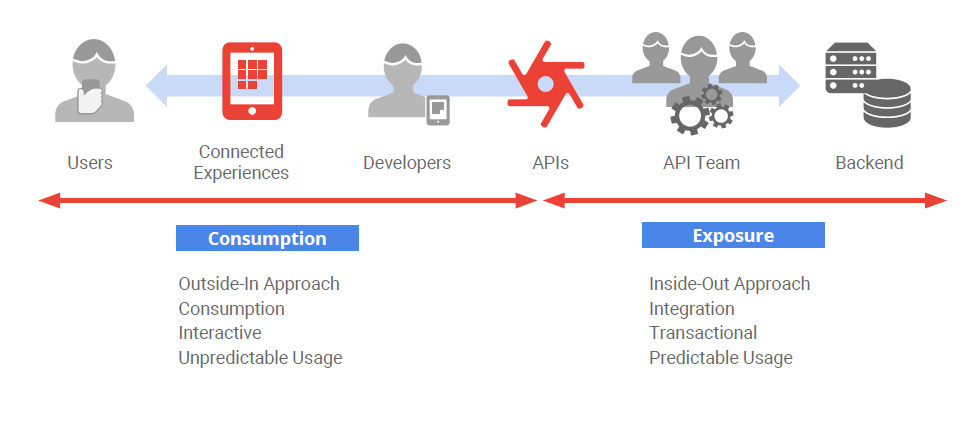
What is Apigee ?

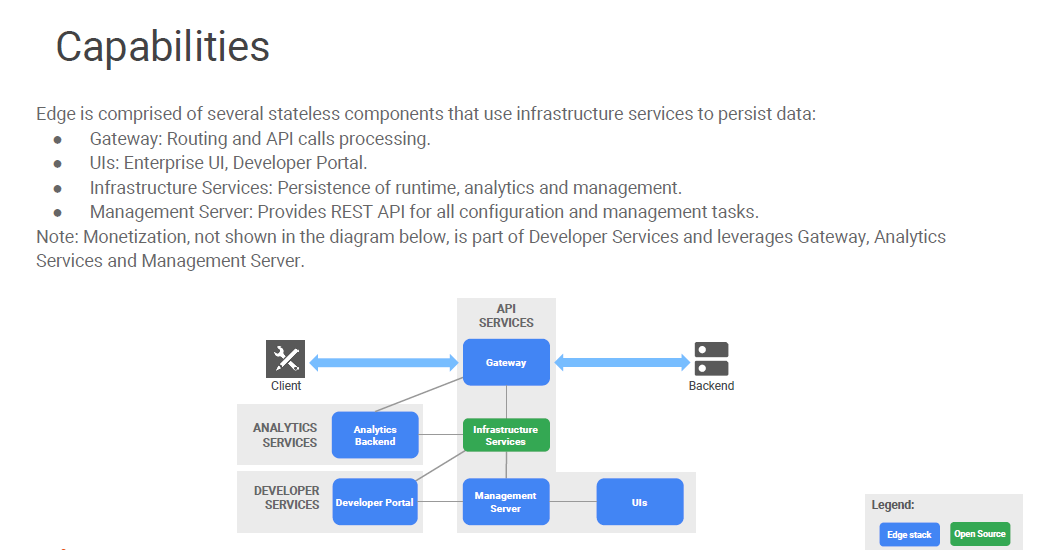
Apigee is an API gateway management tool offered by Google to exchange data across cloud services and applications. Since many sites and services are delivered by way of RESTful APIs, API gateways are used to connect these sites and services in order to provide data feeds and bolster communication abilities. Apigee is one of tool that can manage the API gateway and make it easier to produce and deploy modern, developer-friendly apps.

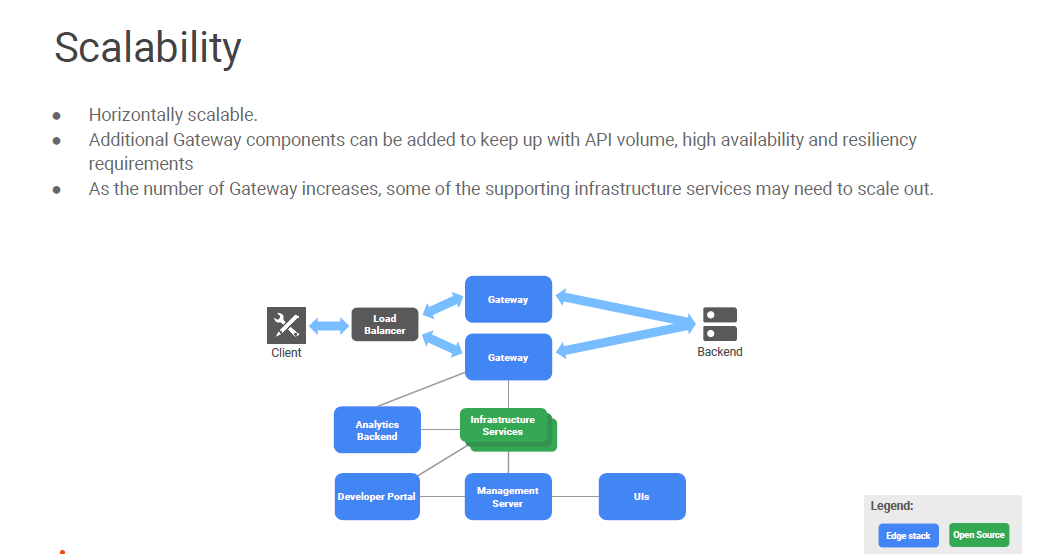
Why is Apigee ?

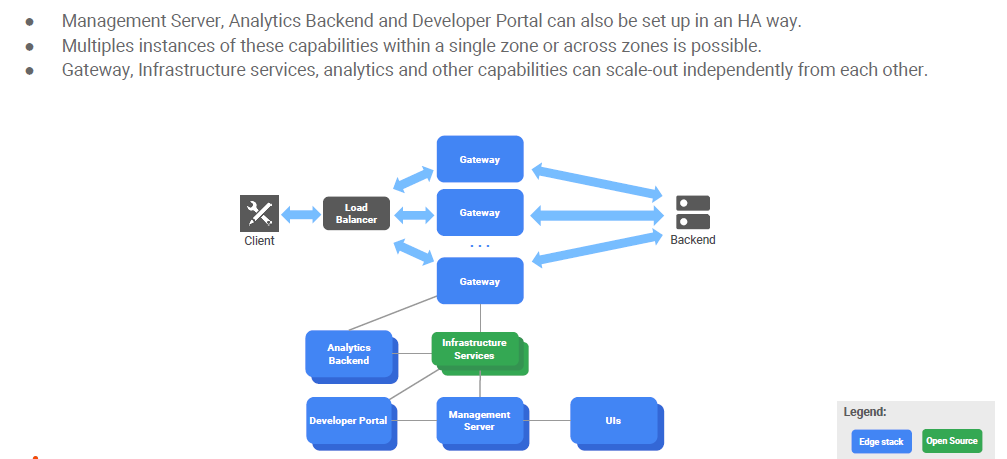
Companies today want to make their backend services available on the web so that these services can be consumed by apps running on mobile devices and desktops. A company might want to expose services that provide product pricing and availability information, sales and ordering services,order tracking services, and any other services required by client apps.

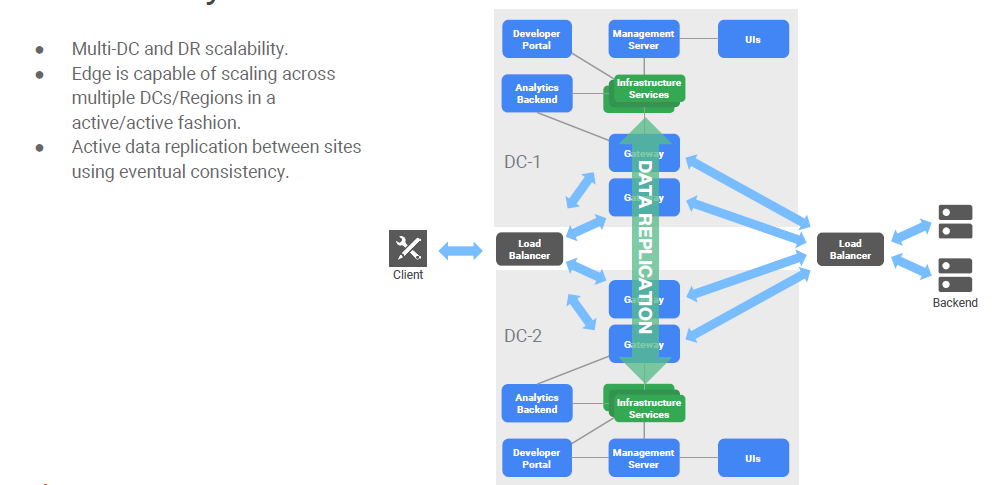
What is Apigee Edge?

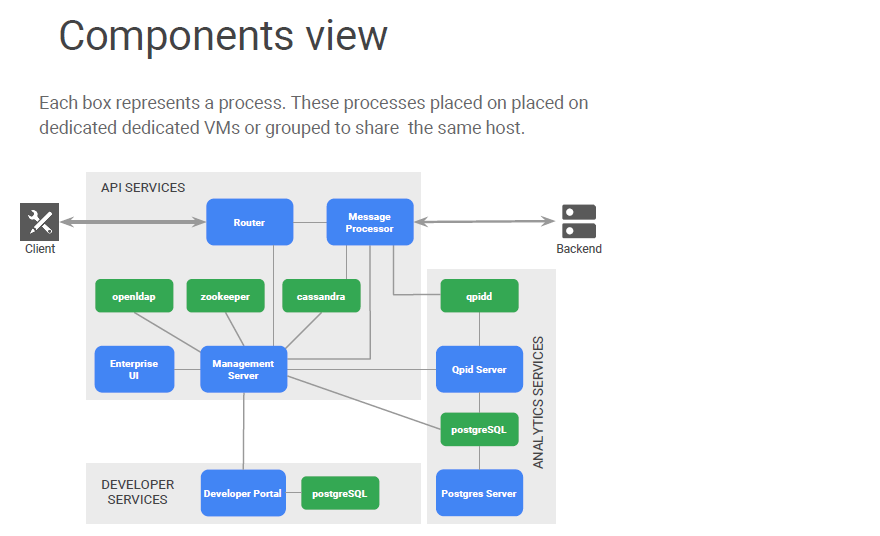


For current documentation related to Apigee: <https://wiki.west.com/pages/viewpage.action?pageId=137115984&src=contextnavpagetreemode> 









Components:

In Apigee , The full critical path consists of the router, message processor, and cassandra services. As long as these components are up and running, others can be down or unreachable without affecting API runtime availability.

Router: Router is responsible for exposing virtual hosts and load balancing incoming requests from clients.

Message processor: The message processor represents the runtime container for APIs executed on Apigee. API calls executing on the message processors may perform one or more calls to backend systems. All calls to backend systems on Apigee are originated by message processors.

Cassandra: Some APIs generate runtime data that needs to be stored within Apigee. API policies such as API key validation, OAuth, key value map, and cache require access to a data store for the storage and retrieval of runtime data. The Apigee runtime data store is cassandra. Cassandra is part of the critical path because it is used by API policies to store data during runtime execution.

API calls executed on the message processors generate analytics events. These events are asynchronously generated and consumed.

Qpidd: The first component in the analytics pipeline is apache Qpid. Qpid is a queue broker, queues are provisioned here to store analytics data.

Qpid Server: Qpid server is responsible for moving data from Qpid queues to postgreSQL.

PG for analytics: Analytics raw data on postgreSQL are eventually aggregated in batches by a service called postgres server. Aggregate data are used to power some Apigee Edge analytics reports.

Management server: The management server exposes the management API. This API allows to add users, deploy APIs, and perform many other actions. While performing these actions, the management server leverages cassandra, zookeeper, and openldap to read and store relevant data associated with runtime or configuration state.

Zookeeper: Zookeeper stores data related to the configuration of the system. The wiring information between components and configuration state are stored in zookeeper.

LDAP/openLDAP: Openldap stores information related to role-based access control users, roles, and permissions used for administrative access to Apigee are stored here.

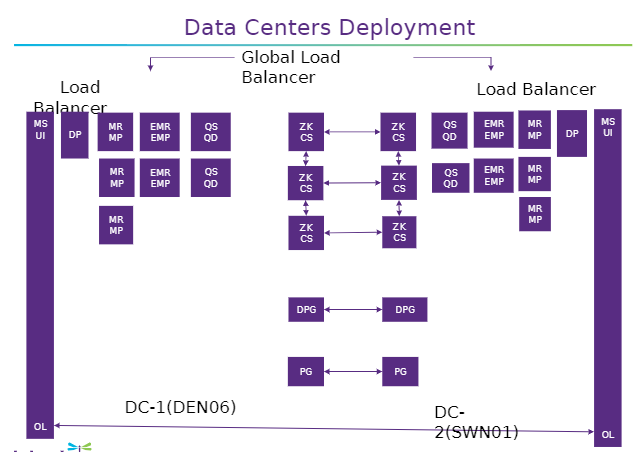
Edge UI: The Enterprise UI is the administrative UI used by API teams to develop and manage APIs. This UI consumes the management API to perform all actions. All features in the Enterprise UI are available through the management API.

Dev Portal:This portal is a Drupal based application. Drupal uses postgreSQL to store data about user accounts, sessions, and modules.

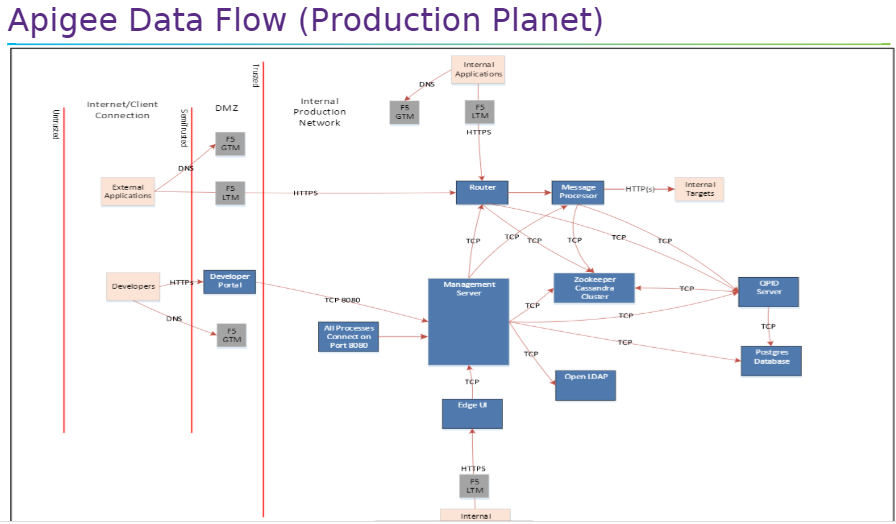
Dev Portal PG: Dev Portal Data base which store all the sessions, modules , user accounts info etc. related to Portal.

Note that : openLDAP, Qpidd, Cassandra, zookeeper and PostgreSQL are open source

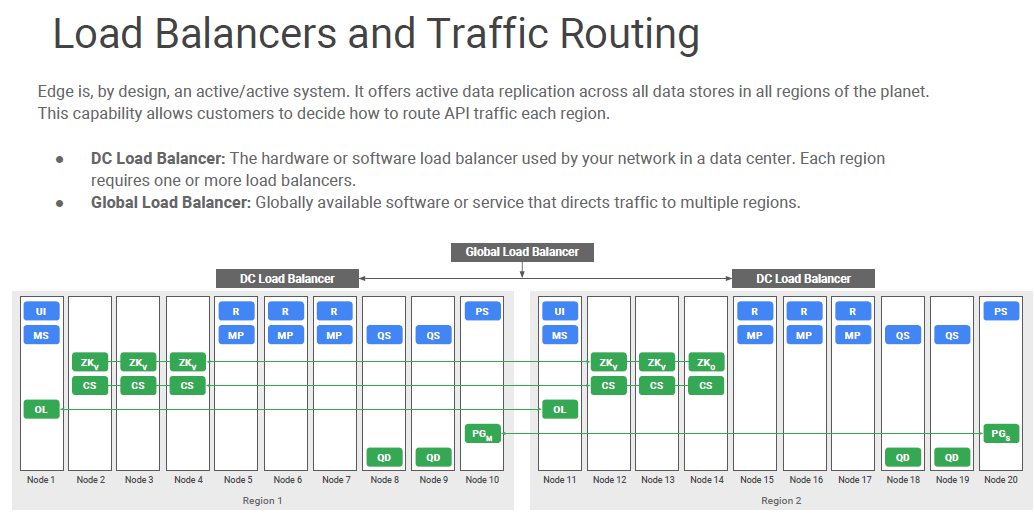
Apigee Data Center deployment diagram:

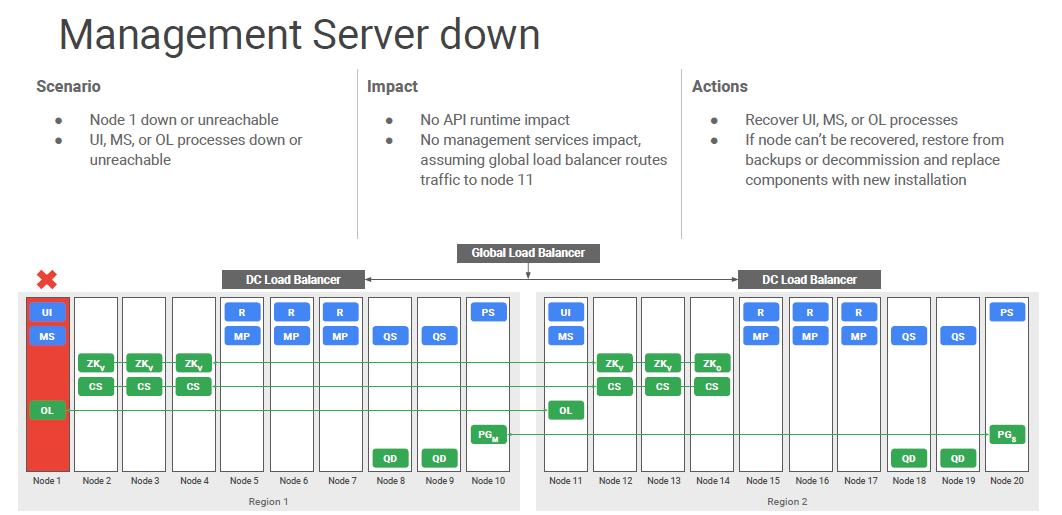


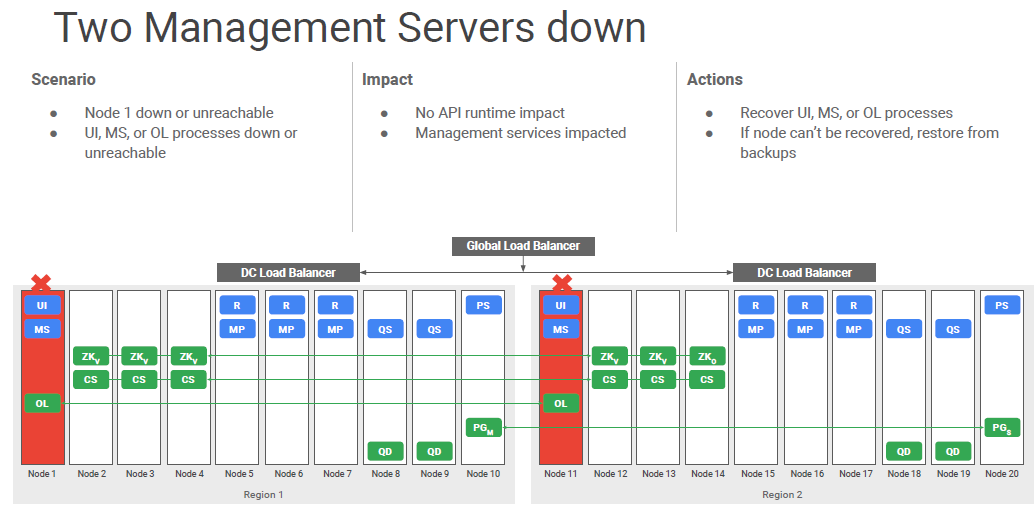
Apigee Data flow diagram:

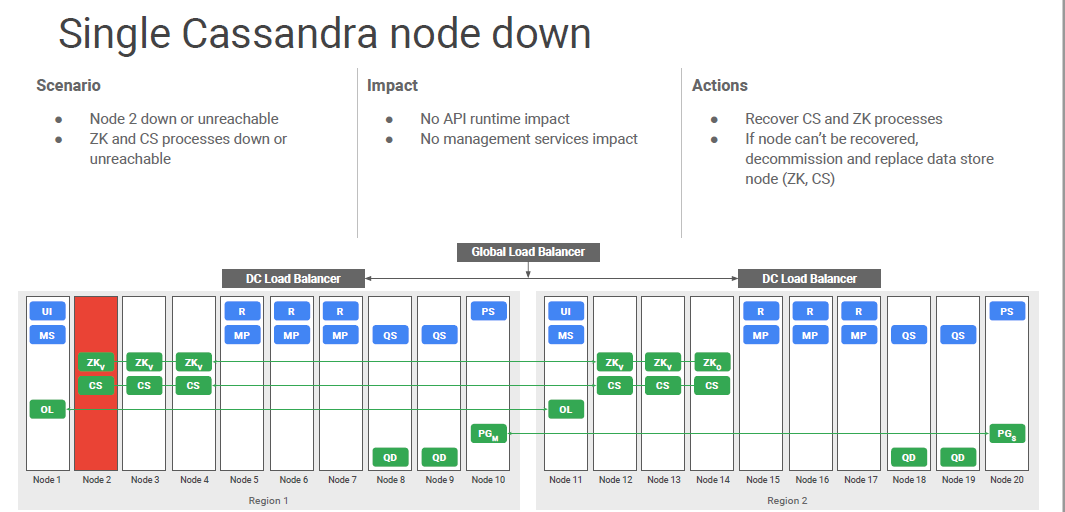


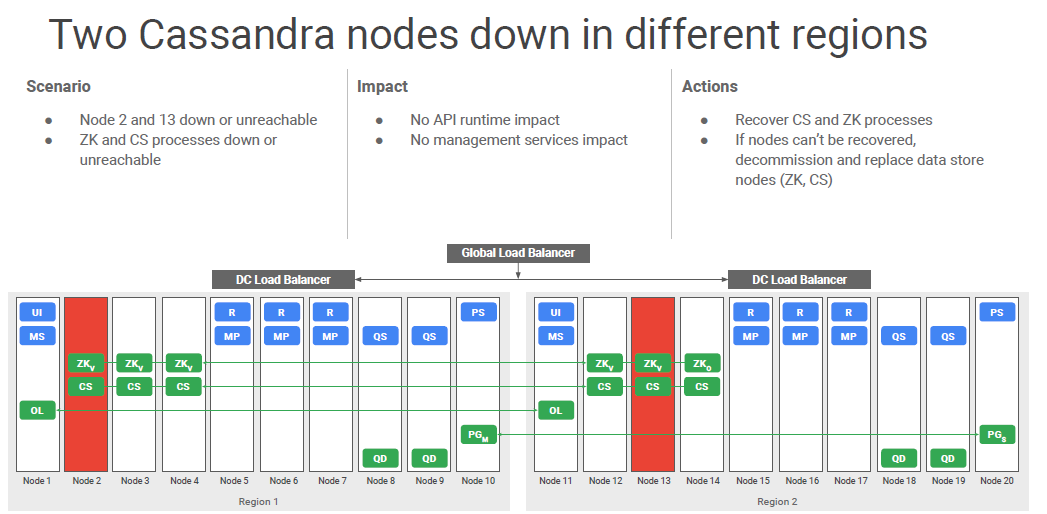
Failover and Disaster Recovery:

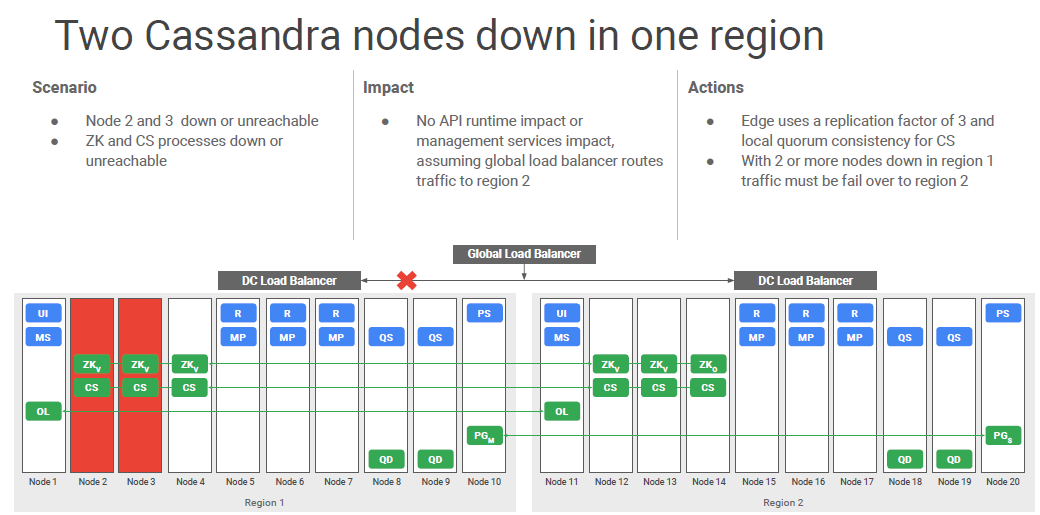


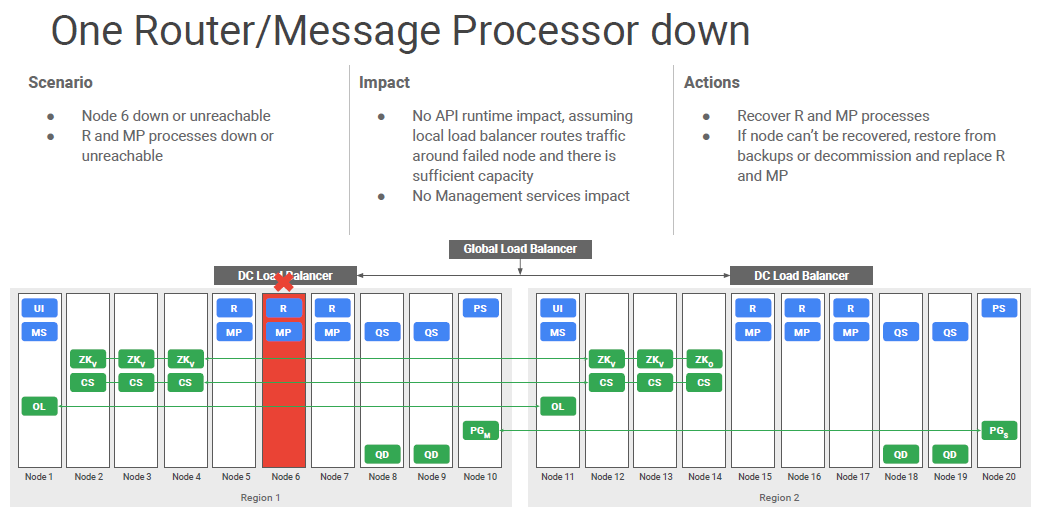


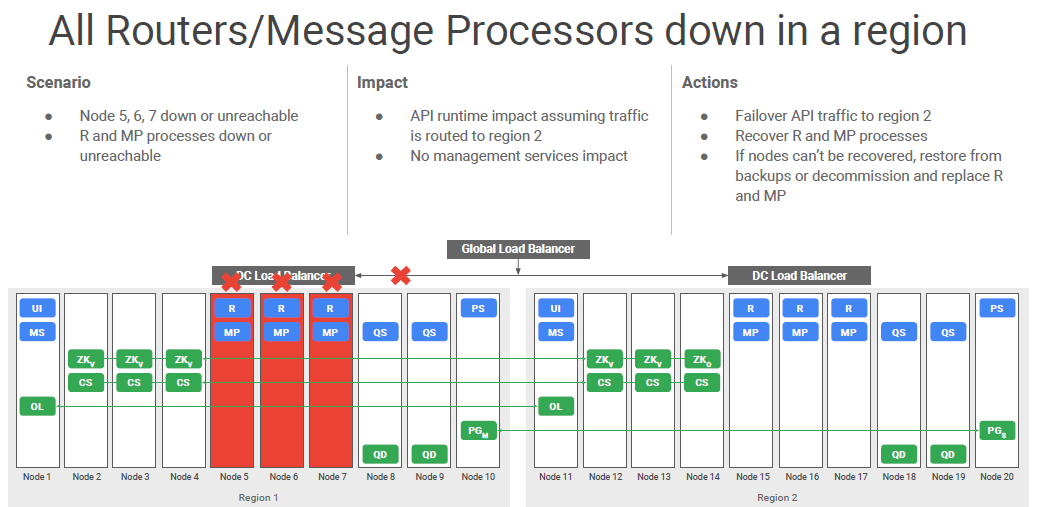


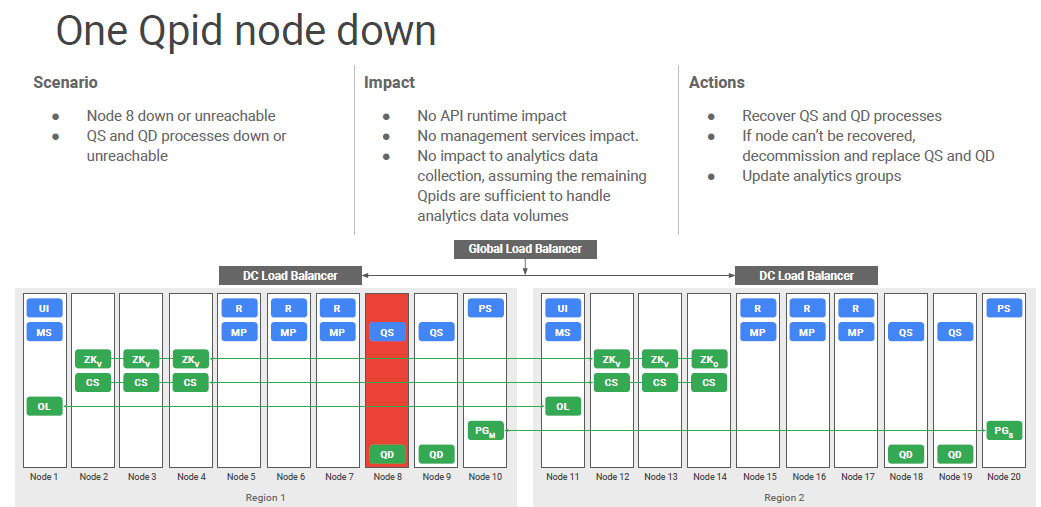


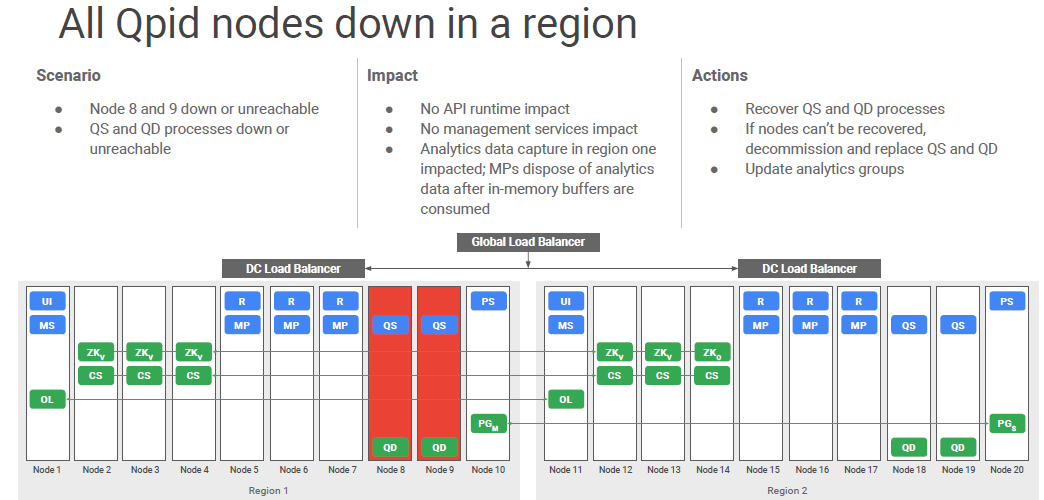


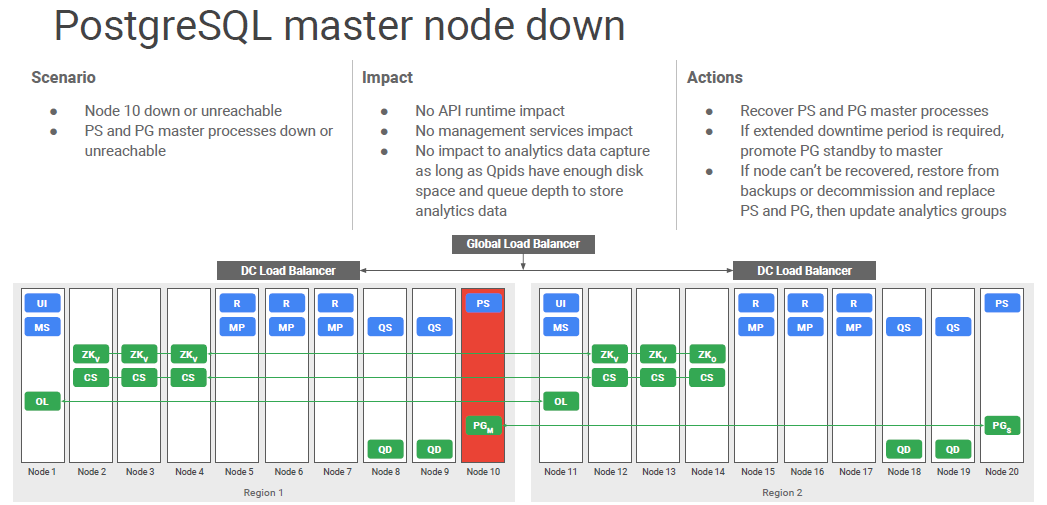


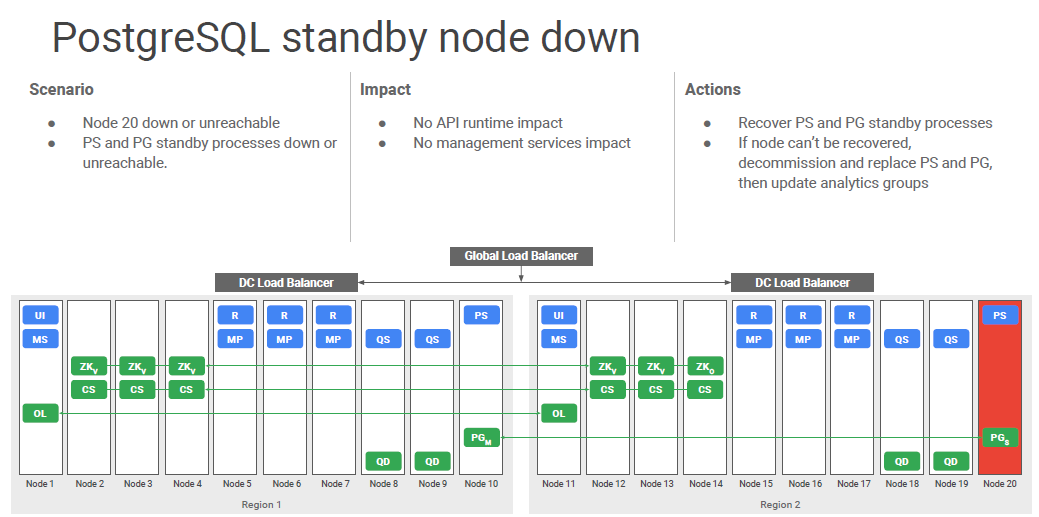


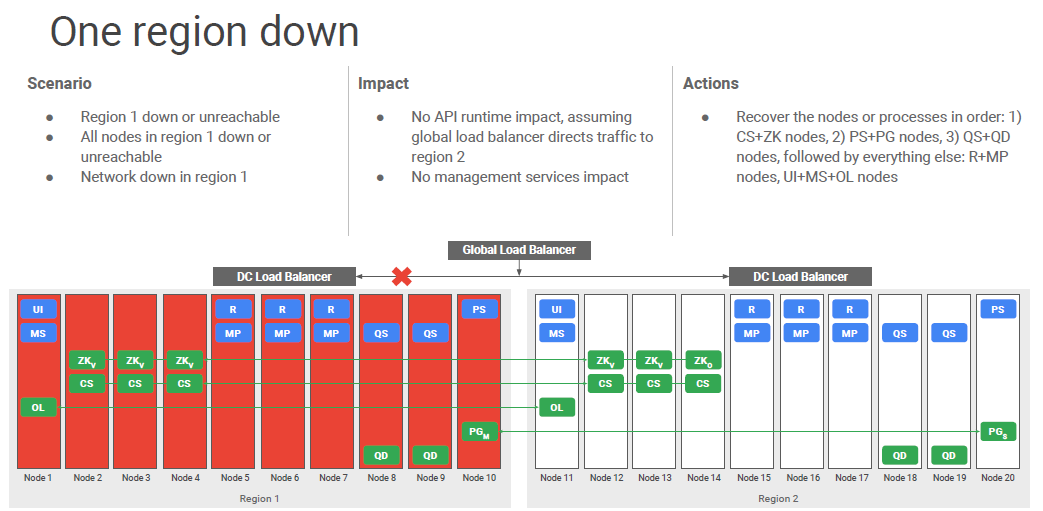






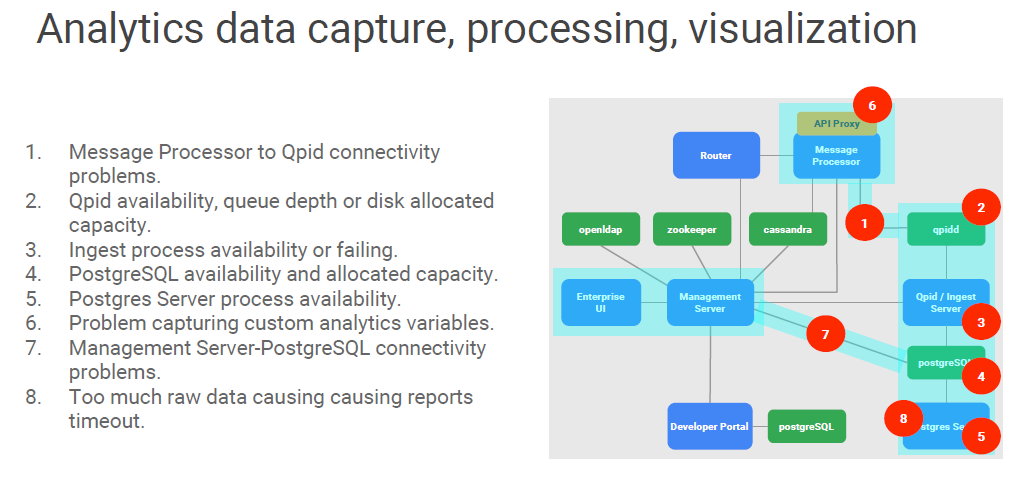


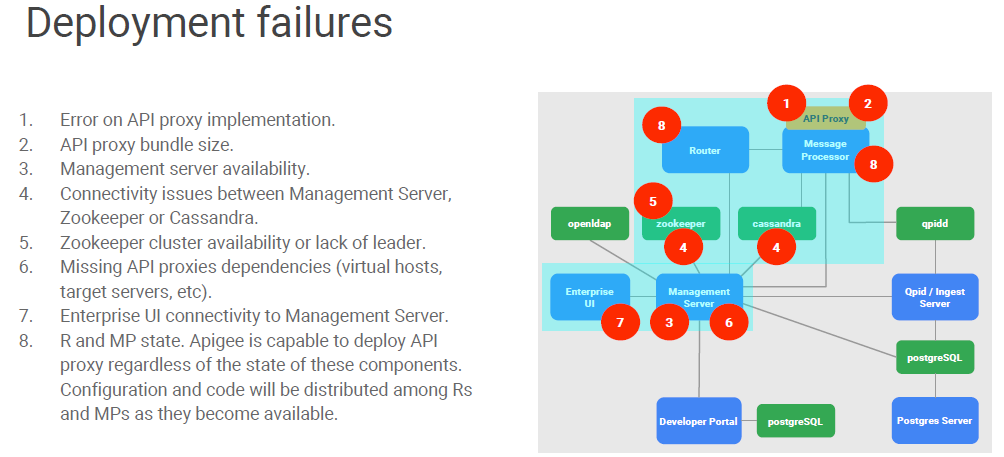


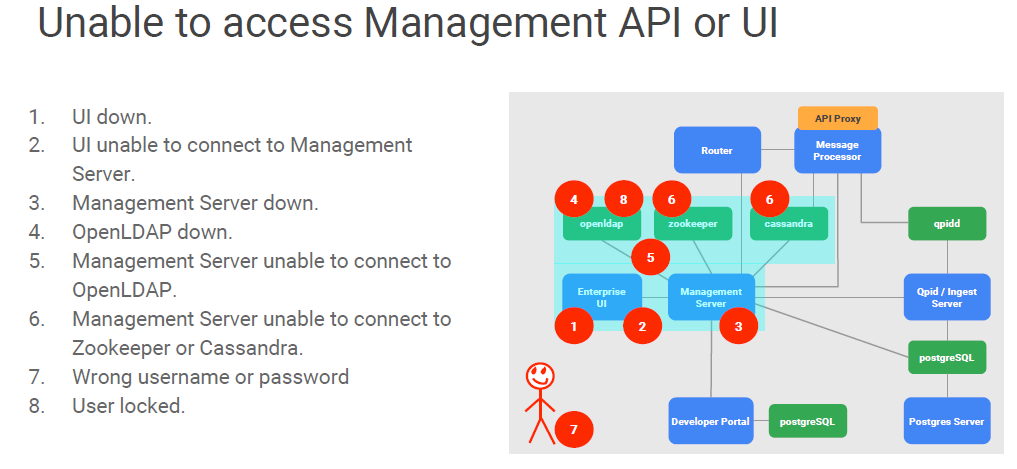


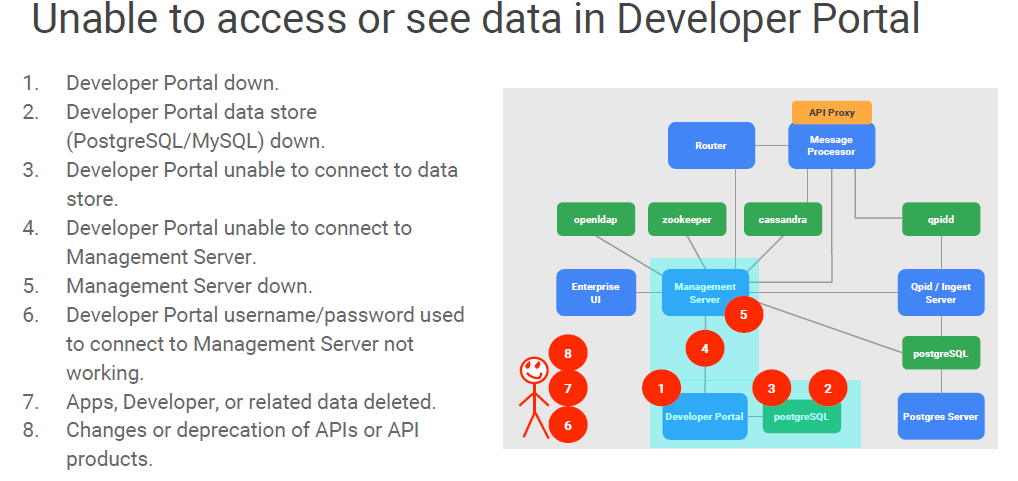


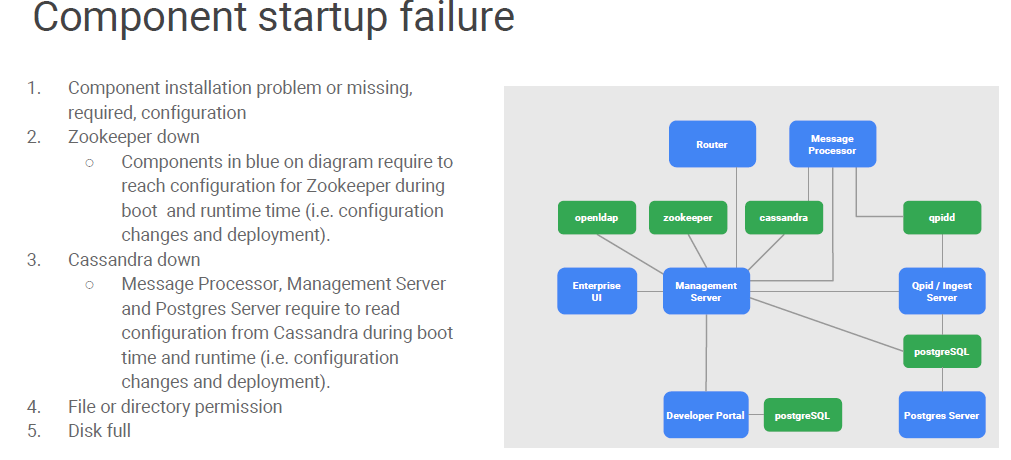
Troubleshooting Components:











!!!! Thank You!!!!