gnmi-gateway

NANOG 80 (October 19-21 2020)

Colin McIntosh, Netflix

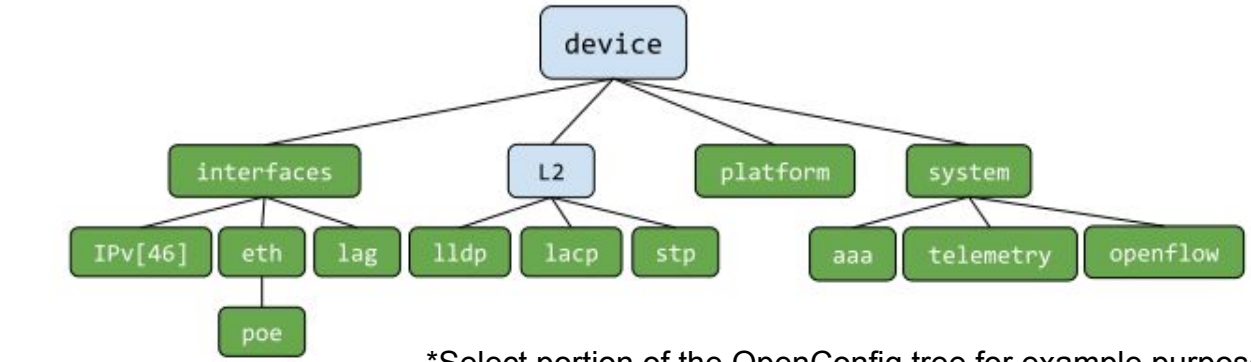
OpenConfig + gNMI: A quick primer

- gNMI == Google Network Management Interface
  - Open source protocol
  - Stream state and configuration data to/from targets (i.e. network devices or gNMI-enabled services)
  - Defined as a specification and protobuf model
  - Provided RPCs are **Subscribe**, **Get**, **Set**, & **Capabilities**

Colin McIntosh
colin@netflix.com

OpenConfig + gNMI: A quick primer

- OpenConfig == working group that defines vendor-agnostic data models
  - Develops various standards and protocols for managing networks
  - Models are defined in YANG (RFC 6020)
  - Tree structure of “leaves”

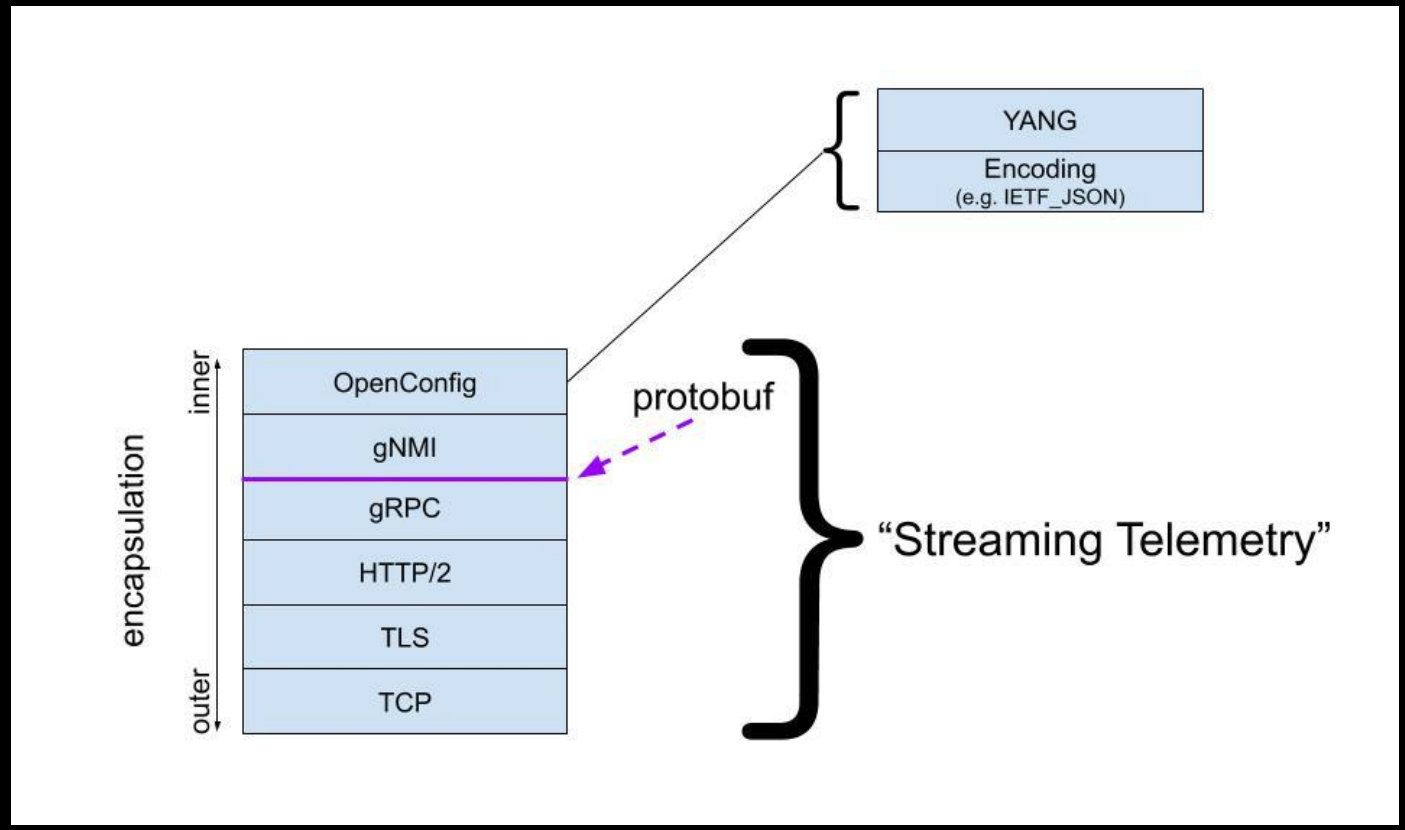


*Select portion of the OpenConfig tree for example purposes

Colin McIntosh
colin@netflix.com

OpenConfig + gNMI: “Streaming Telemetry”

- Polled telemetry (pull)
  - SNMP
  - Screen scraping
  - Proprietary APIs
- Streaming telemetry (push)
  - gNMI
  - NETCONF
  - sFlow
- Models
  - OpenConfig
  - IETF Models
  - Proprietary



Colin McIntosh
colin@netflix.com

OpenConfig + gNMI: Existing gNMI Systems

- Surveyed existing systems that support gNMI and OpenConfig[1]
- We found they were limited in their capabilities:
  - Only supported one or few devices
  - No high availability
  - Focused on Subscribe with no plans to add other RPCs
  - Lacking support for dynamic target and client connections

- Reference work at github.com/openconfig/gnmi
  - Functional code with no robust system

[1] See Appendix A

Colin McIntosh
colin@netflix.com

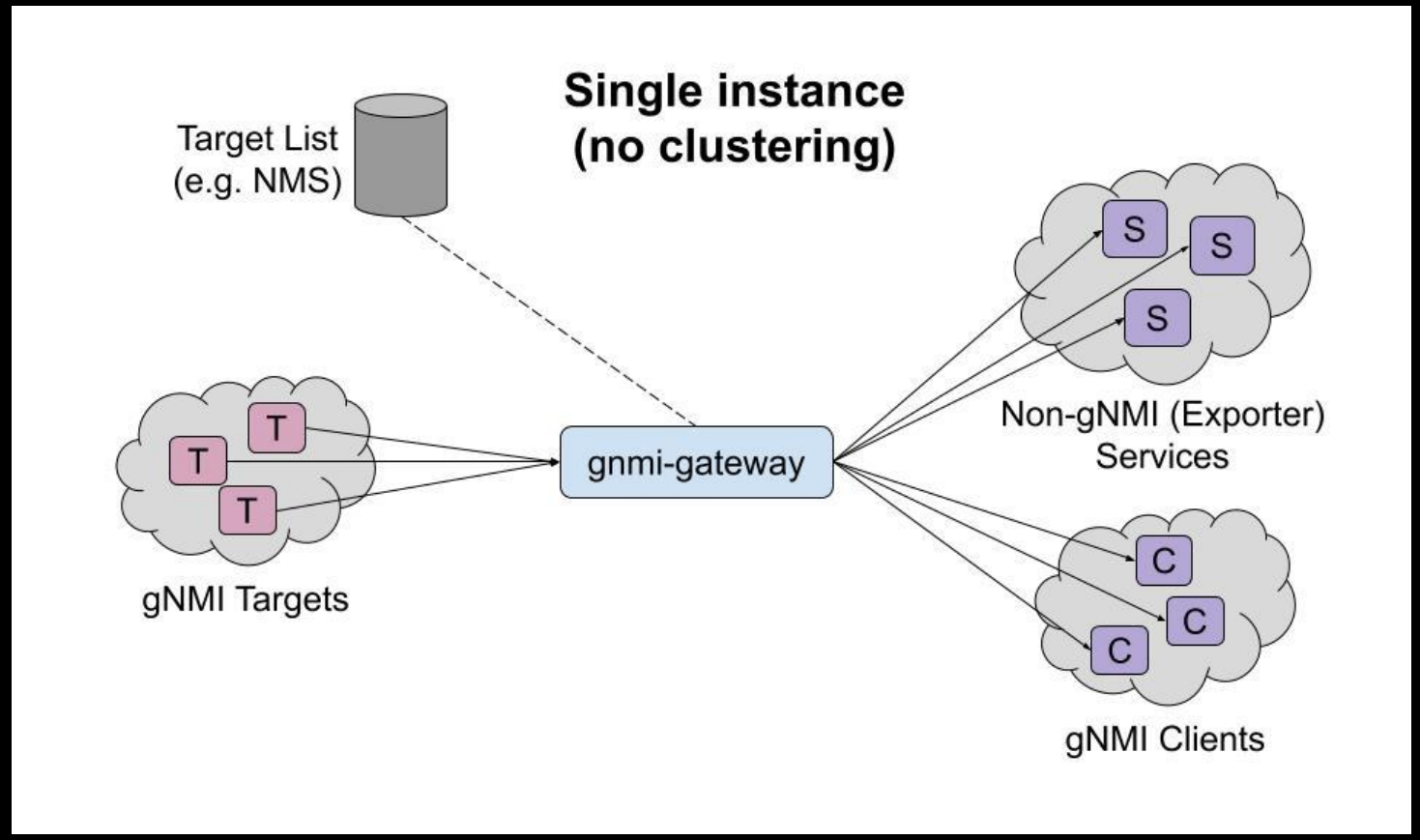
gnmi-gateway: Goals

- Provide highly available gNMI streams to clients (clustering)
- Allow for many dynamic clients and targets (target loaders)
- Distribute gNMI streams to non-gNMI systems (exporters)
- Utilize existing code from *github.com/openconfig/gnmi*
- Plan support for all gNMI RPCs
  (Subscribe, Get, Set, & Capabilities)
- Plan for extensibility
- No coding needed to start using gNMI data

- We built a service in Golang to accomplish our goals:
  gnmi-gateway

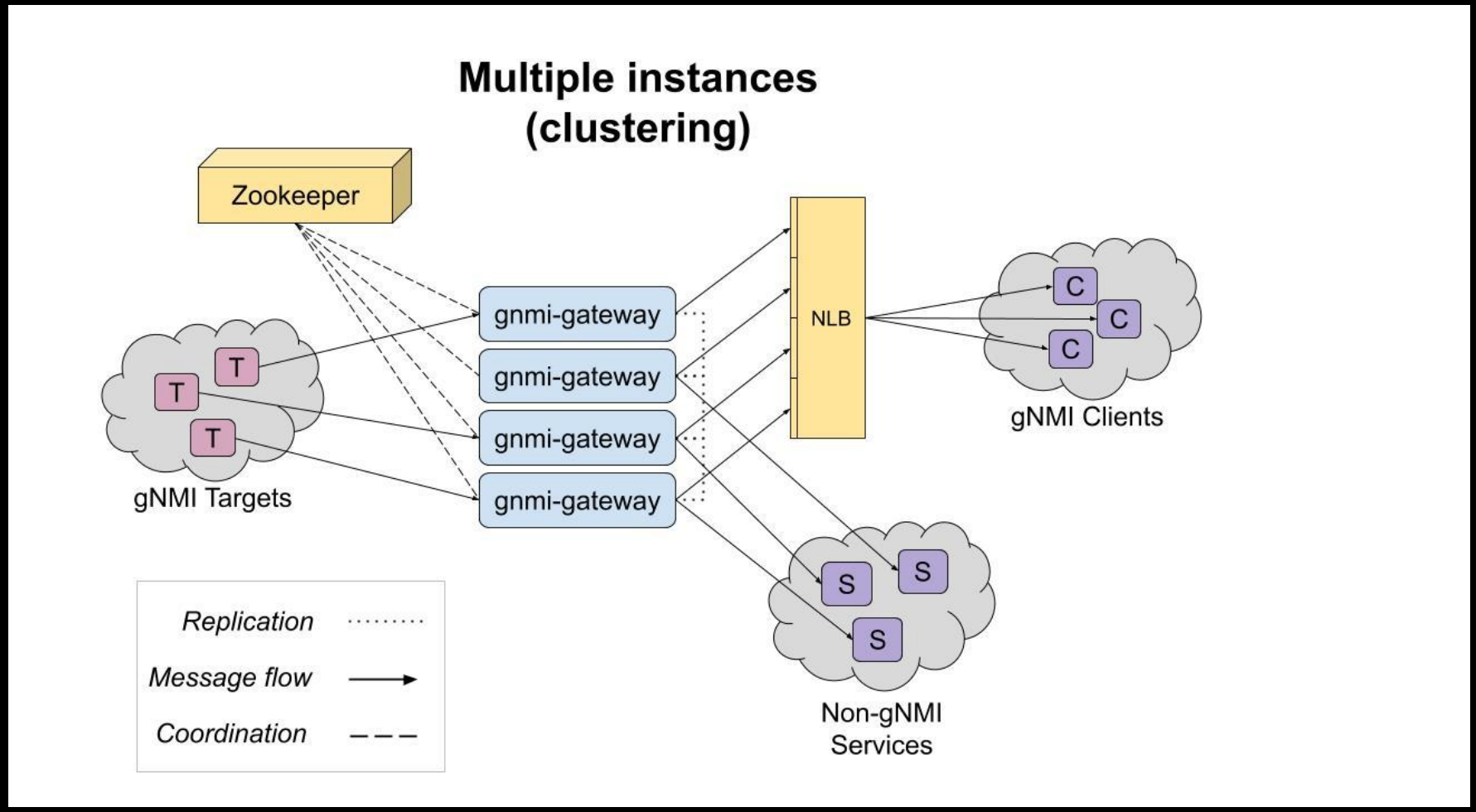
Colin McIntosh
colin@netflix.com

gnmi-gateway: Deployment Scenarios



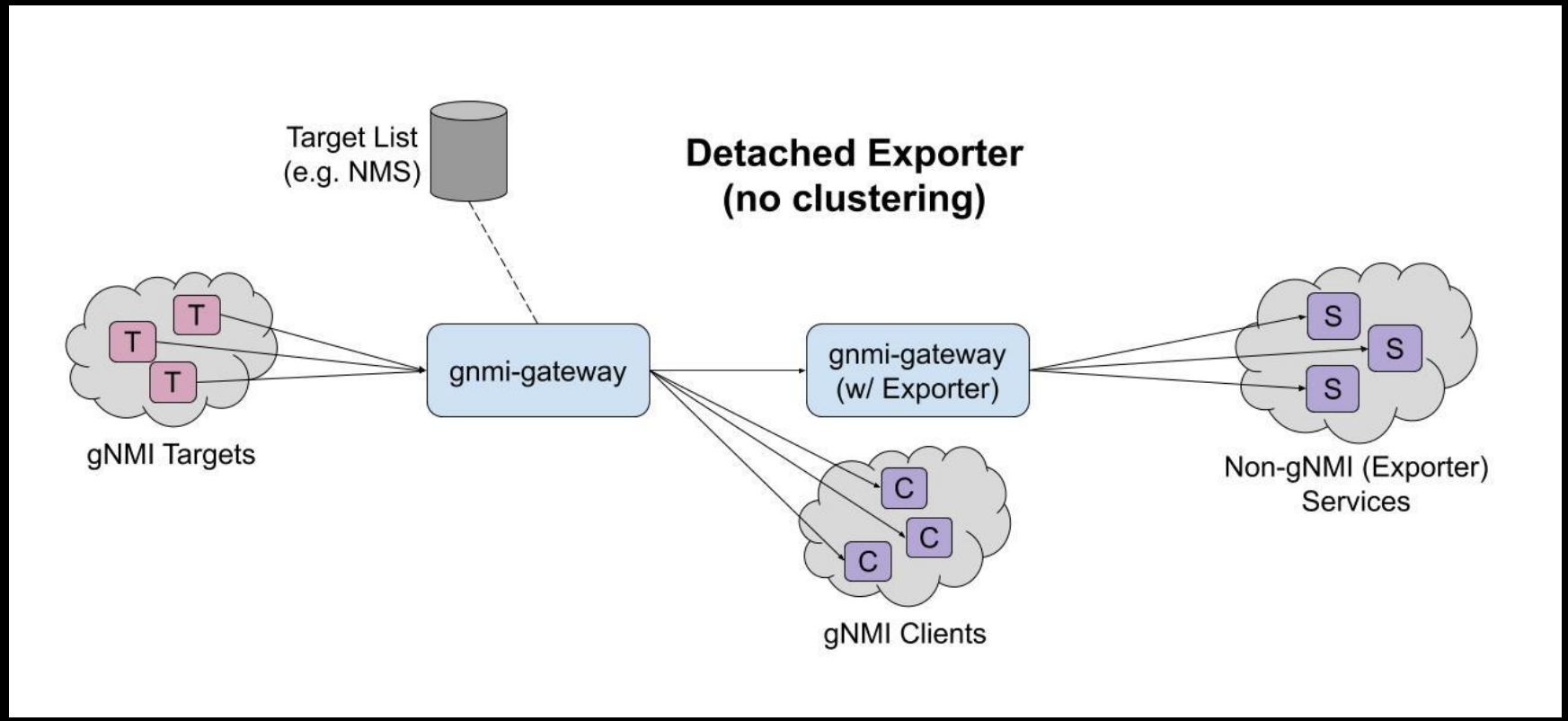
Colin McIntosh
colin@netflix.com

gnmi-gateway: Deployment Scenarios



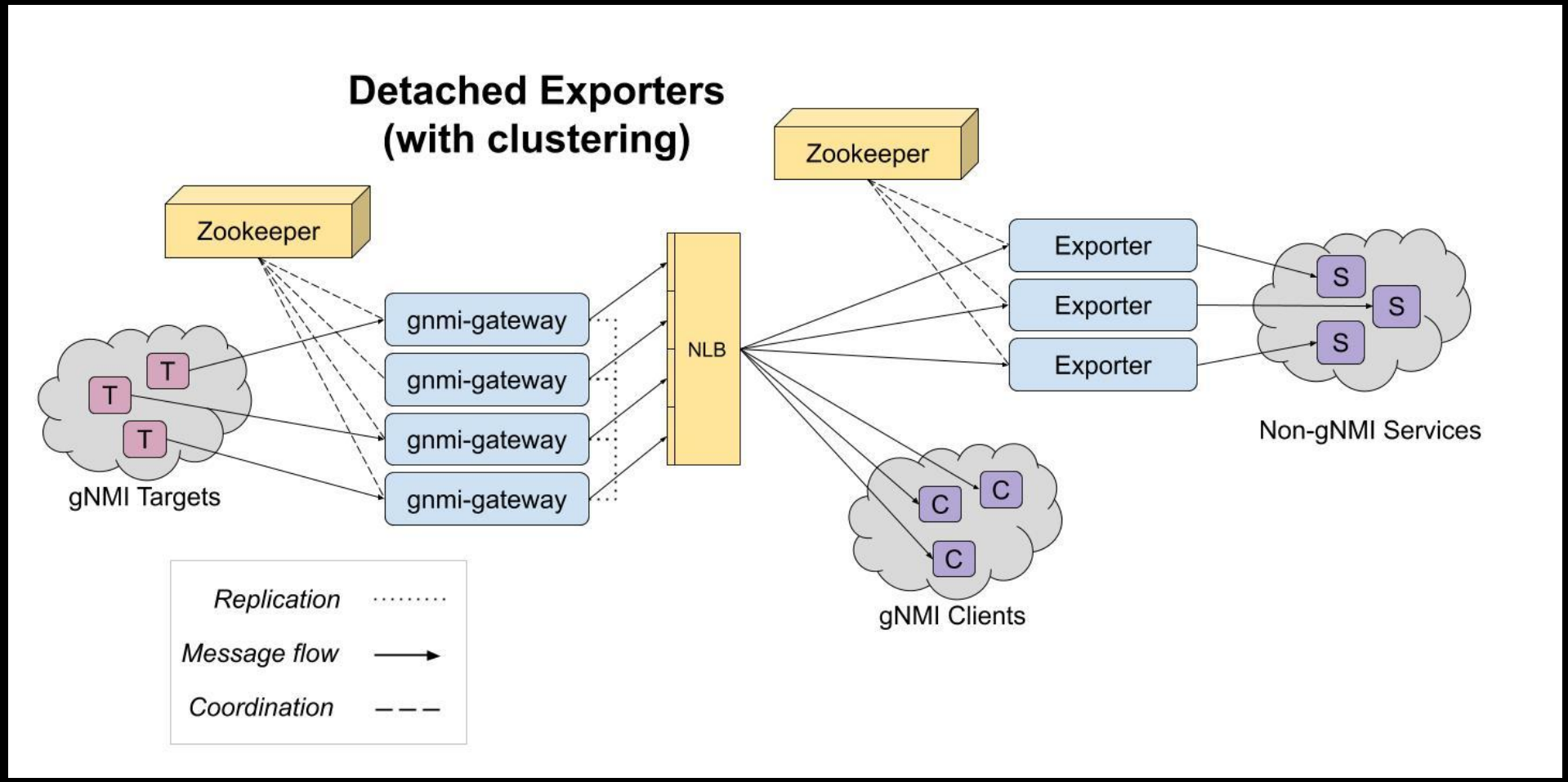
Colin McIntosh
colin@netflix.com

gnmi-gateway: Deployment Scenarios



Colin McIntosh
colin@netflix.com

gnmi-gateway: Deployment Scenarios



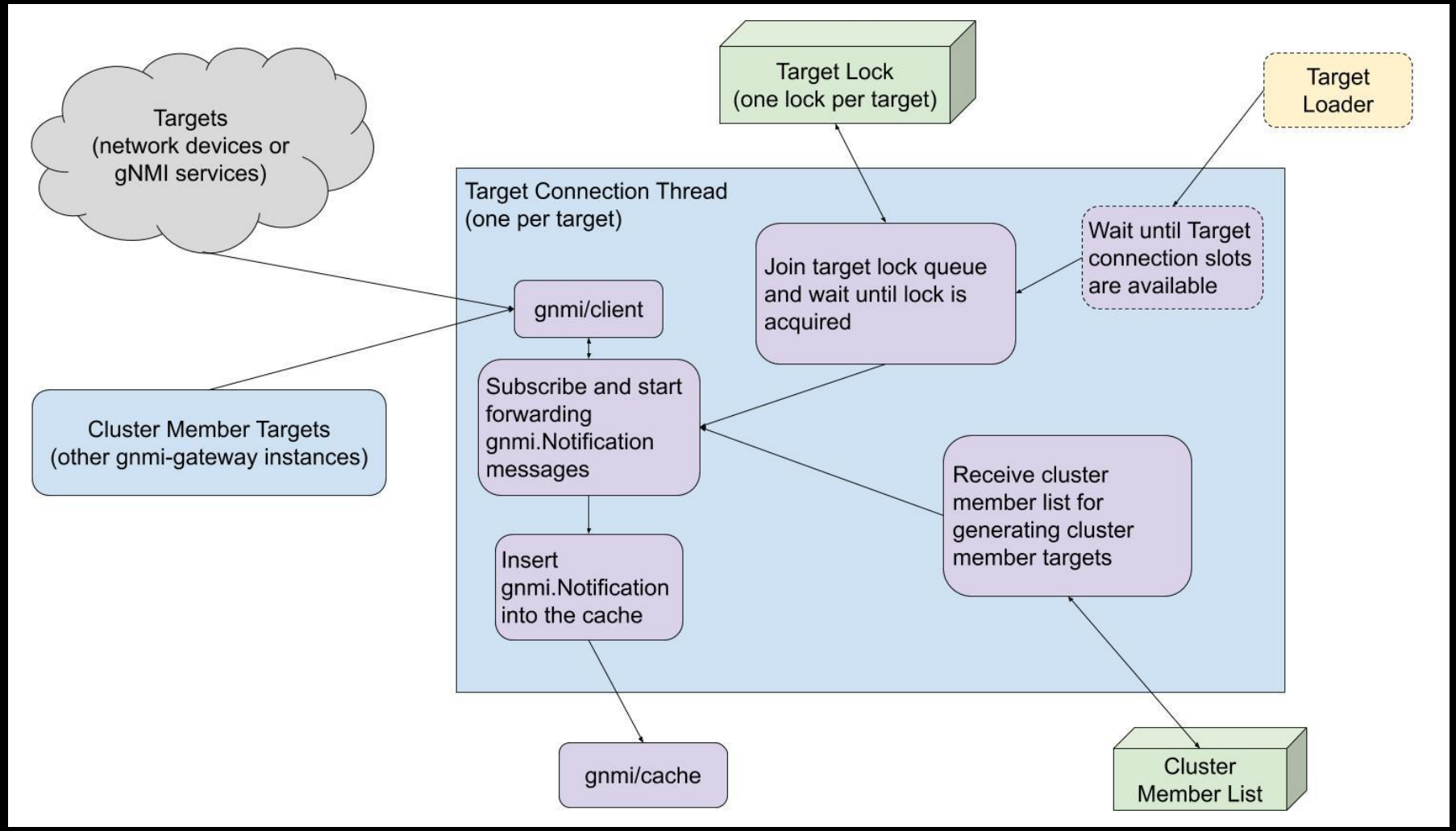
Colin McIntosh
colin@netflix.com

gnmi-gateway: Clustering

- Multiple gnmi-gateway instances can be clustered to provide quick failover for gNMI streams
- Currently using Zookeeper, but can be extended to other services

Colin McIntosh
colin@netflix.com

gnmi-gateway: Clustering



Colin McIntosh
colin@netflix.com

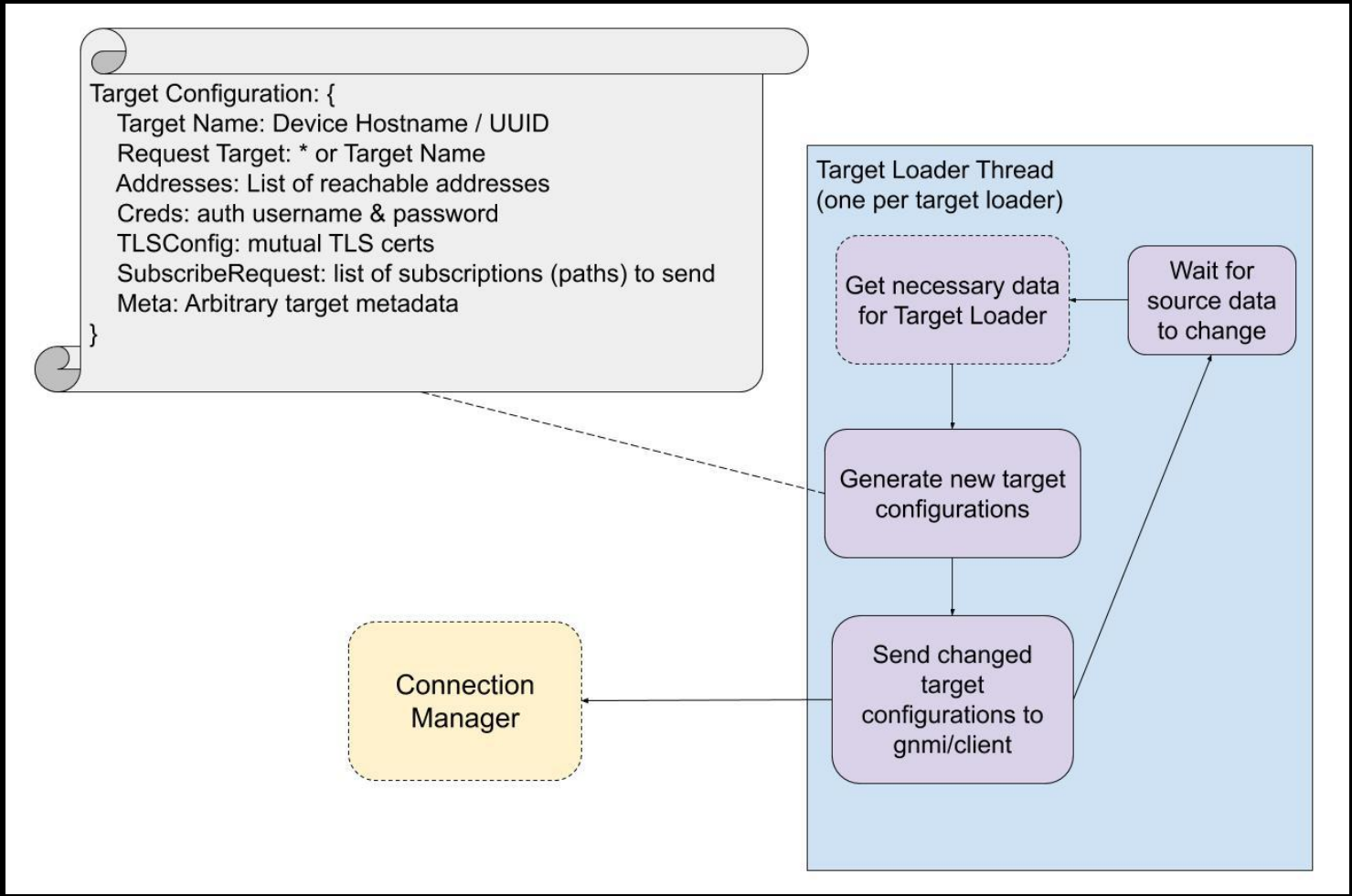
gnmi-gateway: Target Loaders

- Target Loaders allow for dynamic loading/unloading of gNMI targets
- Examples
  - Watch an NMS for new devices with specific tags
  - Form connections to static targets like an SNMP poller that has a gNMI Subscribe interface
- Extendable with a Golang interface
- Included Target Loaders:
  - Watched File
  - Netbox

```
type TargetLoader interface {
    // Get the Configuration once.
    GetConfiguration() (*targetpb.Configuration, error)
    // Start the loader, if necessary.
    // Start will be called once by the gateway after StartGateway is called.
    Start() error
    // Start watching the configuration for changes and send the entire
    // configuration to the supplied channel when a change is detected.
    WatchConfiguration(chan<- *connections.TargetConnectionControl) error
}
```

Colin McIntosh
colin@netflix.com

gnmi-gateway: Target Loaders



Colin McIntosh
colin@netflix.com

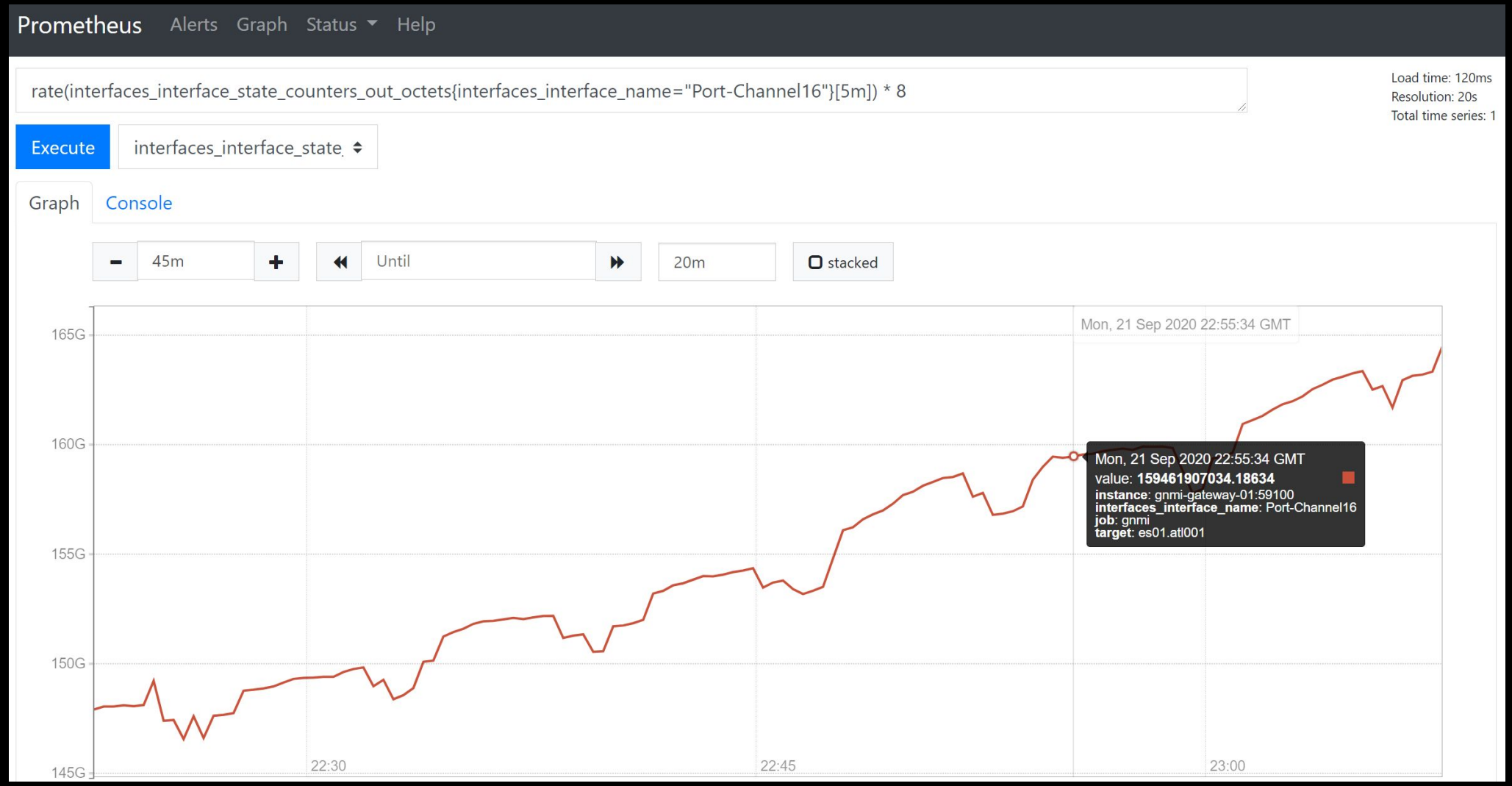
gnmi-gateway: Exporters

- Exporters forwarded data to gNMI + OpenConfig unaware systems
- For example, sending metrics to a time-series database
- Extendable with a Golang interface
- Included Exporters:
  - Prometheus
  - Atlas
  - Kafka

```go
// Exporter is an interface to send data to other systems and protocols.
type Exporter interface {
    // Start will be called once by the gateway.Gateway after StartGateway
    // is called. It will receive a pointer to the cache.Cache that
    // receives all of the updates from gNMI targets that the gateway has a
    // subscription for. If Start returns an error the gateway will fail to
    // start with an error.
    Start(*cache.Cache) error
    // Export will be called once for every gNMI notification that is inserted
    // into the cache.Cache. Export should complete as quickly as possible to
    // prevent delays in the system and upstream gNMI clients.
    // Export receives the leaf parameter which is a *ctree.Leaf type and
    // has a value of type *gnmipb.Notification. You can access the notification
    // with a type assertion: leaf.Value().(*gnmipb.Notification)
    Export(leaf *ctree.Leaf)
}
```

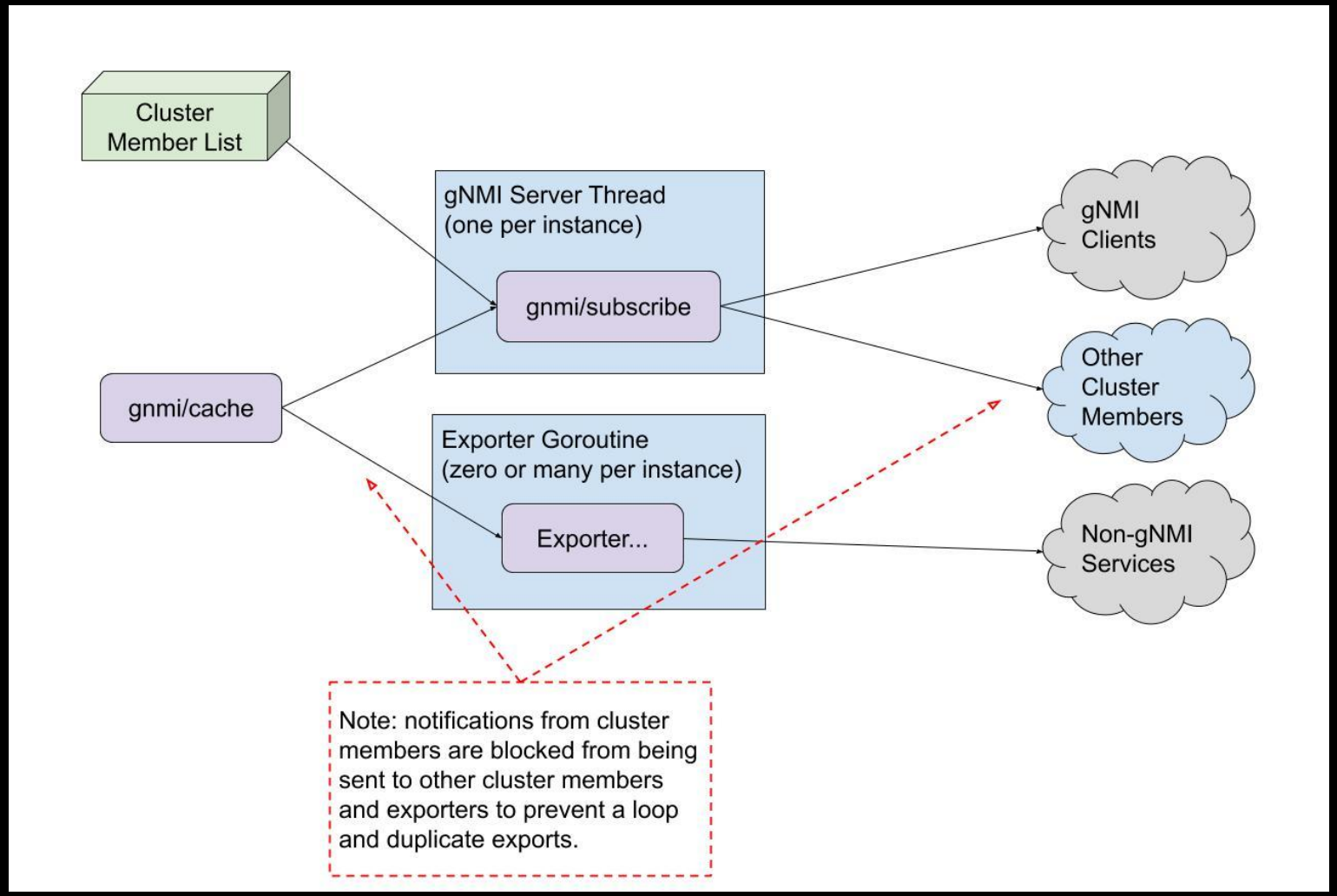
Colin McIntosh
colin@netflix.com

gnmi-gateway: Exporters



Colin McIntosh
colin@netflix.com

gnmi-gateway: Exporters



Colin McIntosh
colin@netflix.com

gnmi-gateway: Other Good Stuff

- Included Dockerfile for easy testing
- Examples for gnmi-gateway configurations
- Design docs and diagrams are available in GitHub

Colin McIntosh
colin@netflix.com

gnmi-gateway: Next Steps

- Add Get, Set, & Capabilities RPCs
- Develop new exporters
  - InfluxDB
  - Hive
- Include additional configuration examples
- Include a more extensive deployment guide in the docs

Colin McIntosh
colin@netflix.com

“Live” Demo

Colin McIntosh
colin@netflix.com

Thank You.

github.com/openconfig/gnmi-gateway

Colin McIntosh
colin@netflix.com

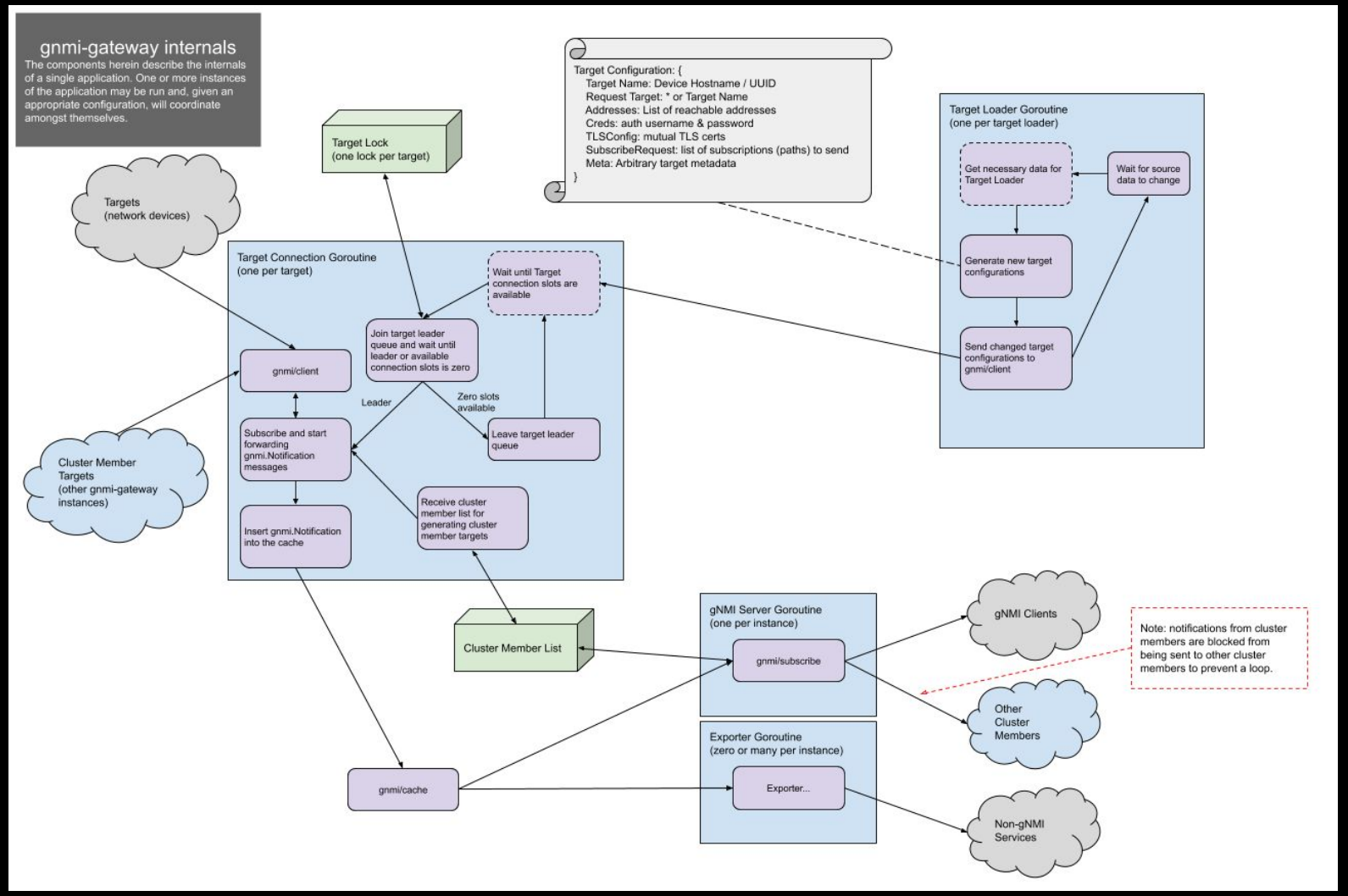
Appendix A - Existing gNMI services

During our research we searched for services that provided gNMI or OpenConfig capabilities and along with the ability to export data to different protocols and systems. These are the existing services we reviewed:

- gnmi_collector
- gNMI Plugin for Telegraf
- Panoptes
- Cisco Big Muddy

Colin McIntosh
colin@netflix.com

Appendix B - Internals



[Diagram Link]

Colin McIntosh
colin@netflix.com

Appendix C - OpenConfig & gNMI Links

- gNMI GitHub Repo
- gNMI Specification
- OpenConfig FAQ

Colin McIntosh
colin@netflix.com