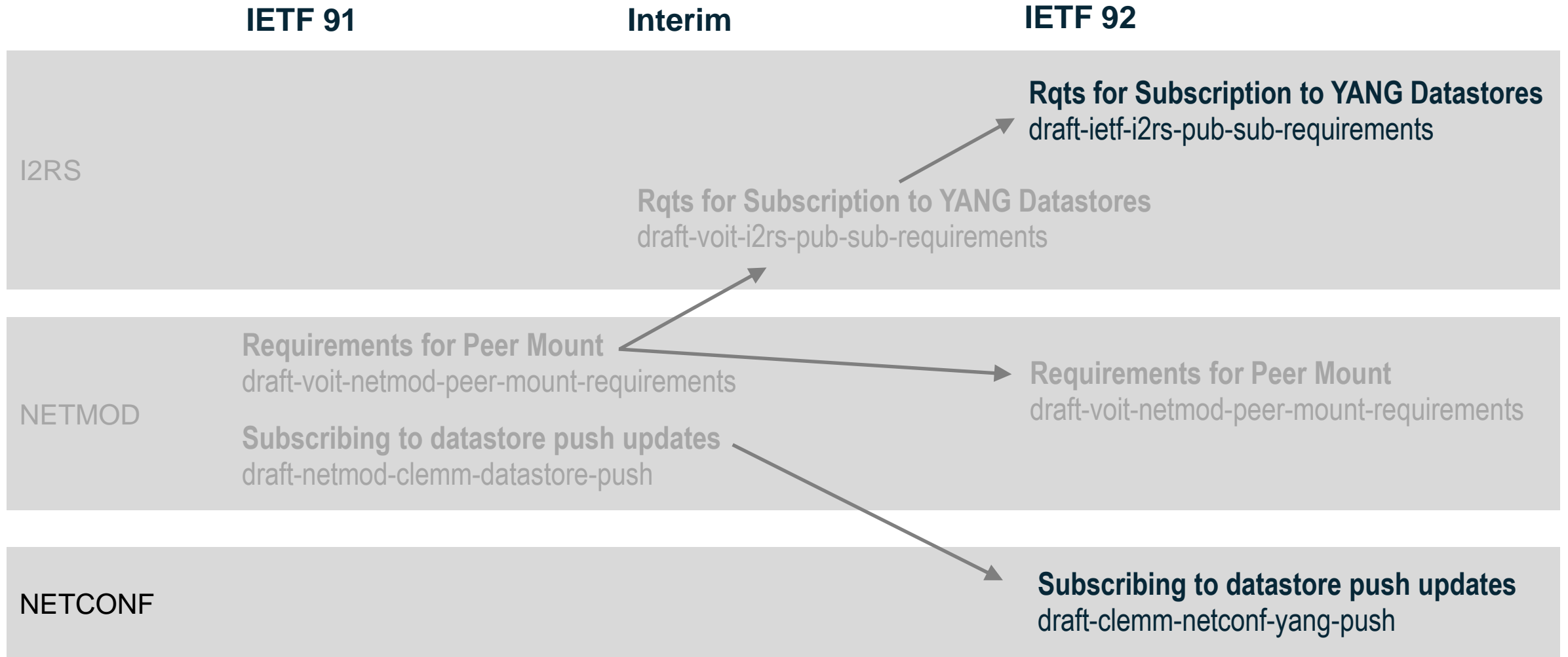
draft-clemm-netconf-yang-push-00

(Will not embody all possible futures, nor do you want it to)



Alex presents this next

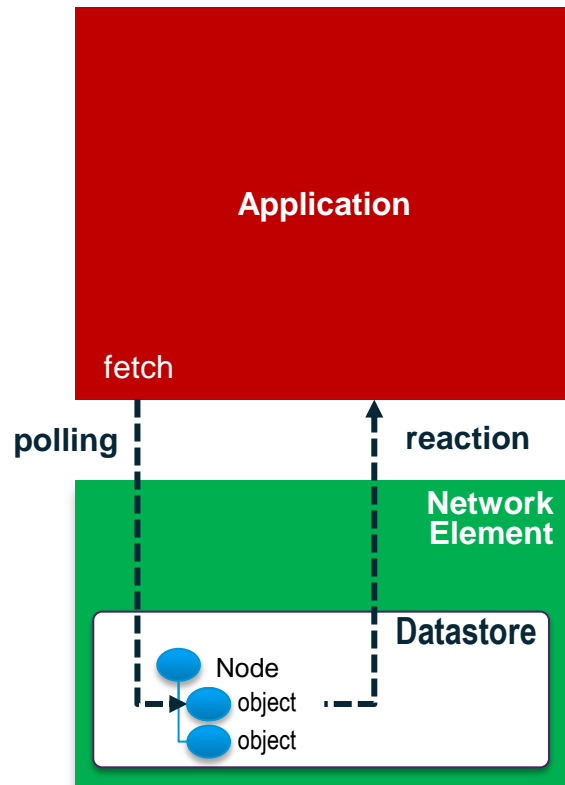
YANG Pub/Sub IETF Draft Evolution



Getting YANG Objects

What we have today:

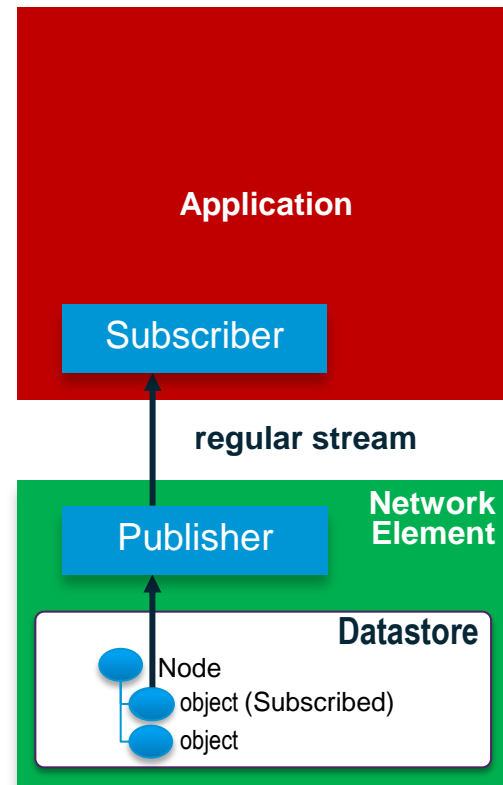
On Demand: ask for Object every time



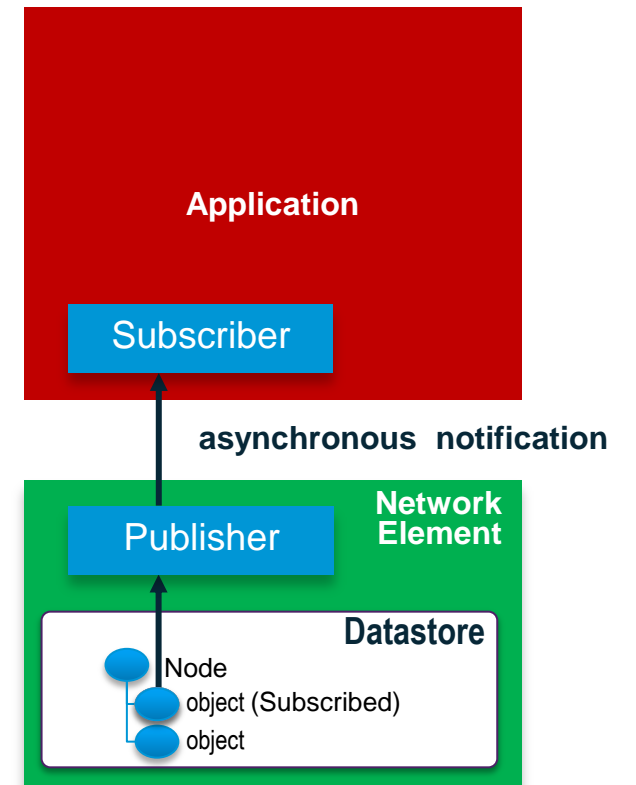
Pub/Sub benefits

- Application performance
- Processing reductions
- Subtree monitoring

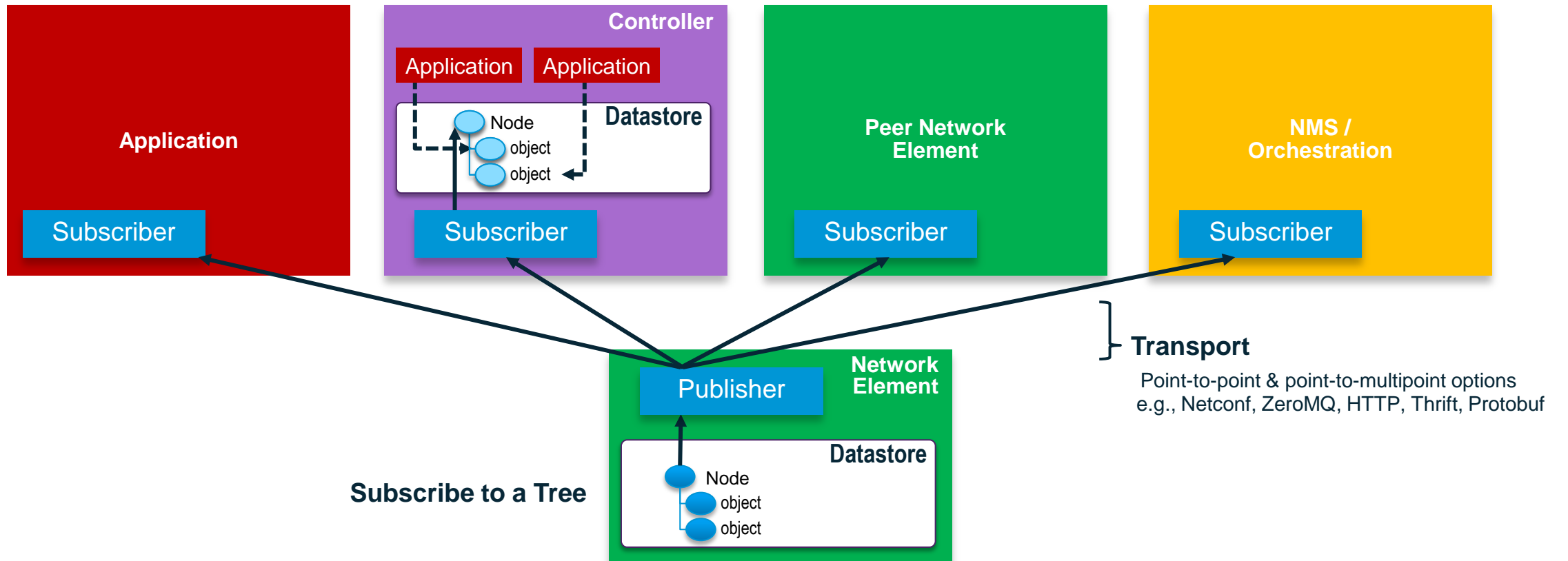
Periodic: Push Object every 'X' seconds



On Change: Push on Object change

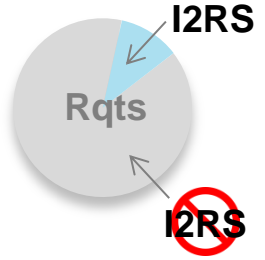


Many Consumption Models



draft-ietf-i2rs-pub-sub-requirements

Intended to cover requirements outside I2RS

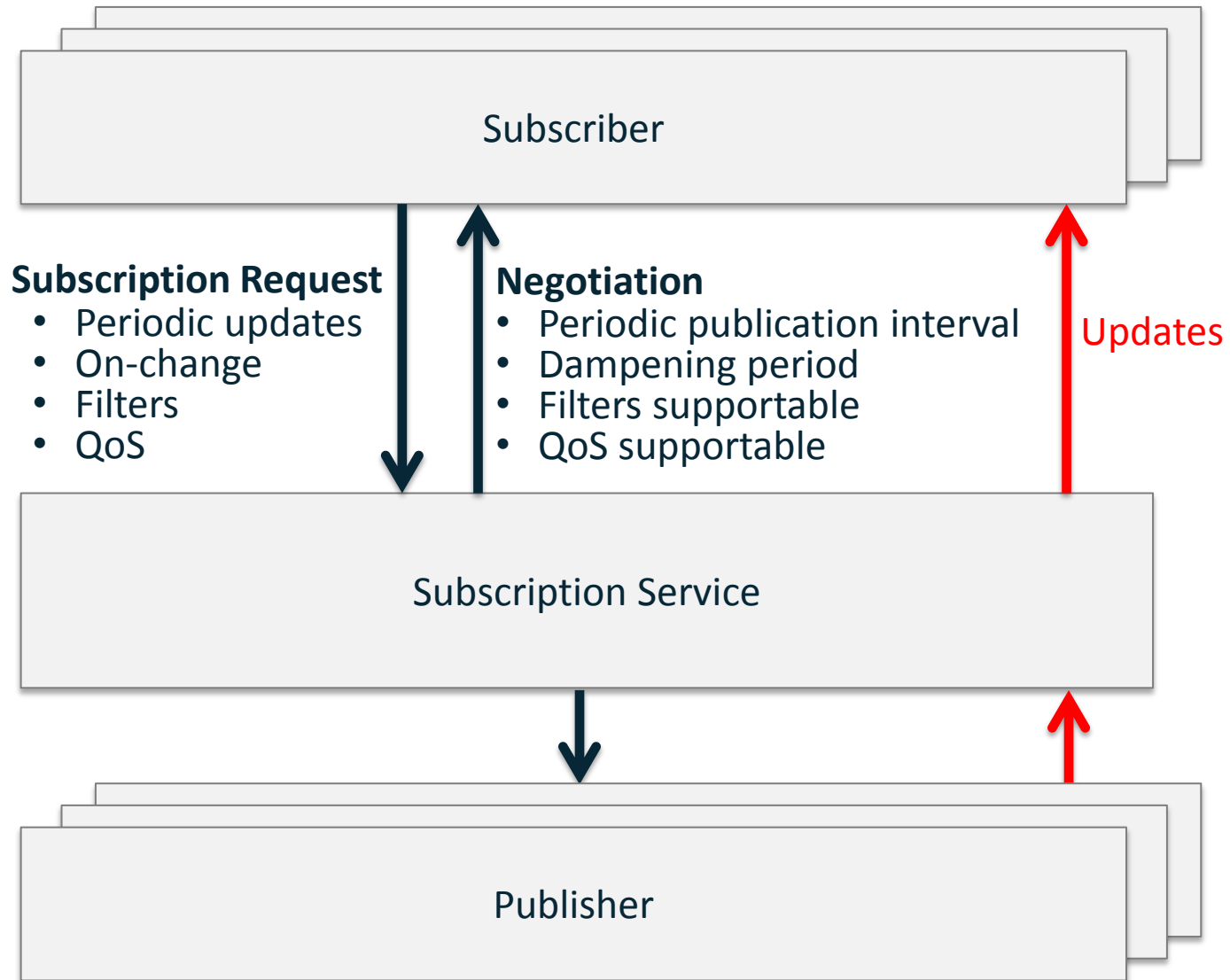


Document	Potentially Subscribed	Subscribers care about
draft-ietf-i2rs-rib-info-model	Nexthop, tunnels, MPLS	Peers going up or down Change announcement latency Filtering, multiple security tiers
RFC 7277 (IP Management)	Interface state, neighbor state	Peer reachability, potential failure Conflicting config between peers
RFC 7223 (Interface Management)	Interface enabled, traffic counters	Volume of telemetry provided Filtering, multiple security tiers
draft-ietf-netmod-acl-model	My allow, deny rules in sequence	Filtering, multiple security tiers
draft-ietf-netmod-syslog-model	Events, who is logged on	Filtering, multiple security tiers
draft-ietf-netmod-routing-cfg	Active routing protocols	Conflicting config between peers
dozens...		



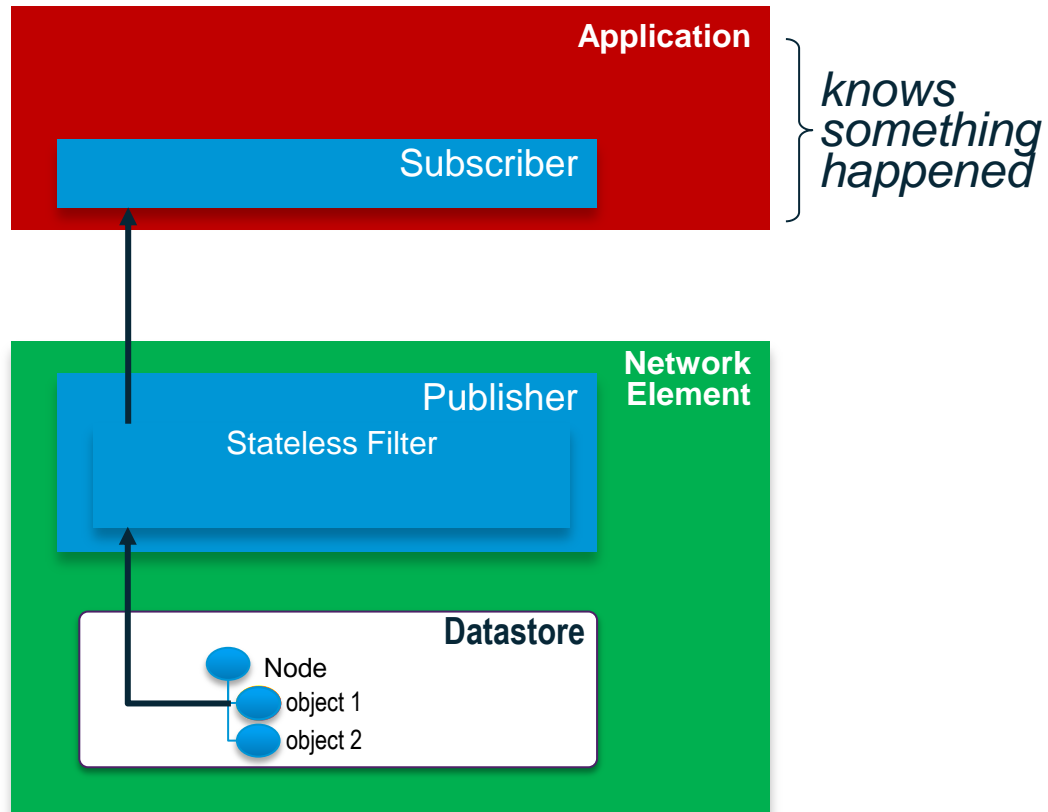
- Σ Rqts
- Periodic
 - On-change
 - Filters
 - Security
 - Dampening
 - Reaction speed
 - Update bundling
 - Negotiation
 - Multipoint

Pub/Sub Subscription Service

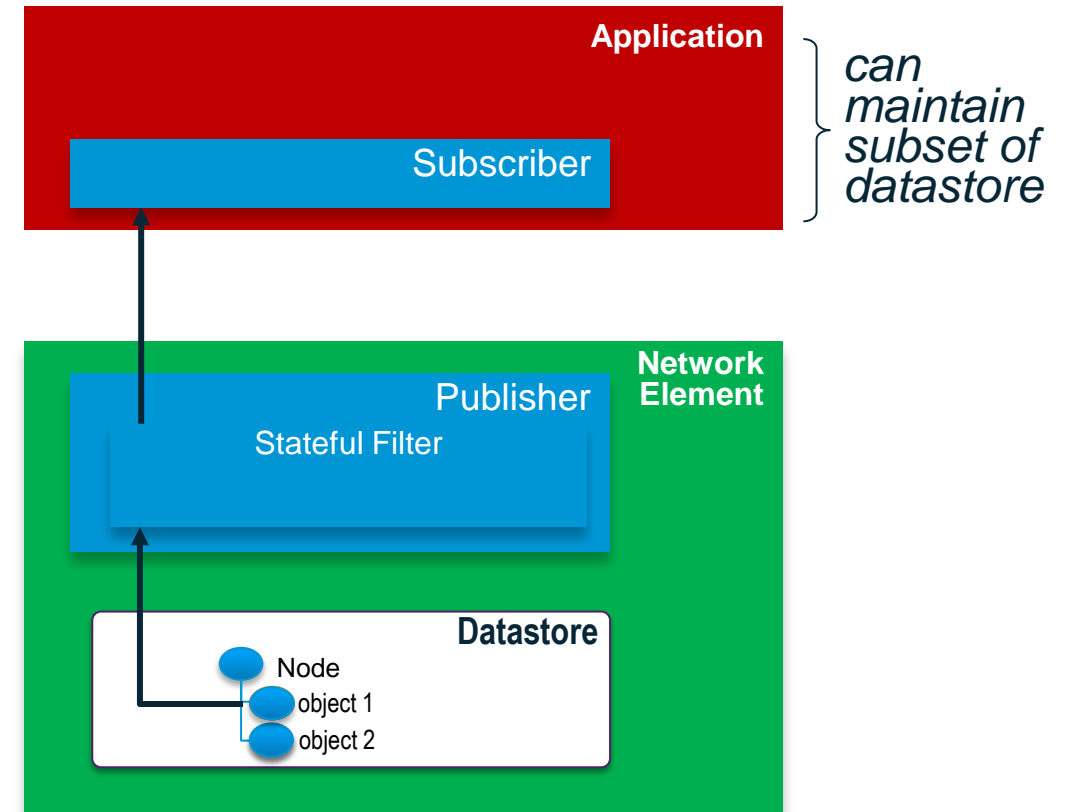


Filtering YANG Objects

Filtering Events



Maintaining Filtered Remote State

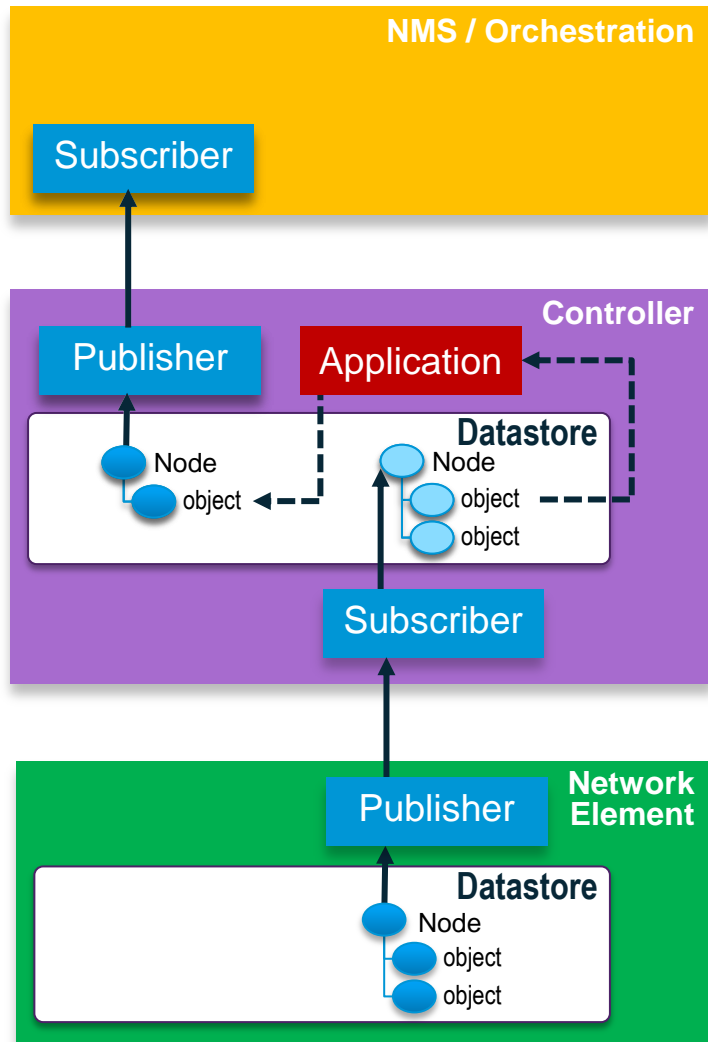


Elements of QoS for the Subscription

Liveliness	<ul style="list-style-type: none">• SHOULD notify a Subscriber if nodes can no longer be monitored or are determined to be stale
Dampening	<ul style="list-style-type: none">• MUST be able to negotiate the minimum time separation since the previous update before transmitting a subsequent update
Reliability	<ul style="list-style-type: none">• If Reliable delivery, MUST reattempt update delivery until all subscribers acknowledge receipt or some duration has passed
Coherence	<ul style="list-style-type: none">• Updates MUST be sent in-order
Presentation	<ul style="list-style-type: none">• SHOULD be able to bundle a set of discrete object notifications into a single update
Deadline	<ul style="list-style-type: none">• MUST be able to push updates at a regular cadence that corresponds with specified start and end timestamps
Push Latency	<ul style="list-style-type: none">• MUST be possible to determine the time between object change and actual Push

Q&A

Applicability beyond the Network Element











YANG Models describing network state can be published as well.

e.g., OpenDaylight reporting to OpenStack

Which can be driven by fast-reacting, multi-tier publication

Periodic Filtering

Send Update if...

	<u>Filter Type</u>	<u>Complexity</u>
Object A exists	n/a	 <p>Intermittent Periodic Reporting</p>
if Object A currently has property  or 	Simple query	
if Object A currently has property  and different property 	Complex query	
if Object A currently has property  and Object B has property 	Multi-object query	
if Object A currently has property  then run process	Distributed Analytics	
















Technology Types:

- XPATH
- SQL
- Distributed Analytics

On-Change Filtering

Send Update if...

	<u>Filter Type</u>	<u>Complexity</u>
Object A property just changed Object A has been created/deleted	n/a	 <p>Filtering Events</p> <p>Maintaining filtered remote state</p>
Object A has been created with property  or 	Stateless	
if Object A property just changed to 	simple	
if Object A property just changed to  and has different property 	simple	
if Object A property just changed and Object B has property 	complex	
if Object A property just changed, run process	multi-object	
if Object A property just changed/deleted away from 	Stateful	
if Object A property just changed from  to 	simple	
if Object A property just changed from  and Object B has property 	complex	
if Object A property just changed/deleted away from  , run process	multi-object	
	distributed analytics	

Terminology

A **Subscriber** makes requests for set(s) of YANG object data. The Subscriber is the owner of the Subscription.

A **Publisher** is responsible for distributing subscribed YANG object data per the terms of a Subscription. In general, a Publisher is the owner of the YANG datastore that is subjected to the Subscription.

A **Subscription Service** provides Subscriptions to Subscribers of YANG data. A Subscription Service interacts with the Publisher of the YANG data as needed to provide the data per the terms of the Subscription.

A **Subscription Request** for one or more YANG subtrees made by the Subscriber of a Publisher and targeted to a Receiver. A Subscription MAY include constraints which dictates how often or under what conditions YANG subtree updates might be sent.

A **Subscription** is a contract between a Subscription Service and a Subscriber that stipulates the data to be pushed and the associated terms.