May 2023

PROACTIVE EXCHANGE OF DATA BETWEEN CLOUD PROVIDERS VIA CONTROLLER COORDINATION AND TRIGGER DYNAMIC WORKFLOWS

Ram Mohan R

Rajesh I V

Vinay Saini

PROACTIVE EXCHANGE OF DATA BETWEEN CLOUD PROVIDERS VIA CONTROLLER COORDINATION AND TRIGGER DYNAMIC WORKFLOWS

AUTHORS:
Ram Mohan R
Rajesh I V
Vinay Saini

ABSTRACT

A multi-cloud Software Defined Network (SDN) controller proactively learns insights about subscribers, such as enterprise users, end users, and/or other cloud providers. Based on the learned insights, the multi-SDN controller applies dynamic policies on other cloud provides to which those subscribers are attached to. The multi-cloud SDN controller co-ordinates with various cloud providers, enterprise network controllers, and Internet Service Providers (ISPs) to proactively notify other cloud providers with information about affected users so that those providers can install additional resources at cloud edge/core on the fly. Additionally, the multi-cloud SDN controller facilitates a warm hand off from one cloud region to another cloud region. When the multi-cloud SDN controller learns about an enterprise outage, it proactively notifies other cloud providers of the outage event and the other cloud providers can use this for a warm hand off of session to the region(s) through which the users will be reconnected. The likely regions are derived based on telemetry obtained from multi-cloud SDN controller. The multi-cloud SDN controller also triggers a proactive cleanup of user context of the cloud provider side. The cloud provider cleans up after the connection reset event based on information from the multi-cloud SDN controller, rather than wait on a timeout of the connection.

DETAILED DESCRIPTION

Application services hosted in the cloud typically use cloud-native technologies such as containers and microservices design paradigms. Applications that previously performed multiple functions and resided as a single integrated package on dedicated hardware on premise may now be implemented as a group of microservices. The microservices model on cloud-native platforms gives design flexibility with respect to reusing certain functions (e.g., ingress, authentication, platform utilities, troubleshooting service, etc.) across various cloud solutions.

Additionally, cloud native solutions provide the advantage of automatic scaling/elasticity. Consumers of cloud services typically use client applications running on various forms of devices (e.g., laptops, mobile clients, browsers, Voice over Internet Protocol (VoIP) telephones, or other specialized endpoints) to consume the cloud services typically by connecting over the top.

Many cloud solutions have a rich set of telemetry (e.g., application usage, quality, transport level metrics, etc.) from both client devices and cloud services that give visibility into the operation of the application services. There may be various levels of telemetry provided by different aspects of the cloud solution. For instance, the cloud native platform infrastructure may provide some telemetry data and application pods may provide other telemetry data.

One of the problems that cloud services experience is the ability to proactively learn about client issues and/or client network/provider network issue and apply actions based on those issues. For example, many enterprises subscribe to multiple cloud providers for various services. If the data of one cloud provider experiences a data leak for some of the enterprise users, then there is a significant likelihood that the data from those enterprise users is vulnerable. It will be useful for other cloud providers to whom these subscribers are attached to receive a notification about this leak allowing the other cloud providers to apply additional defenses, monitoring, or quarantine as need be.

Some existing approaches may include an offline update of enterprise administration to other cloud providers. Other cloud providers may use available metrics (e.g., client application metrics, network metrics) that are periodically provided from the client side to the server side to look for anomalies and trigger remediation actions. However, waiting for these metrics to be received at the cloud provider may trigger the remediation actions too late, since damage may have already been done.

The techniques included in this proposal use a multi-cloud Software Defined Networking (SDN) controller that coordinates with cloud providers, enterprise network controllers, and Internet Service Providers (ISPs) to proactively notify other cloud providers with information on affected users so that those other cloud providers can install additional defenses on the fly.

Another issue with cloud providers is handing off active sessions from one region to another region and the cleanup of the user context in the original region. Typically, when an enterprise loses network connectivity due to an ISP outage or enterprise network link issues, they would use fallback paths to reach the cloud providers. This may move the enterprise connection to a different region of the cloud provider and not the region where the enterprise users were originally connected. One existing solution to this issue may involve the secondary region re-authenticating the user and setting up a new session on the cloud side for the services the users are trying to access. Another option is for the cloud provider to save session context checkpoints in a database that is synchronized and distributed globally, which incurs significant additional expense for the cloud provider.

The techniques of this proposal facilitate a warm handoff process in which the multi-cloud controller can proactively notify individual cloud providers on learning about an enterprise outage event. The notification from the multi-cloud controller may prompt the cloud provider to start a warm handoff of user sessions to the region(s) to which the users will likely be reconnected. The cloud provider may derive the likely region(s) based on telemetry obtained from multi-cloud controller. This notification may also enable the cloud provider to clean up disconnected user contexts, which presents a challenge - many cloud providers wait for a user session timeout or a connection reset event to trigger cleanup of the disconnected user context.

Yet another issue with an enterprise subscribing/consuming services multiple cloud providers is that the multiple cloud providers have no method for exchanging context about user behavior and/or enterprise traffic patterns to proactively apply policies. For example, an enterprise may use one cloud provider for its telephone calling flows, another cloud provider for its meeting flows, a different cloud provider for email, and other cloud providers for network visibility telemetry. When the enterprise users access services across these clouds, there would be behavior patterns that would be common (e.g., time of the day the users access services, number/frequency of meetings, telephone call volume, etc.). Patterns of use observed in one cloud provider may provide useful information for other cloud providers with the same set of enterprise users.

There could be many use-cases in which a change in user behavior from a typical pattern detected by one cloud provider may provide useful insight and/or an actionable notification for other cloud providers. For instance, a disruption in one of the core services (like an authentication service) may cause a surge in network traffic when the core service is restored as all of the clients/devices try to re-login to the cloud provider and re-subscribe to cloud services. Typically, each of these cloud providers will see a sudden surge in inbound connection attempts, access requests to applications, and database services. The cloud infrastructure may have issues in handling this sudden surge and may resort to standard solutions, such as throttling. Having some knowledge that this type of network surge is imminent may help cloud providers to dynamically scale up and accommodate the surge.

In another example, there is currently no way for cloud providers to exchange data about user specific parameters, such as network conditions, and proactively adjust connection characteristics. For instance, a user attending a meeting hosted by one cloud provider may experience network issues, and that cloud provider may share this network state with other cloud providers. The other cloud providers may in turn proactively optimize their connection traffic towards that user until the network conditions improve.

In the above two examples, a multi-cloud SDN controller coordinating between the various cloud providers may be able to share information (e.g., user state, enterprise network state, etc.) just-in time or slightly ahead of time by learning about events from one of the cloud providers and sharing to other cloud providers. The shared data may include information that would help the other cloud providers to proactively install workflows to handle those conditions.

The techniques in this proposal insert a multi-cloud controller that uses the telemetry from one of the cloud providers, applies policies, and triggers appropriate work flows proactively on other cloud providers to which the same customer has subscribed. The multi-cloud SDN controller proactively learns insights about subscribers (e.g., enterprise networks, end users, other cloud providers, etc.) and based on the learned insights, the multi-cloud SDN controller applies dynamic policies on other cloud provides to which those subscribers are attached.

This proposal describes a mechanism by which a multi-cloud SDN controller proactively learns enterprise metadata (e.g., services provided by various cloud providers, health status of network access to each service, overall enterprise network health, etc.) and uses this enterprise metadata to proactively notify cloud providers and recommend actions. The notifications may trigger various workflows and actions in the cloud providers that provide the services to the enterprise users. The enterprise administrators may use the multi-cloud SDN controller to set up the workloads/actions that trigger based on events detected in the gathered metadata.

The exchange of telemetry between the enterprise network and the multi-cloud SDN controller may use JavaScript Object Notation (JSON) or Proto data elements with various parameters to provide the visibility for the multi-cloud SDN controller. The multi-cloud SDN controller may expose a Representational State Transfer (REST) Application Programming Interface (API) or a Proto element that may be implemented by clients to facilitate the transfer of metadata. The interface between the multi-cloud SDN controller and the cloud providers may also be REST APIs or bi-directional application interfaces that can send telemetry as JSON/Proto elements. An algorithm is also provided that a multi-cloud SDN controller can use to assimilate telemetry data from various sources and make a decision on what actions to take.

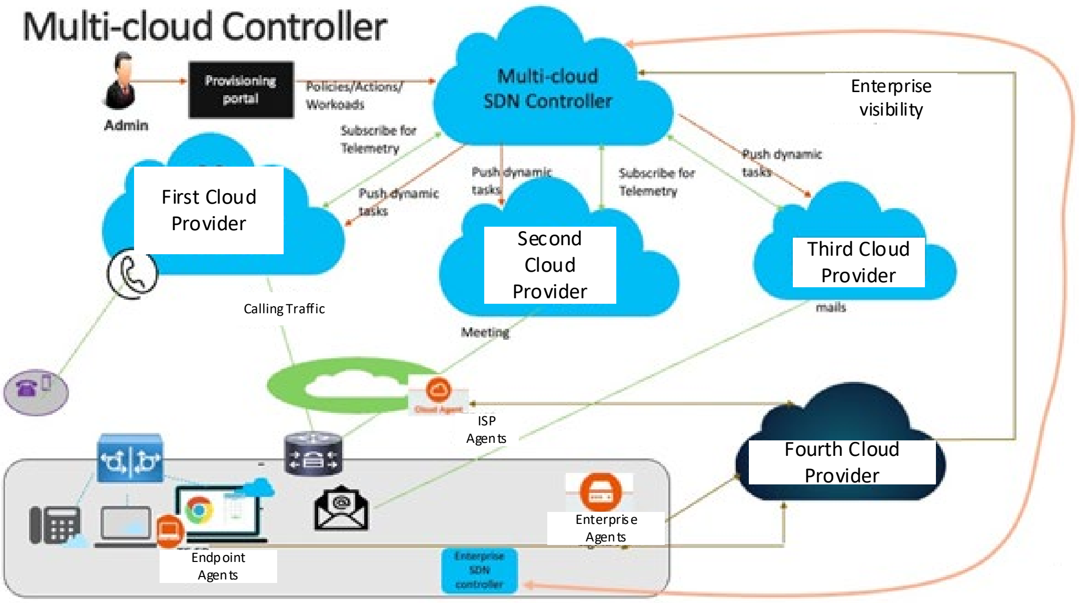
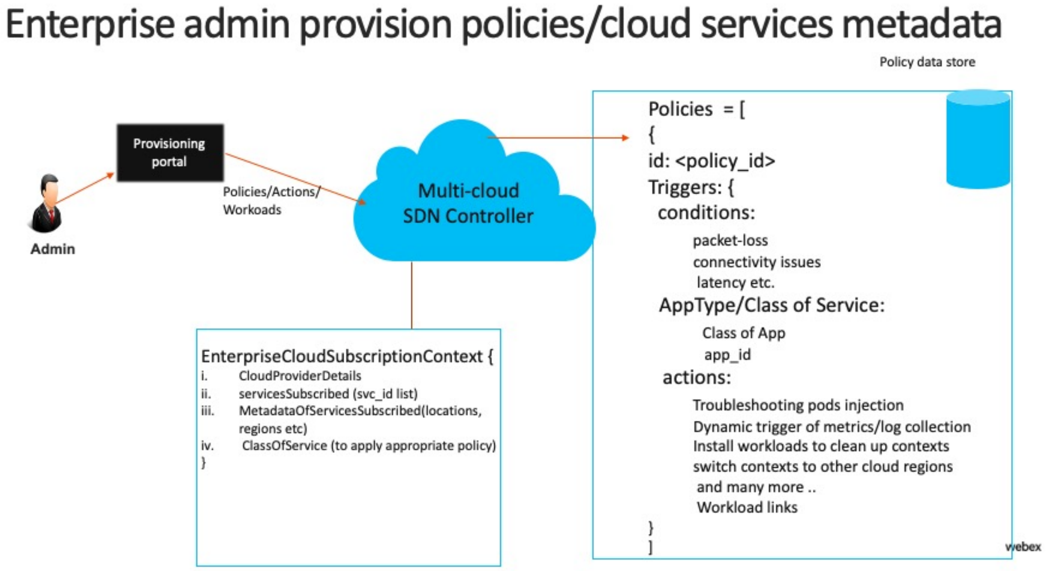


Figure 1 – Example of a Multi-cloud SDN Controller system

The example shown in Figure 1 illustrates steps that enable the multi-cloud SDN controller to improve the operation of multiple cloud providers supporting users from an enterprise network. Initially, the enterprise administrators subscribe to multiple cloud providers for various services. For example, as shown in Figure 1, the enterprise subscribes to a first, second, third, and fourth cloud provider for various different services. As the enterprise administrators subscribe to each could provider for each service, the enterprise controller/administrator pushes this subscription context to the multi-cloud SDN controller. In this instance, the multi-cloud SDN controller has a pre-established trust relationship with the various cloud providers. This may be set up by means of existing multi-cloud federation techniques (e.g., Mutual Transport Layer Security (mTLS) at transport, Identity Provider (IdP) chaining, Open Authorization (OAuth) integrations, etc.).

As shown in Figure 2, the enterprise administrators may also define policies and actions to trigger when policy thresholds are hit. For instance, the triggered actions may include applying security policies that block/throttle network traffic, installing additional workload modules dynamically to obtain more visibility (e.g., troubleshooting pods, workflows for additional metric collection, etc.), and/or installing additional workload modules dynamically to clean up user contexts or switch session contexts across cloud regions. The administrators may further define the types/classes of applications that are being consumed from cloud.



Enterprise admin provision policies/cloud services metadata

```
Policies  = [
{
id: <policy_id>
Triggers: {
 conditions:
        packet-loss
        connectivity issues
         latency etc.
 AppType/Class of Service:
          Class of App
          app_id
 actions:
        Troubleshooting pods injection
        Dynamic trigger of metrics/log collection
        Install workloads to clean up contexts
        switch contexts to other cloud regions
         and many more ..
         Workload links
}
]
```

```
EnterpriseCloudSubscriptionContext {
i.     CloudProviderDetails
ii.    servicesSubscribed (svc_id list)
iii.   MetadataOfServicesSubscribed(locations,
       regions etc)
iv.     ClassOfService (to apply appropriate policy)
}
```

Figure 2 – Enterprise Administrator Provisioning Multi-cloud SDN Controller

The multi-cloud SDN controller subscribes to each cloud provider for specific telemetry data based on the types/classes of applications that the enterprise consumes from each of the cloud providers. The enterprise administrator may also use a telemetry cloud service to push endpoint metrics and/or enterprise level metrics to the telemetry cloud service. The telemetry cloud service may also learn about the ISP connectivity points of the enterprise network and its metrics via ISP telemetry agents pushing information to the telemetry cloud service. All of this data would be sent to the multi-cloud SDN controller. The multi-cloud SDN controller will have visibility into enterprise ISP traffic pattern, the network state, the enterprise network health, endpoint metrics, as well as other information about entities affecting the enterprise network.

When an event happens, for example, when a portion of the enterprise users' data stored in one of their subscribed cloud providers is leaked, the affected cloud provider immediately notifies the multi-cloud SDN controller with information about the data leak (e.g., a list of users, type of data that was leaked, etc.). The multi-cloud SDN controller looks up the application subscription data of the affected users for applications from other cloud providers and notifies the other cloud providers about the incident.

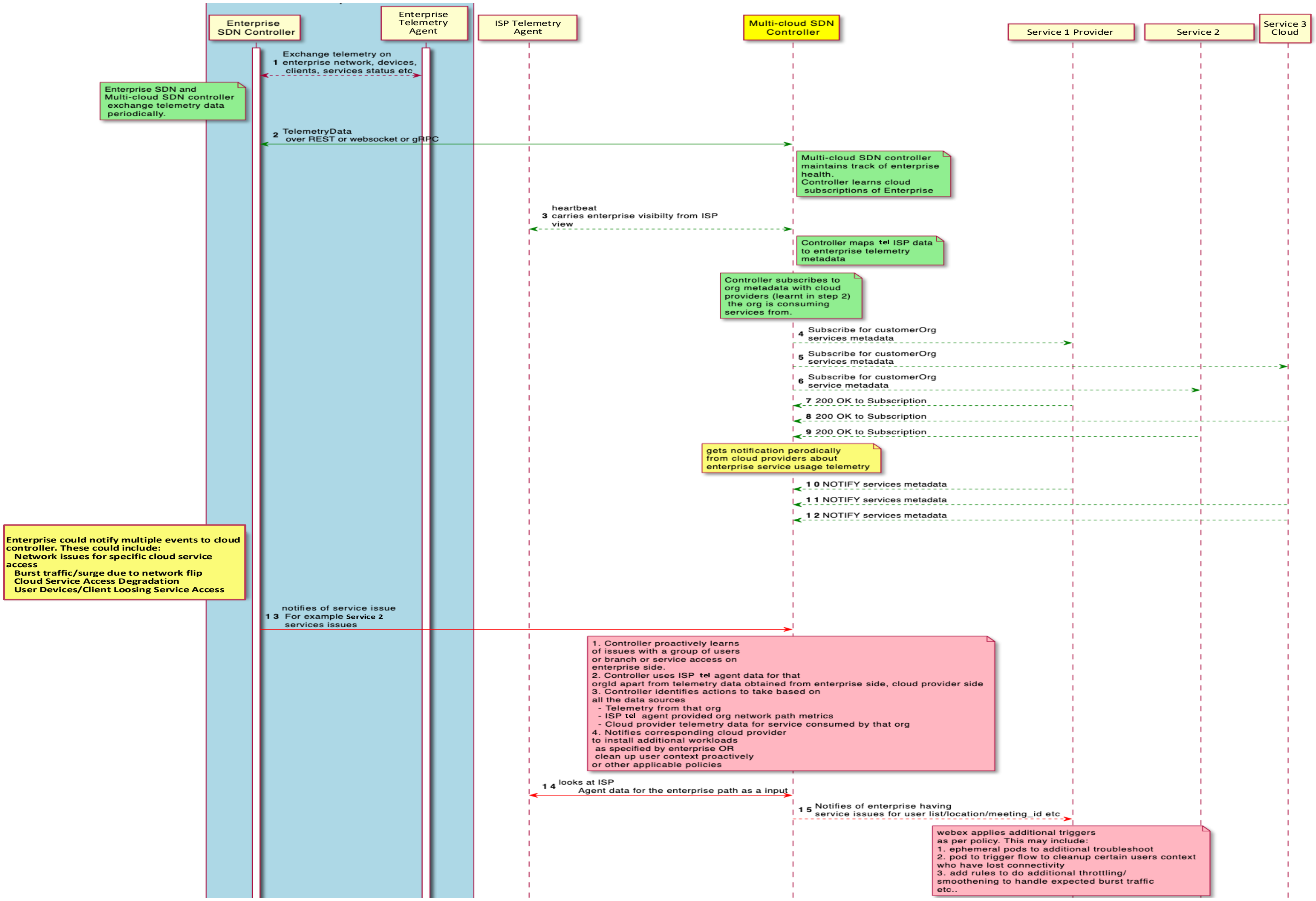
Based on the notification from the multi-cloud SDN controller, the other cloud providers may take one or more actions to mitigate any damage caused by the data leak. The other cloud providers may dynamically install additional defenses, such as ephemeral pods on the ingress, that will do a deeper inspection of inbound traffic from the affected user client devices. The actual insertion of this dynamic workload may use existing techniques like ephemeral pods/containers. Additionally, the cloud providers may collect additional telemetry data/logs for the affected users and trigger actions based on that telemetry. Furthermore, the cloud providers may restrict access to sensitive data/applications for the affected users.

In another example, when the enterprise network experiences an ISP outage/network issue, the telemetry cloud provider that monitors the ISP connection points to the enterprise network may notify the multi-cloud SDN controller of this event. The telemetry cloud provider may also have visibility into backup paths for that ISP or backup ISPs for that enterprise and may also notify the multi-cloud SDN controller of the available backup options. The multi-cloud SDN controller notifies each of the cloud providers of this event along with event metadata, such as the type of event (network outage), the client region affected, any backup ISP paths, or other relevant information.

The cloud providers may take actions to proactively handle the network outage event before the network issues significantly impact the operation of the subscribed functions of the cloud providers. For instance, the cloud providers may initiate a warm handoff of any active sessions to other cloud regions associated with the backup ISP path of the enterprise network. Additionally, the cloud providers may trigger workflows to proactively clean up user sessions will fail due to the network outage.

In another example, when one of the cloud providers that provide services to the enterprise notices a surge in network traffic (e.g., a higher-than-normal connection request, burst traffic, etc.), that cloud provider may notify the multi-cloud SDN controller, which in turn may notify the other cloud providers. This advance notification may allow the other cloud providers to scale up and apply additional modules to handle this anticipated surge in usage.

Figure 3, below, shows an example workflow illustrating actions that the multi-cloud SDN controller may take based on various data points. The multi-cloud SDN controller assimilates telemetry data from various sources (e.g., enterprise network controllers, telemetry agents, cloud providers to which the enterprise is subscribed, etc.) and determines appropriate actions to take.


Enterprise SDN Controller
Enterprise Telemetry Agent
ISP Telemetry Agent
Multi-cloud SDN Controller
Service 1 Provider
Service 2
Service 3 Cloud

1 Exchange telemetry on enterprise network, devices, clients, services status etc

Enterprise SDN and Multi-cloud SDN controller exchange telemetry data periodically.

2 TelemetryData over REST or websocket or gRPC

Multi-cloud SDN controller maintains track of enterprise health.
Controller learns cloud subscriptions of Enterprise

3 heartbeat carries enterprise visibilty from ISP view

Controller maps tel ISP data to enterprise telemetry metadata

Controller subscribes to org metadata with cloud providers (learnt in step 2) the org is consuming services from.

4 Subscribe for customerOrg services metadata

5 Subscribe for customerOrg services metadata

6 Subscribe for customerOrg service metadata

7 200 OK to Subscription

8 200 OK to Subscription

9 200 OK to Subscription

gets notification perodically from cloud providers about enterprise service usage telemetry

10 NOTIFY services metadata

11 NOTIFY services metadata

12 NOTIFY services metadata

Enterprise could notify multiple events to cloud controller. These could include:
Network issues for specific cloud service access
Burst traffic/surge due to network flip
Cloud Service Access Degradation
User Devices/Client Loosing Service Access

13 notifies of service issue For example Service 2 services issues

1. Controller proactively learns of issues with a group of users or branch or service access on enterprise side.
2. Controller uses ISP tel agent data for that orgId apart from telemetry data obtained from enterprise side, cloud provider side
3. Controller identifies actions to take based on all the data sources
 - Telemetry from that org
 - ISP tel agent provided org network path metrics
 - Cloud provider telemetry data for service consumed by that org
4. Notifies corresponding cloud provider to install additional workloads as specified by enterprise OR clean up user context proactively or other applicable policies

14 looks at ISP Agent data for the enterprise path as a input

15 Notifies of enterprise having service issues for user list/location/meeting_id etc

webex applies additional triggers as per policy. This may include:
1. ephemeral pods to additional troubleshoot
2. pod to trigger flow to cleanup certain users context who have lost connectivity
3. add rules to do additional throttling/ smoothening to handle expected burst traffic
etc..

Figure 3 – Example workflow of multi-cloud SDN controller

In summary, the techniques described in this proposal include a method for a multi-cloud SDN controller to learn various metadata parameters about cloud services to which an enterprise subscribes. These parameters may include service identifiers, region, type/class of service, desired Quality of Service (QoS), and the number and identity of users. Additionally, the multi-cloud SDN controller obtains the enterprise view of the performance of the network access/quality via the enterprise SDN controller and enterprise telemetry agents running at various points in the enterprise network. The various cloud providers may also provide service usage telemetry data. The multi-cloud SDN controller correlates the data gathered from the enterprise network controller and telemetry gathered from the cloud providers and derives actionable events. Based on the events determined from the correlated data, the multi-cloud SDN controller invokes the appropriate policy.