



Aug, 2015

# Developer Bootcamp

## Edge Overview and Architecture





# Course Topics

---

## High-Level Introduction

- Apigee Edge API Services
- Apigee Edge Backend as a Service (BaaS)
- Analytics and Developer Services

## Architectural Details

- Overview
- High Level Architecture
- Apigee Components View

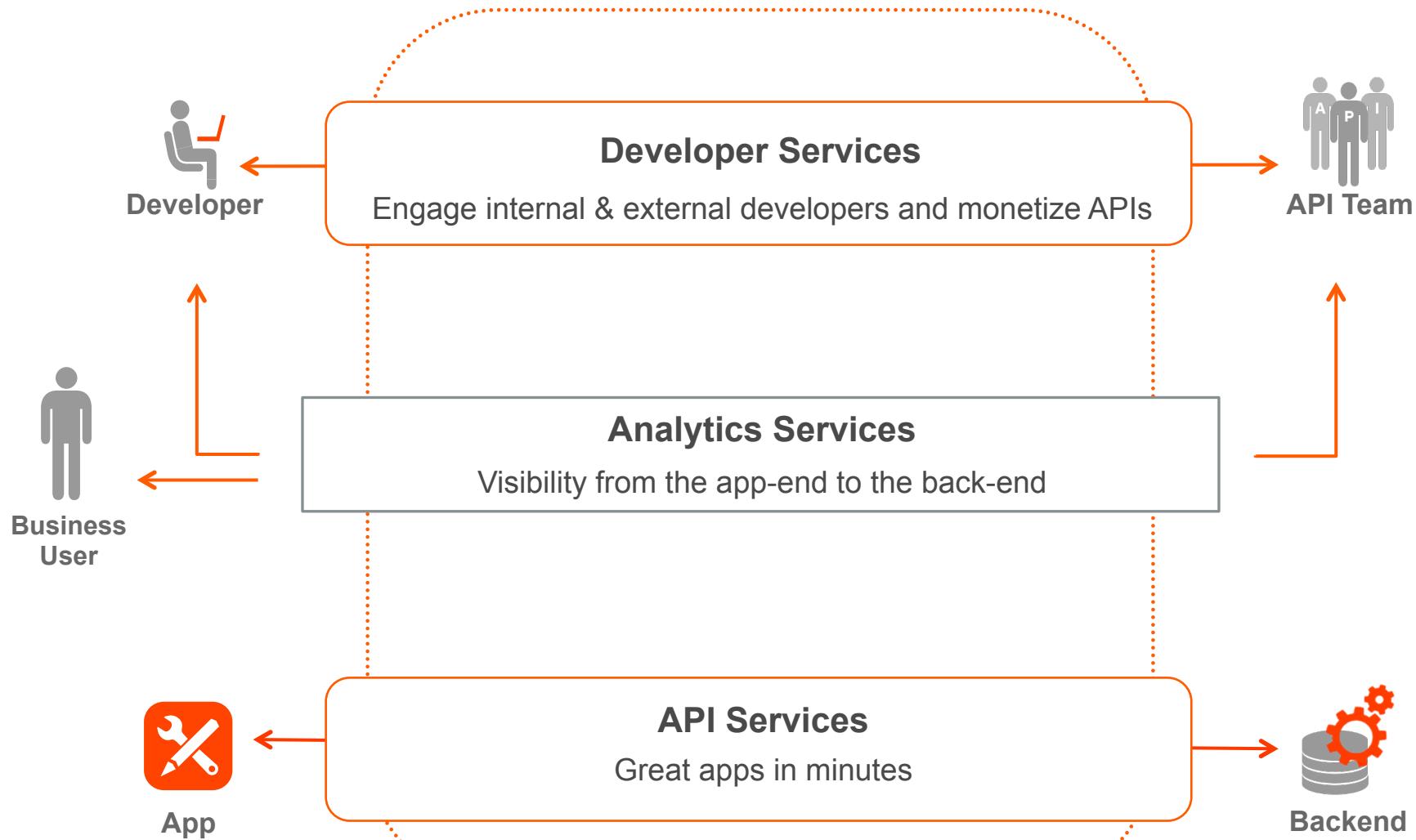
## Concepts

- Organizations and Environments
- Developers, Apps, API Products
- API Proxies
- Flows, Conditions, Variables

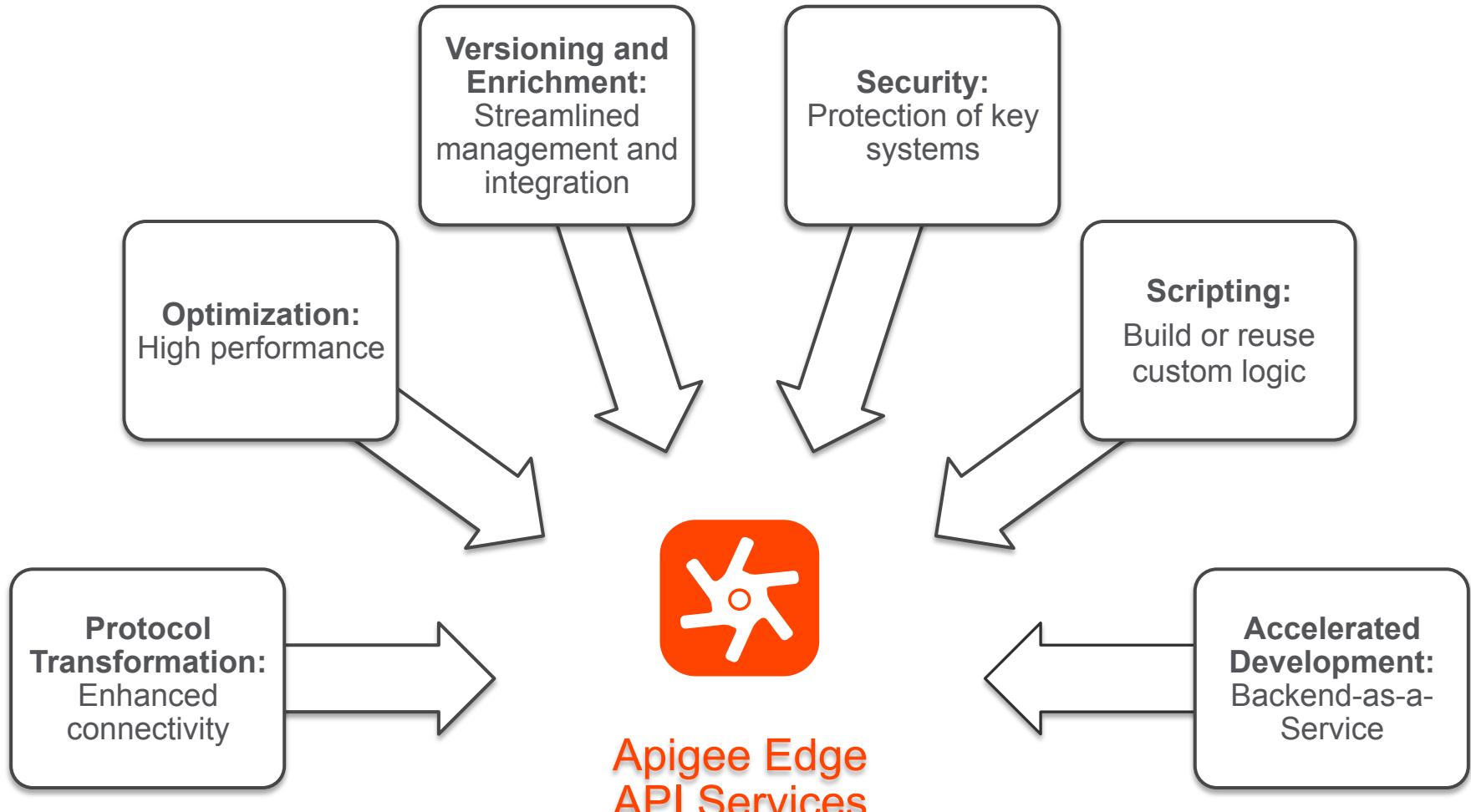
# High-Level Introduction



# Apigee Edge



# Apigee Edge – API Services



# API Services: Protocol Mediation and Mobile Optimization



## Transport Protocol

SOAP,  
REST

JMS

## Message Protocol

XML

JSON

## Streaming

Chunked

Full  
Streaming

## Cache

Distributed  
Cache

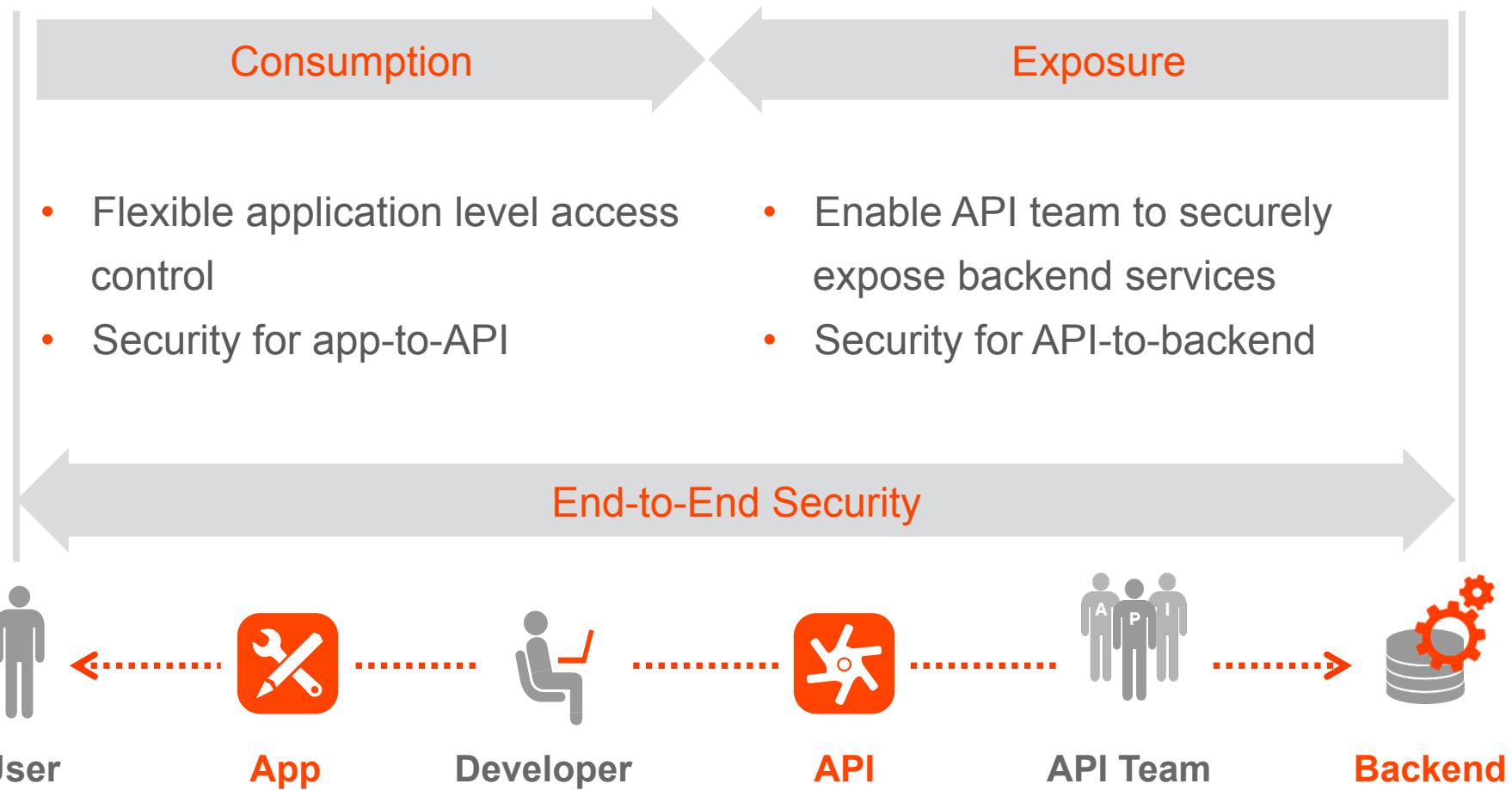
Response  
Cache

## Distributed Network

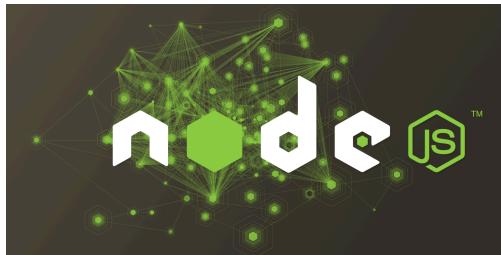
Content-  
Based  
Routing

Distributed  
“Smart”  
Routing

# Apigee Edge End-to-End Security



# Configuration or Coding... or Both



apigee   Dashboard   APIs   Publish   Analytics   Insights   Admin   Help   cvonsee@apigee.com - API Platform -

Dashboard / API Proxies / advapi-token-gen

### advapi-token-gen

Project Save Revision 1 New New Policy Attach Policy Deployment Overview Develop Trace

**Navigator**

- Policies
  - Fault\_InvalidCredentials
  - GenerateApigeeAccessToken\_CCG
  - InvalidateRefreshToken
  - ExtractCredentials
  - DEBUG
  - SetTokenTargetUrl
  - SetFinalResponse\_CCG
  - ConvertExpiresToString
  - SetAppServicesTokenRequestPayload
  - SetFinalResponse\_PG
  - Fault\_GenericFault
  - SetAppServicesTokenRequestPayload
  - ExtractAccessTokenFromTokenResponse
  - DecodeAuthorizationHeader
  - SetAuthorizationHeader
  - SetFinalResponse\_REFR
  - SetAppServicesTokenRequestPayload
  - SetFinalResponse\_INVALIDATE
  - Fault\_InvalidRequest
  - GenerateApigeeAccessToken\_PG
  - ExtractDatastoreInfo
  - GenerateApigeeAccessToken\_REFR
  - VerifyAPIKeyFromAuthHeader
  - Fault\_NotFound
  - InvalidateAccessToken
- Proxy Endpoints
  - default
    - ALL PreFlow
    - POST token-pg
    - POST token-ccg
    - DELETE token-invalidate
    - Cond defaultFlow
    - ALL PostFlow
  - Target Endpoints
    - default
      - ALL PreFlow
      - ALL PostFlow
- Scripts
- Python
  - DecodeAuthorizationHeader nv

Deployed to Environment: Not deployed

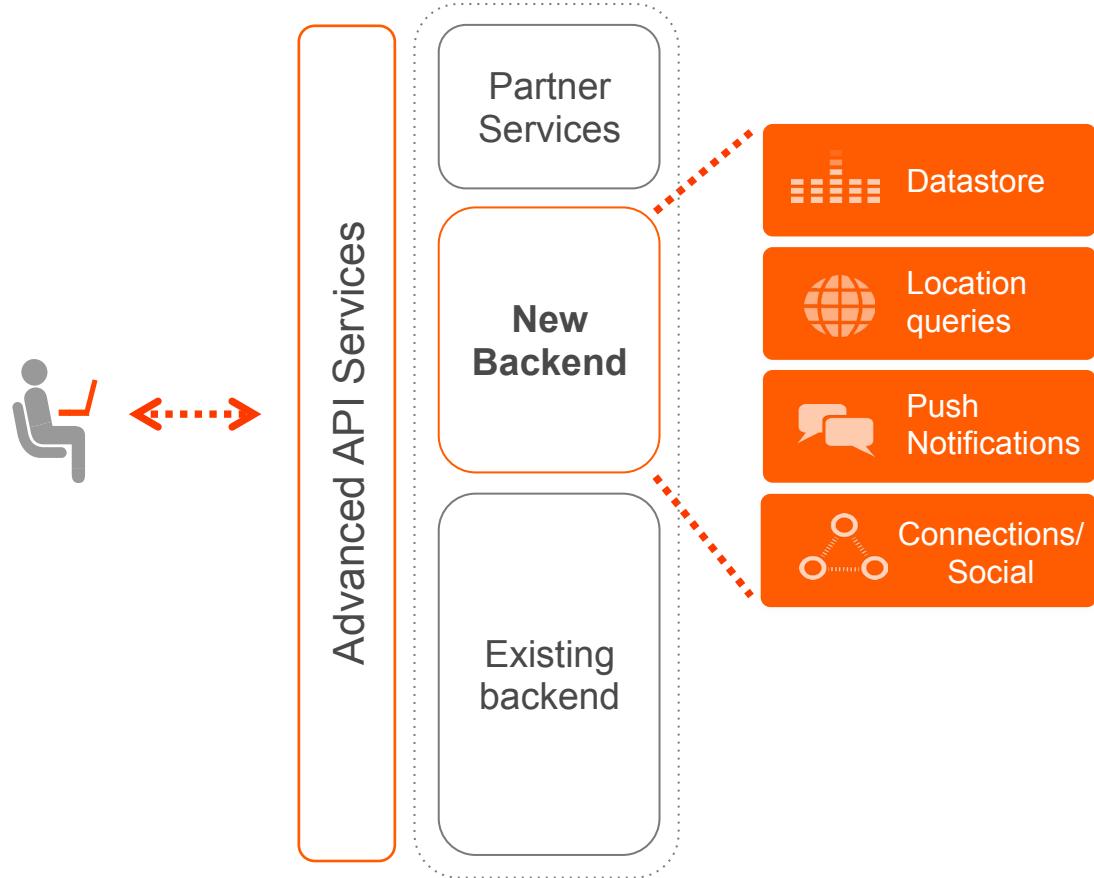
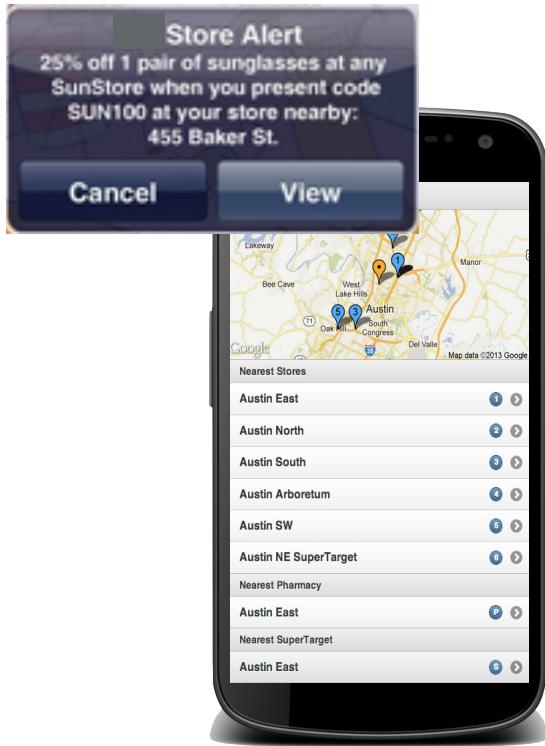
Copyright © 2014 Apigee Corp. All rights reserved V 4.24.7

# Node.js for building new APIs



- **Build highly-customized *standalone APIs*** by leveraging Apigee's integrated node.js as your back-end system
- **Solve complex orchestration or mobile optimization problems** by combining Apigee policies with the power of a scriptable target endpoint
- Use many of the thousands of **third-party node.js modules** in your APIs **without modification**
- Leverage Apigee's **world-class cloud operations**

# Apigee Edge API Backend-as-a-Service (BaaS)

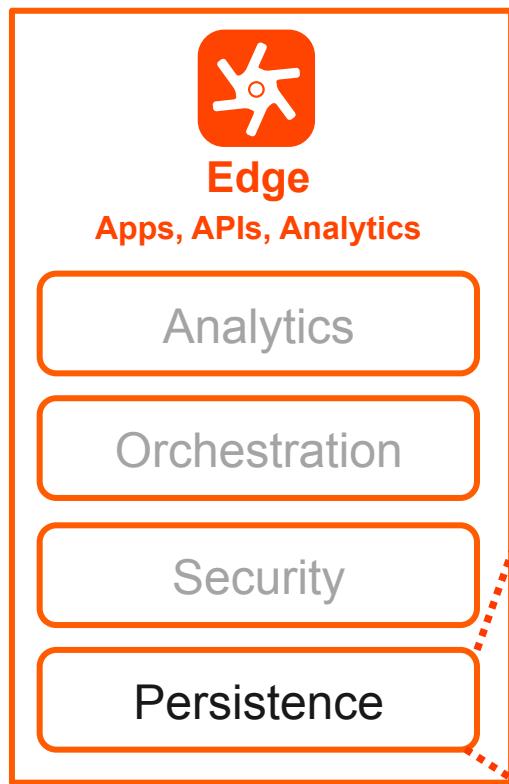


SDKs for:

- iOS
- Android
- JavaScript
- Ruby
- Node.js
- Microsoft .Net

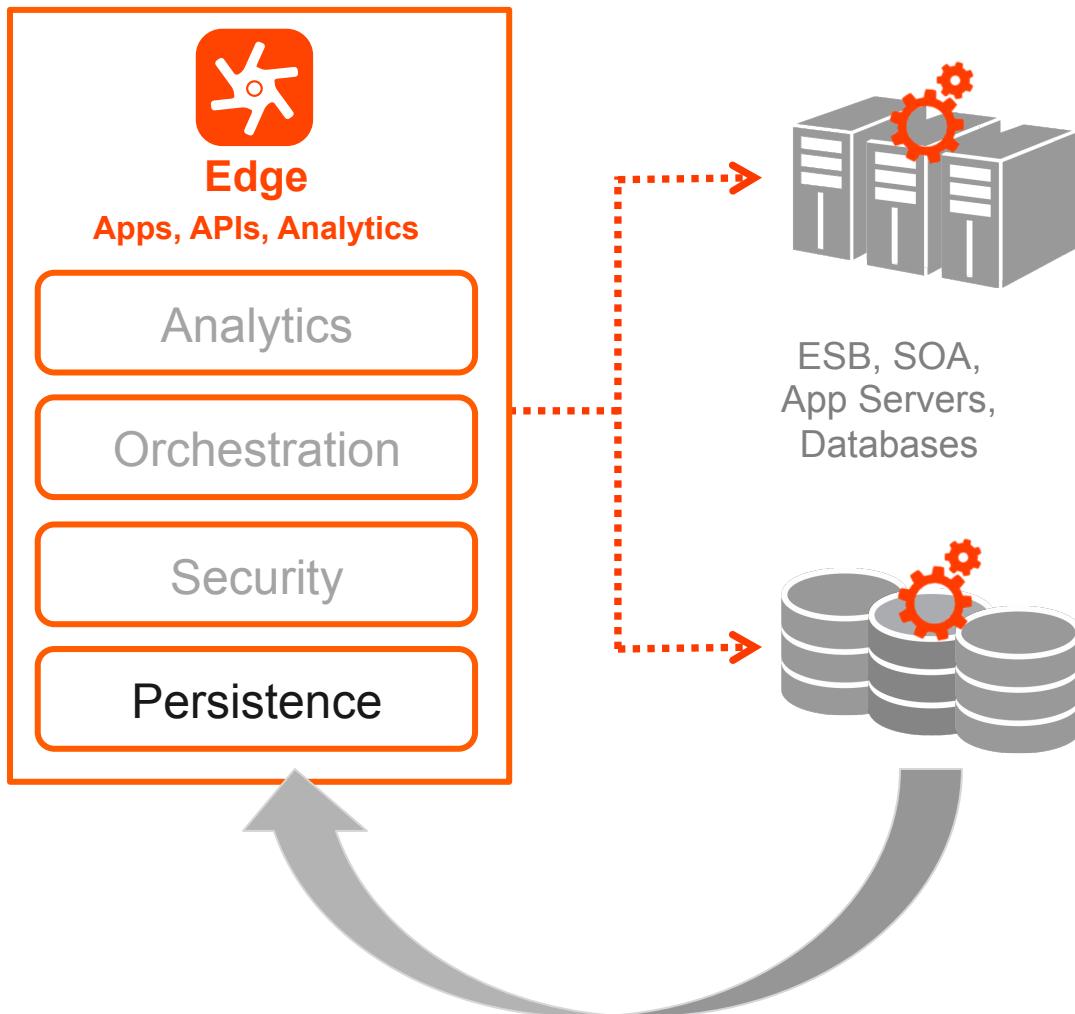
Faster and cost-effective building  
of new back-ends as APIs

# Apigee Edge API Backend-as-a-Service (BaaS)



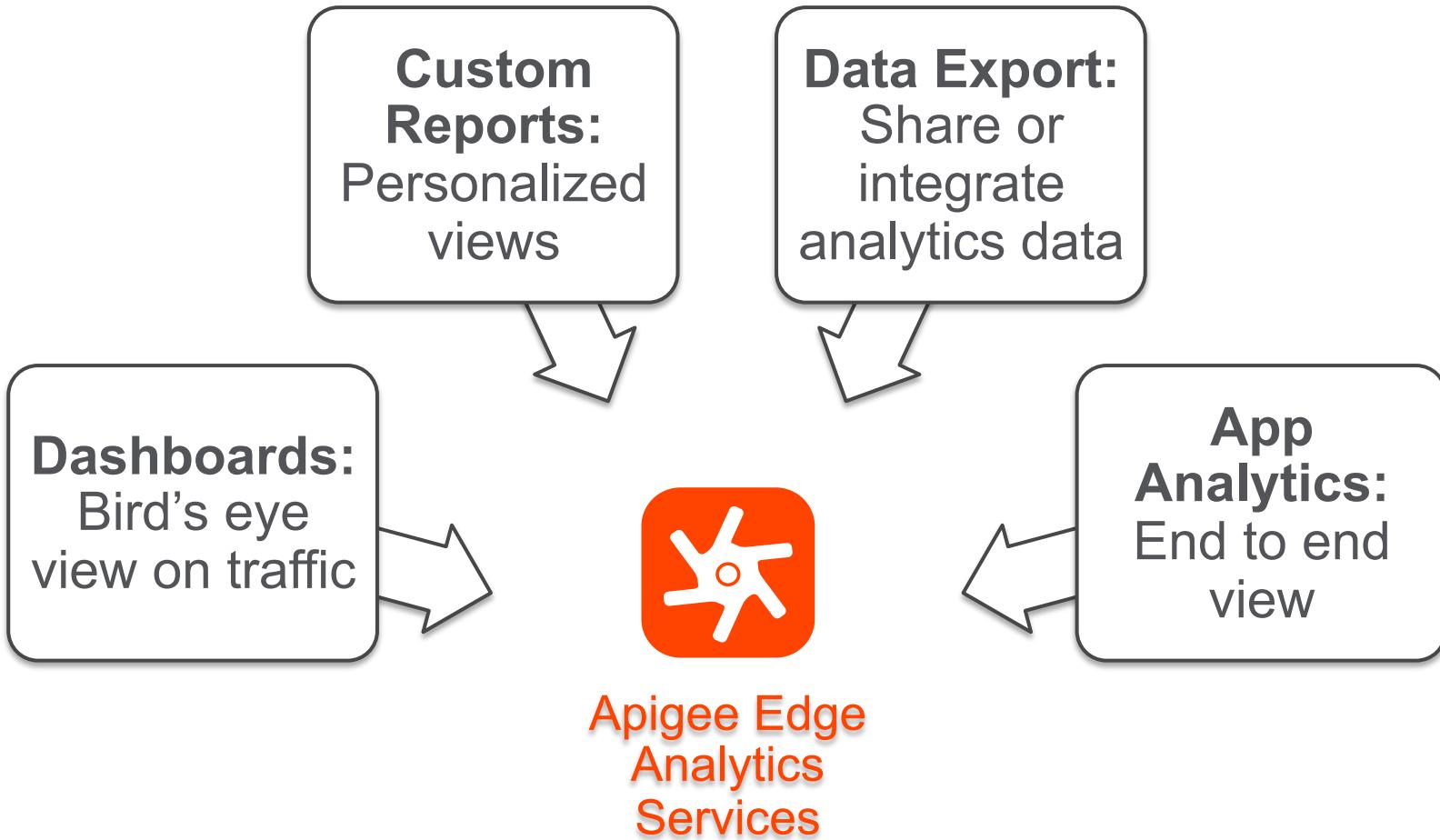
- Data store
  - Collection-oriented access
  - Entity relationships
  - Complex queries
- Users and Devices
  - Authorization and Authentication
  - Multiple devices per user
- Connections/ Social
  - Activities and follower/following relationships
  - Social login
- Location queries
  - Based on latitude/longitude
  - “LOCATION WITHIN” queries
  - Applicable to any entity
- Push Notifications
  - One device, multiple devices, or all devices
  - Send based on query and entity relationships

# Apigee Edge API BaaS Use Case



- No ready API access
- Underpowered or unreliable source
- View over multiple disparate sources needed
- Data “pre-optimization” and “enrichment”
- “System of record” or “reference database”

# Apigee Edge – Analytics Services



# Analytics Services Dashboard

apigee

Dashboard

APIs

Publish

Analytics

Insights

Admin

Help

cvonsee@apigee.com -

API Platform -

Organization

apigee-internal

Environment

prod

Hour Today Week Month Custom

By Day

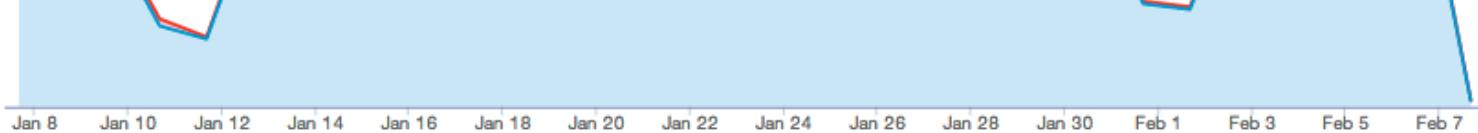
Wed Jan 8 2014, 12:00:00 am – Fri Feb 7 2014, 11:59:59 pm PST

API Program View

API Traffic

[Detail]

All Traffic



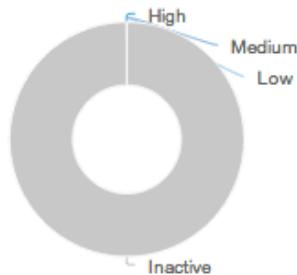
TRAFFIC TOTAL  
**27,121,698**

TODAY VERSUS YESTERDAY

TRAFFIC ERRORS  
**876,199**

Developer Engagement

Traffic Level



DEVELOPER SIGNUPS TOTAL

**3**

THIS WEEK VERSUS LAST WEEK

—

- High: 0
- Medium: 0
- Low: 0
- Inactive: 3

Developer Apps

No traffic in the selected date range

# Custom Reporting

apigee Dashboard APIs Publish Analytics Insights Admin Help cvonsee@apigee.com - API Platform - Organization apigee-internal

Dashboard / Custom Reports / Response time for 4G / Edit

## Response time for 4G

### Basics

Report Name: Response time for 4G  
Report Description: Attempting to measure the performance of our own APIs.  
Chart Type: Column Line  
For Column charts, the x-axis represents groups designated by Drilldowns. For Line charts, the x-axis represents time.

### Data

Aggregation Interval: Daily Hourly Per-Minute  
Environment: prod

### Measures

The y-axis represents measurement values.

Measure	Aggregate Function	Actions
Total Response Time	Sum Average Min Max	+ Measure

### Drilldowns

Drilldowns have two purposes:

- Initially, the drilldown is used to group data, similar to the GROUP BY clause in SQL.
- Once a drilldown is selected, it becomes a filter, similar to the WHERE clause in SQL, and the subsequent drilldown becomes the grouping mechanism.

Drilldown	Actions
+ Drilldown	

### Filter

Basic Advanced

Filter Conditions	Connector	Name	Operator	Value	Actions
					+ Filter Condition

Cancel Save

Drilldown

1 Select...  
Access Token API Product API Proxy API Proxy Revision Client Host Client ID Client IP Address custom1 custom2 custom3 Developer Developer App Developer Email Environment

Actions

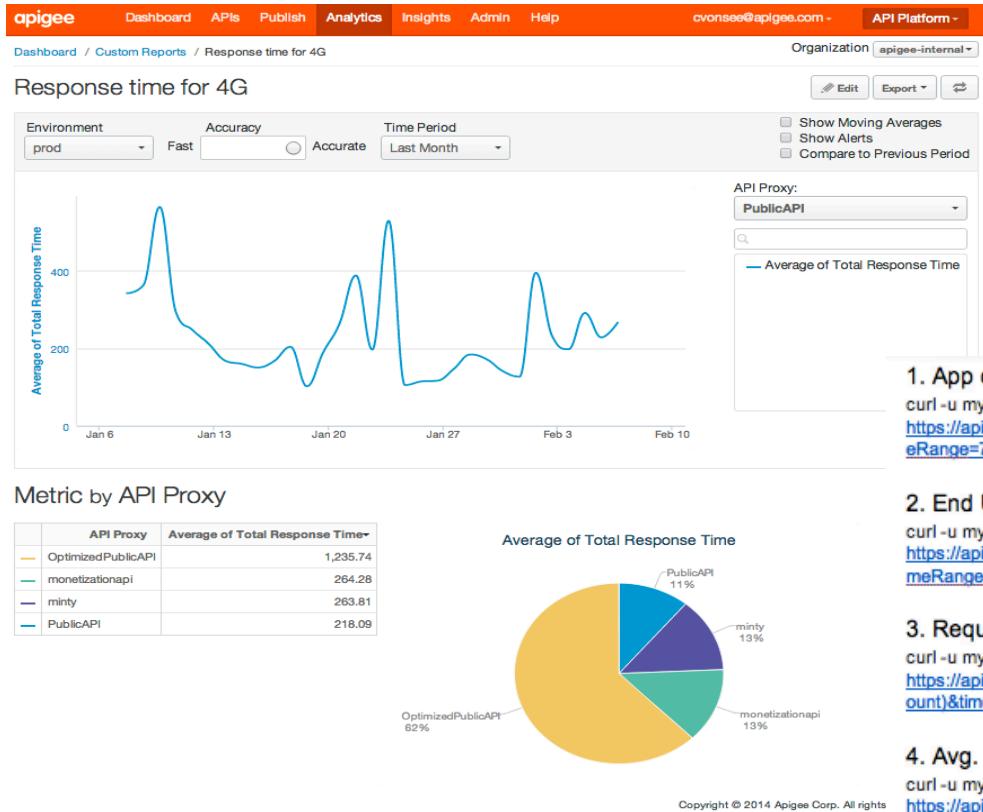
x Delete + Drilldown

Filter

Name Operator

Custom reports can show a number of different measures and provide details based on a variety of different drilldown criteria

# Exporting Statistics



## 1. App count by Developers

curl -u myname:mypass

[https://api.enterprise.apigee.com/v1/organizations/{myorg}/environments/\\*/stats/devs?select=sum\(app\\_count\)&timeRange=7/24/2012%2000:00~8/25/2012%2000:00&timeUnit=day](https://api.enterprise.apigee.com/v1/organizations/{myorg}/environments/*/stats/devs?select=sum(app_count)&timeRange=7/24/2012%2000:00~8/25/2012%2000:00&timeUnit=day)

## 2. End User count by Developers

curl -u myname:mypass

[https://api.enterprise.apigee.com/v1/organizations/{myorg}/environments/\\*/stats/devs?select=sum\(user\\_count\)&timeRange=7/24/2012%2000:00~8/25/2012%2000:00&timeUnit=day](https://api.enterprise.apigee.com/v1/organizations/{myorg}/environments/*/stats/devs?select=sum(user_count)&timeRange=7/24/2012%2000:00~8/25/2012%2000:00&timeUnit=day)

## 3. Requests by Developers

curl -u myname:mypass

[https://api.enterprise.apigee.com/v1/organizations/{myorg}/environments/prod/stats/devs?select=sum\(message\\_count\)&timeRange=7/24/2012%2000:00~8/25/2012%2000:00&timeUnit=day](https://api.enterprise.apigee.com/v1/organizations/{myorg}/environments/prod/stats/devs?select=sum(message_count)&timeRange=7/24/2012%2000:00~8/25/2012%2000:00&timeUnit=day)

## 4. Avg. Total Response Time by Developers

curl -u myname:mypass

[https://api.enterprise.apigee.com/v1/organizations/{myorg}/environments/prod/stats/devs?select=avg\(total\\_response\\_time\)&timeRange=7/24/2012%2000:00~8/25/2012%2000:00&timeUnit=day](https://api.enterprise.apigee.com/v1/organizations/{myorg}/environments/prod/stats/devs?select=avg(total_response_time)&timeRange=7/24/2012%2000:00~8/25/2012%2000:00&timeUnit=day)

## 5. Max Total Response Time by Developers

curl -u myname:mypass

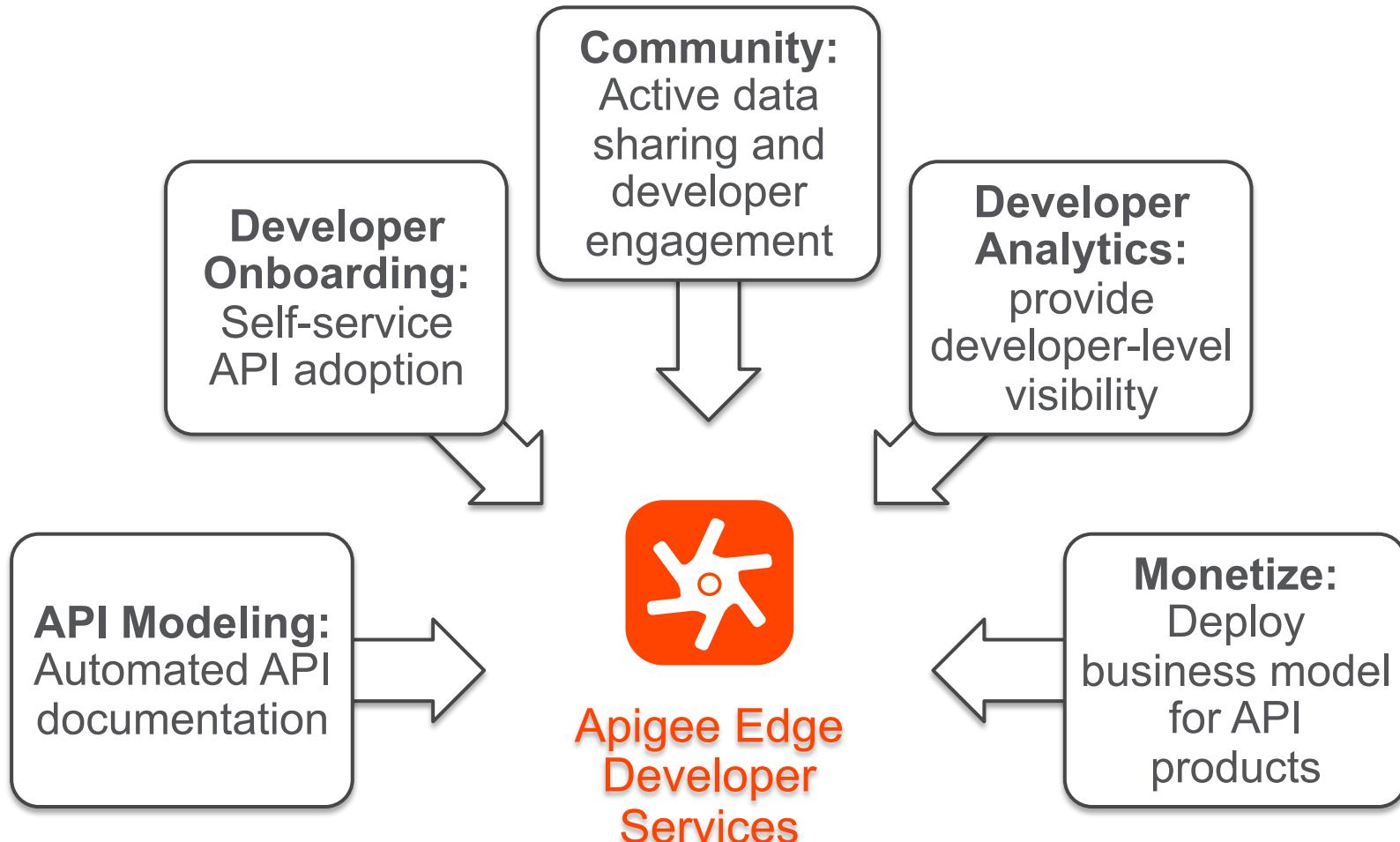
[https://api.enterprise.apigee.com/v1/organizations/{myorg}/environments/prod/stats/devs?select=max\(total\\_response\\_time\)&timeRange=7/24/2012%2000:00~8/25/2012%2000:00&timeUnit=day](https://api.enterprise.apigee.com/v1/organizations/{myorg}/environments/prod/stats/devs?select=max(total_response_time)&timeRange=7/24/2012%2000:00~8/25/2012%2000:00&timeUnit=day)

## 6. Min Total Response Time by Developers

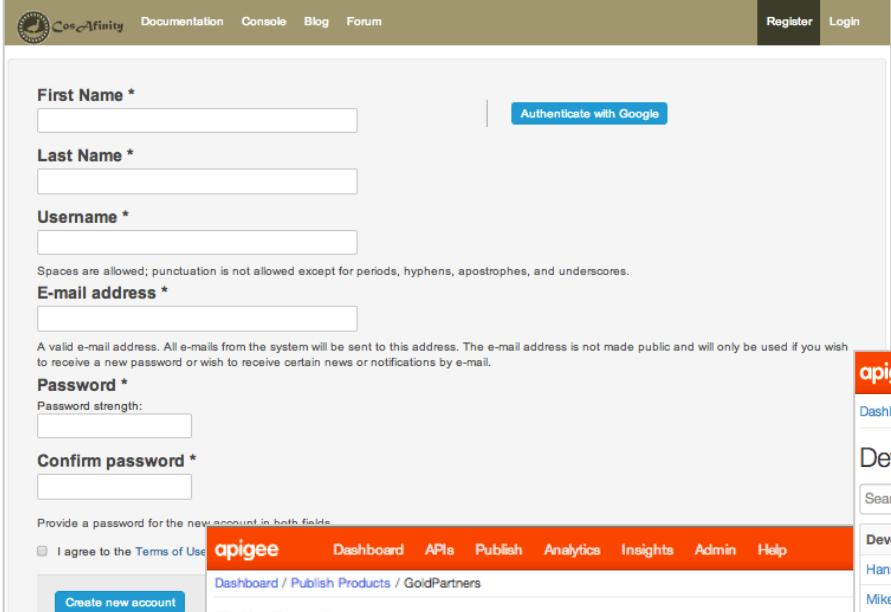
curl -u myname:mypass

[https://api.enterprise.apigee.com/v1/organizations/{myorg}/environments/prod/stats/devs?select=min\(total\\_response\\_time\)&timeRange=7/24/2012%2000:00~8/25/2012%2000:00&timeUnit=day](https://api.enterprise.apigee.com/v1/organizations/{myorg}/environments/prod/stats/devs?select=min(total_response_time)&timeRange=7/24/2012%2000:00~8/25/2012%2000:00&timeUnit=day)

# Apigee Edge – Developer Services



# Onboard your developers via self service



Developer signs up and requests key

Key approval designation

First Name \*

Last Name \*

Username \*

E-mail address \*

Spaces are allowed; punctuation is not allowed except for periods, hyphens, apostrophes, and underscores.

Password \*

Confirm password \*

Provide a password for the new account in both fields.

I agree to the Terms of Use

Create new account

apigee Documentation Console Blog Forum

Dashboard APIs Publish Analytics Insights Admin Help

Dashboard / Publish Products / GoldPartners

Edit Product Products give developers access to your APIs. Learn more

Product Details

Name GoldPartners

Description

Environment prod

Access

- Internal only — Visible only to developers at cosafinity during app registration
- Private — Visible only to external developers with explicit permission during app registration
- Public — Visible only to any registered developer during app registration

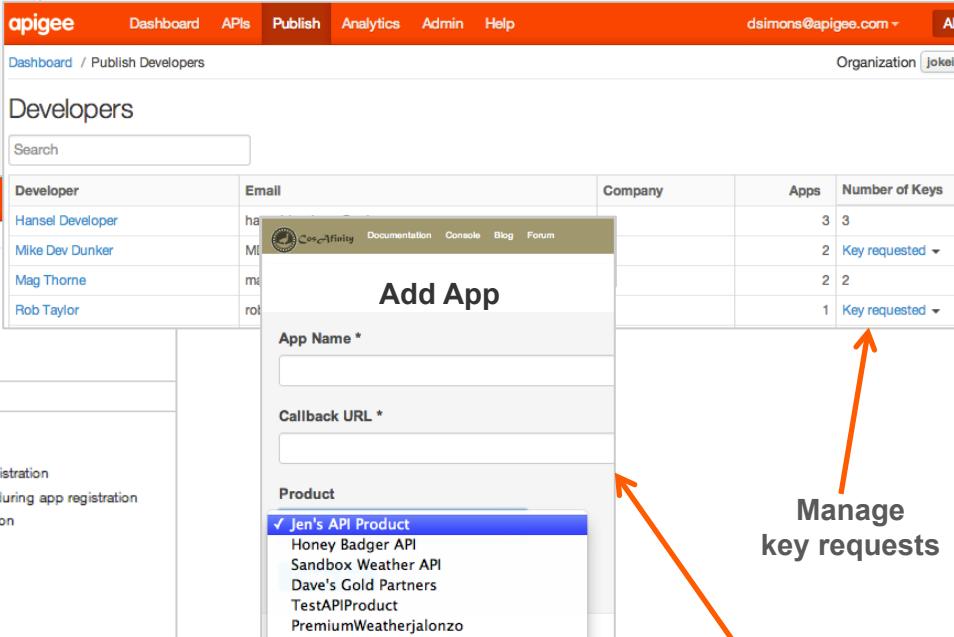
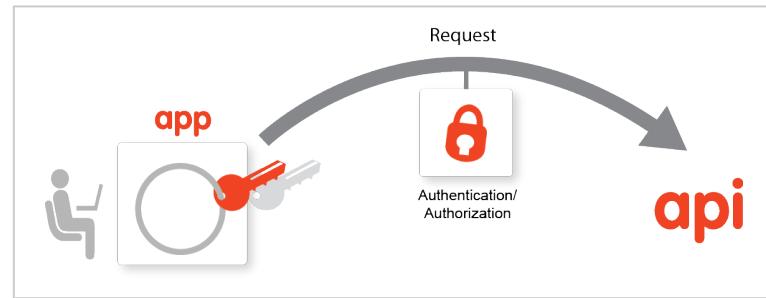
Key Approval Type

- Automatic
- Manual

Quota 1200 Requests per Minute

Scopes

Comma separated scope names. Learn more



apigee Dashboard APIs Publish Analytics Admin Help

dsimons@apigee.com AP

Organization jokelin

Developers

Developer	Email	Company	Apps	Number of Keys
Hansel Developer	hansel@cosafinity.com	cosafinity	3	3
Mike Dev Dunker	mike.dunker@cosafinity.com	cosafinity	2	Key requested ▾
Mag Thorne	mag.thorne@cosafinity.com	cosafinity	2	2
Rob Taylor	rob.taylor@cosafinity.com	cosafinity	1	Key requested ▾

Add App

App Name \*

Callback URL \*

Product

- Jen's API Product
- Honey Badger API
- Sandbox Weather API
- Dave's Gold Partners
- TestAPIProduct
- PremiumWeatherjalonzo
- Airport Status

Developer ready to go

Manage key requests

# Customize the Developer Experience

gracenote. *Developer*

Platforms ▾ Support ▾ Blog

register | log in | search

Search

WELCOME TO THE WHOLE FOODS DEVELOPER CENTER

Providing support for the API Development Community

Whole Foods provides open API to access Whole Foods' product catalog, featuring full product description (with image), pricing and availability. One can also search for 'Discounts' across range of product categories. These API can also be used to locate Acme stores in a town/city.

HOME APIs SUPPORT BLOG

LEARN MORE

EXPLORE Explore Our APIs

Explore the API provided by Whole Foods... from Recipe API, to Sales Flyer API

LEARN Discuss & Learn

Get in touch with the Whole Foods API team and developer community. See if an issue has already been discussed in our forums.

REGISTER Register for an Account

You need to register for an account to use the APIs. Or use OpenID to use your Facebook or Google account.

## Feature Requests

- Log in to post new content in the forum.

Topic	Replies	Last reply
Additional album and track information By pdiefen1 1 week 3 days ago	1	By derek 5 days 1 hour ago
audio recognition By developer 1 week 3 days ago	3	By developer 6 days 14 hours ago
Wrapper for other development platforms By h.dederichs 2 weeks 2 days ago	4	By pdiefen1 1 week 11 hours ago
Information about feature requests By mgubman 2 weeks 2 days ago		

mBlox DEVELOPERS

PRODUCTS PRICING DOCS HELP LOGIN

Engagement Manager

Rest API Hwy

mBlox Control Center

Developer Server

SMS Push Notifications Geo-fencing

iOS and Android

Send push notifications to keep app users engaged.

Use REST API to send alerts, coupons and more to iOS and Android users. Engage users months after install.

Learn more or START NOW FOR FREE

Getting Started

Why mBlox?

Resources

Getting started is quick and easy.

We provide robust, global services.

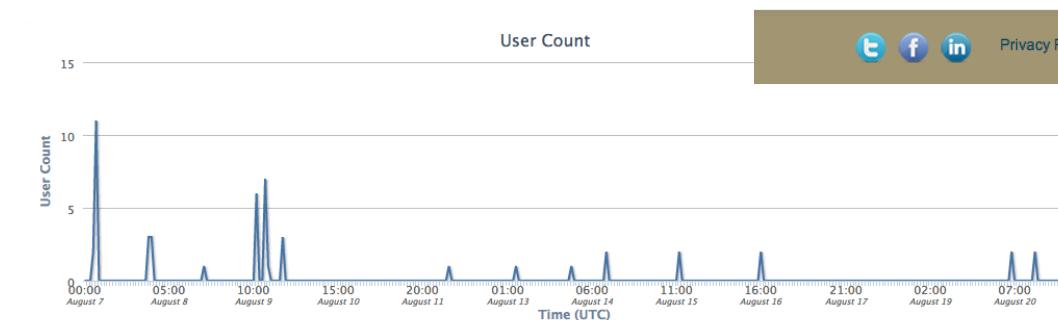
Get answers to the most common

© 2013 Apigee Confidential – All Rights Reserved

# Provide developers with their own analytics

The screenshot shows the CosAffinity interface with a dropdown menu open over a search bar. The menu lists several metrics: Throughput, Max Response Time, Min Response Time, Endpoint Response Time, Message Count (which is checked), Error Count, and User Count.

The screenshot shows the CosAffinity interface with a search form. The 'Display' dropdown is set to 'User Count'. The 'From' field is set to '08/07/2013 00:00 UTC' and the 'To' field is set to '08/21/2013 00:00 UTC'. A 'Submit' button is partially visible.



The screenshot shows the CosAffinity interface with a search form. The 'Display' dropdown is set to 'Message Count'. The 'From' field is set to '08/07/2013 00:00 UTC' and the 'To' field is set to '08/28/2013 00:00 UTC'. A 'Submit' button is visible.



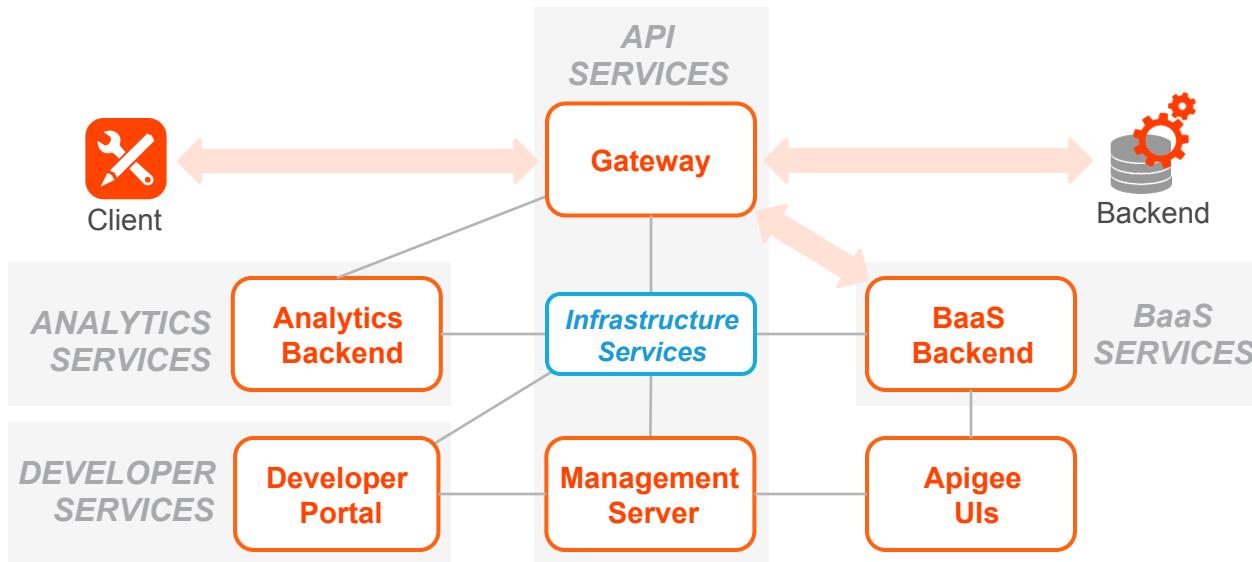
# Architectural Details



# Overview

Apigee Edge comprises of several stateless components that use “Infrastructure Services” to persist data.

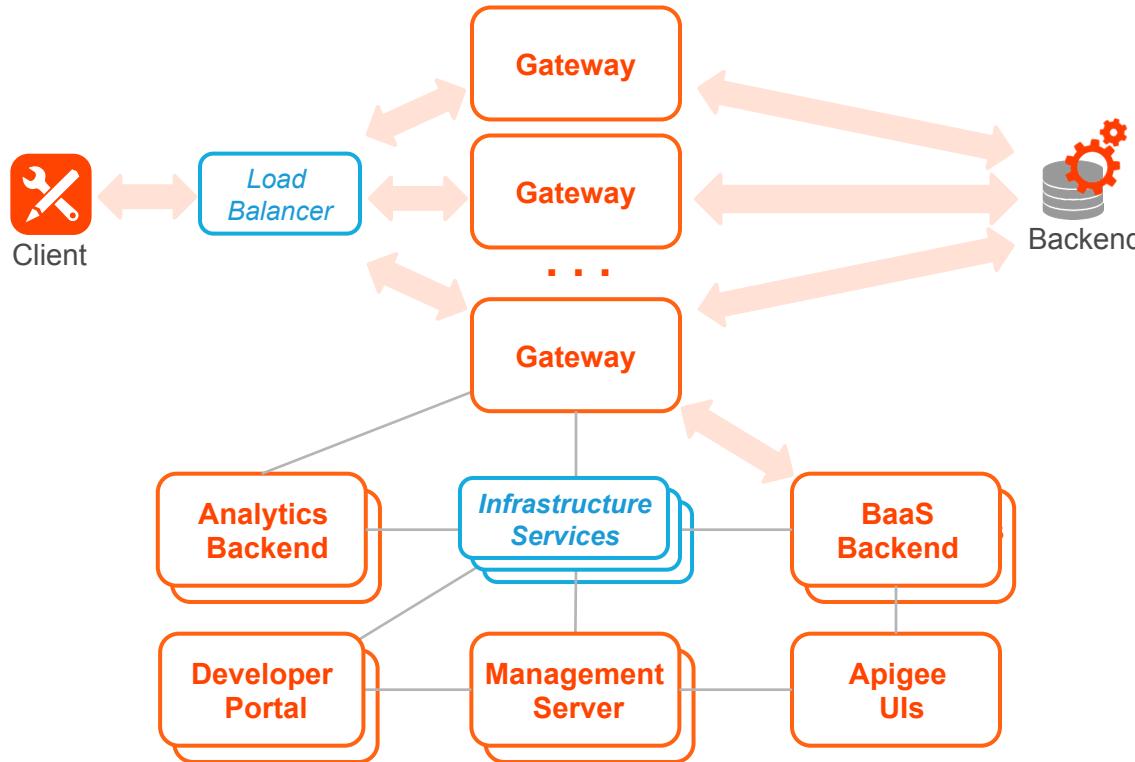
- **Gateway:** Router + Message Processor
- **Apigee UI's:** Enterprise UI, BaaS Portal
- **Infrastructure Services:** Cassandra, Zookeeper, LDAP
- **BaaS Services:** Nginx, Tomcat
- **Analytics Services:** PostgreSQL, Apache Qpid, Ingest
- **Developer Services:** Drupal, mySQL
- **Management Server:** Provider of REST APIs for all configuration tasks



# Architecture (High level)

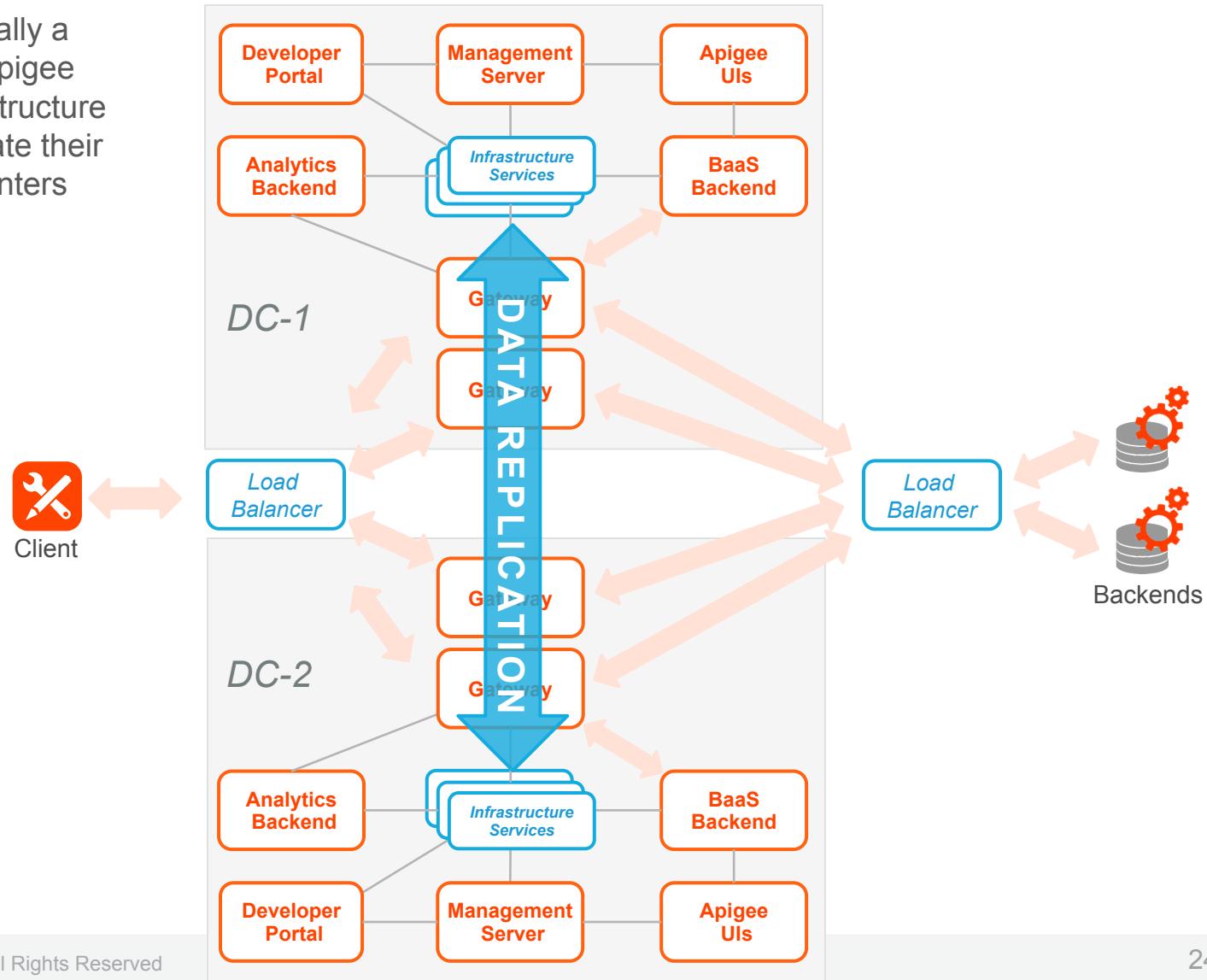
## Additional scaling and redundancy of Backend Services

- Additional gateways can be added easily to follow increased load.
- Management Server, Analytics Backend, App Services Backend and Developer Portal can also be configured in a Highly Available scenario.



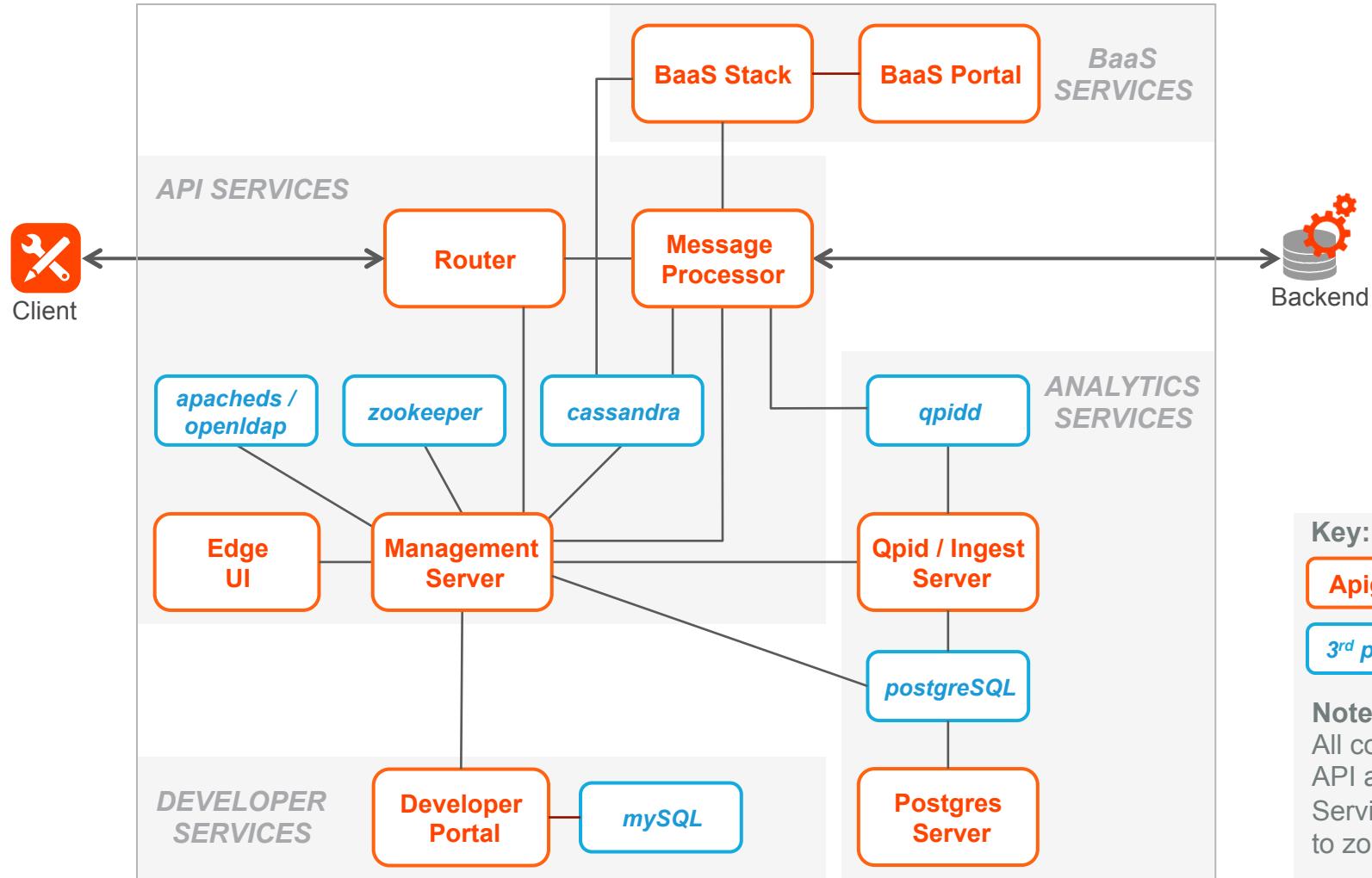
# Architecture (High level)

A DR setup is basically a duplicate setup of Apigee Gateway with Infrastructure Services that replicate their data across data centers



# Architecture (Component View)

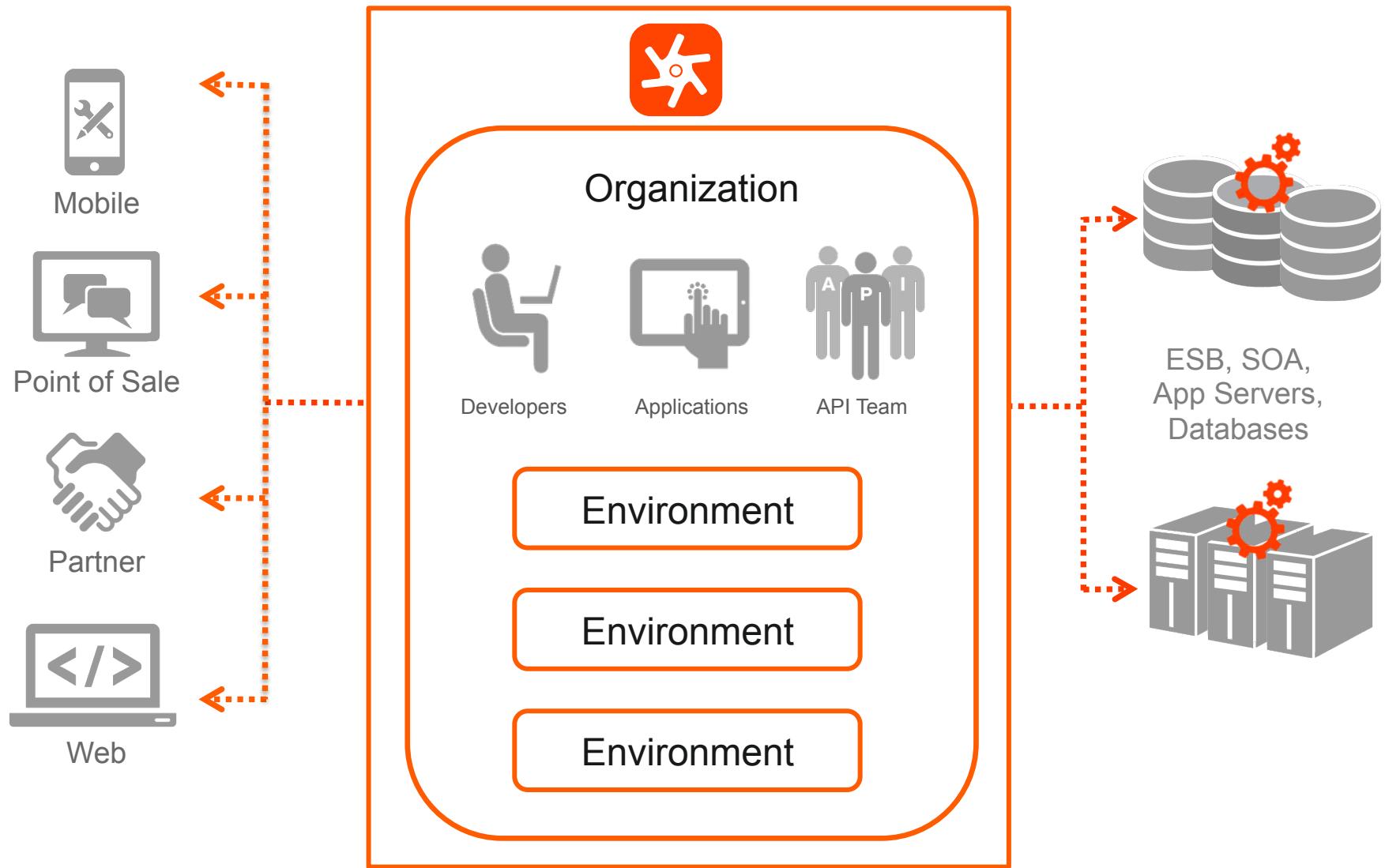
The four high level blocks consist of several java applications together with storage components (cassandra, mySQL, postgreSQL, ldap) and the distributed configuration manager, zookeeper.



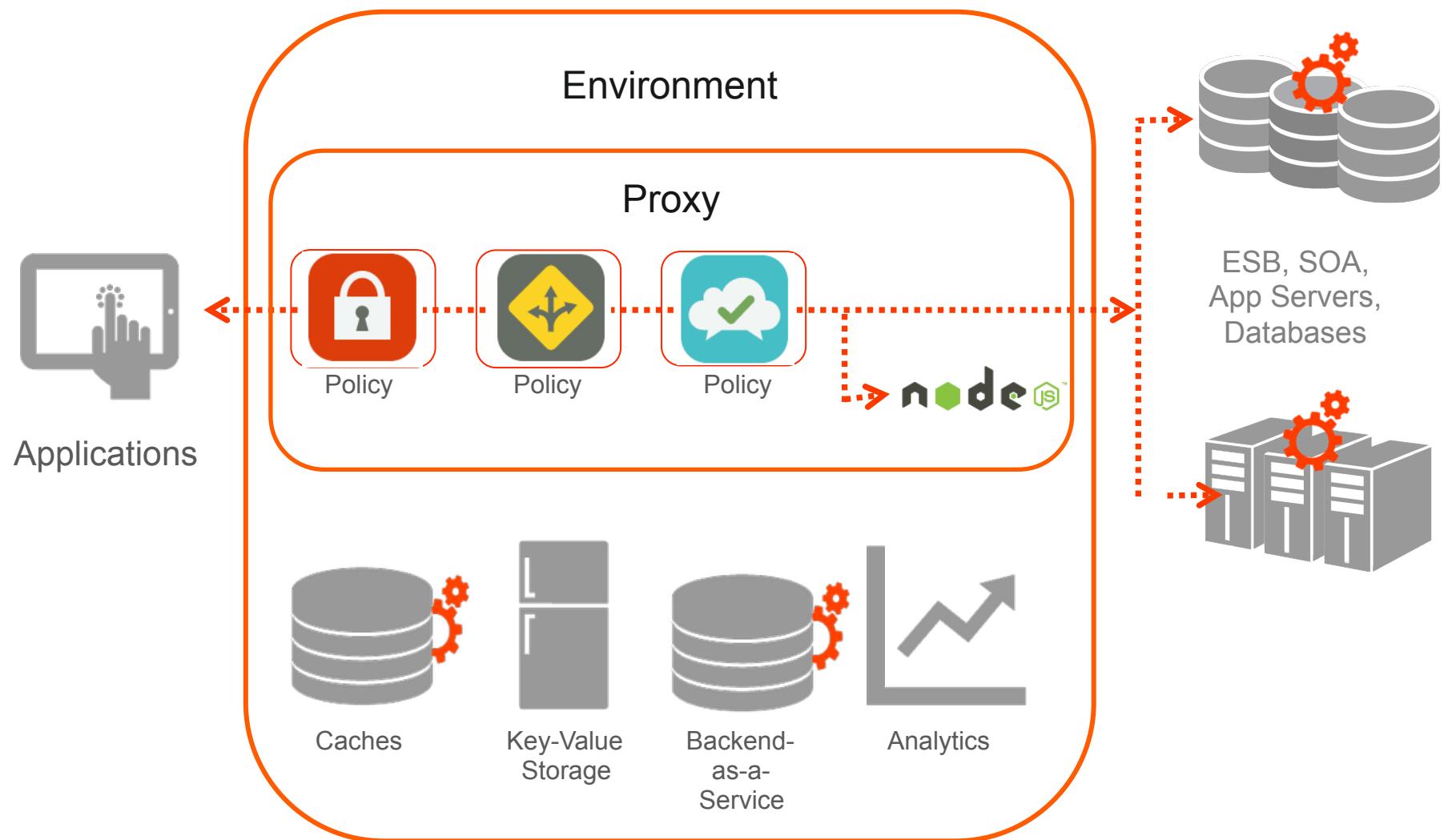
# Concepts



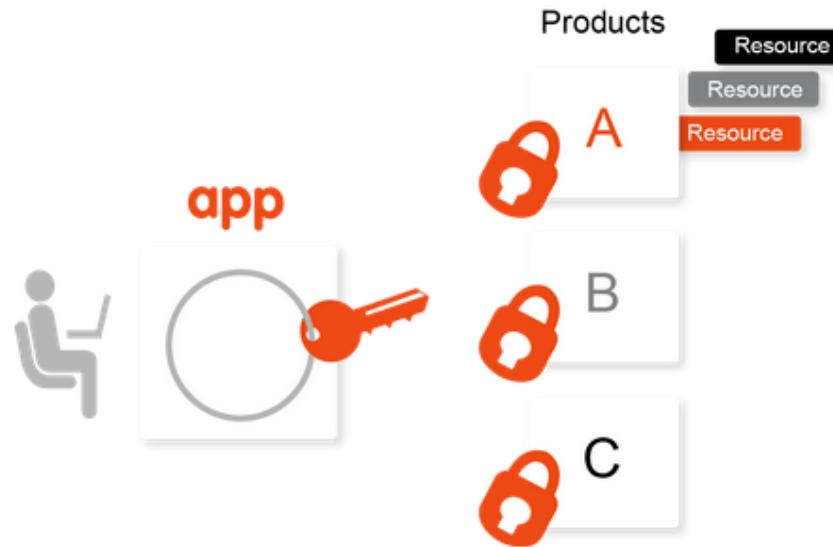
# Apigee concepts in a nutshell



## Apigee concepts in a nutshell (cont'd.)



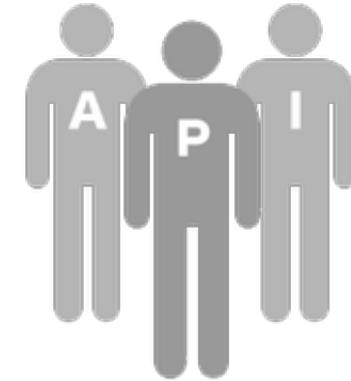
## Apigee concepts in a nutshell (cont'd.)



- **Keys** are:
  - assigned to an application when it is created
  - linked to a Product when the Product is associated with an application
- **API Products** are:
  - collections of API resources, combined with a service plan and presented to developers as a bundle.
  - the central mechanism for authorization and access control.

# Apigee concepts in a nutshell (cont'd.)

- **Permissions** define create/read/update/delete access to resources
- **Roles** identify a collection of permissions that can be assigned to a user
- Predefined roles assign common permissions to key Apigee resources



## Organization Administrator

- Administrator of an organization
- Responsible for mainly user management but has super access to everything

## Business User

- API program manager
- Responsible for success of API Program and developer management, KPIs

## Operations Administrator

- API Operation Manager
- Responsible for production and test deployment, troubleshooting

## User

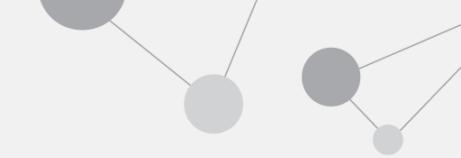
- API Developer
- Responsible for development of API proxy, Policy management, troubleshooting etc.



# What does it mean to “develop with Apigee”?

---

- Organization and Environment Setup
- Determine Users and Access Levels
- Mechanism to Developer Register/Approval, Portal access to Documentation
- Provide guidelines for API Grouping
- Define Deployment, Release and Management Strategy for APIs
- Create API Proxy Design, Testing and Access Control and Source Code Mgmt Guidelines
- Design, implement and document your APIs using policies, Node.js, security, API BaaS, etc.
- Implement tests that verify the functionality, performance and load-bearing capacity
- Deploy your APIs into production, and grant developers access via API keys
- Monitor the adoption and operation of your APIs



## The Basics

---

- An **Organization** is a collection of users, APIs and other resources
- An **Environment** is a subset of APIs in an Organization that are in a given deployment state
  - Default Environments: Test and Prod
- An **API Proxy** is a set of configurable logic that handles API requests.
- **Resources** represent a specific request type within an API proxy - usually qualified by verb and path, but often by other request parameters as well
- **Policies** (also known as Flow Steps) are bits of logic that can be executed during the course of processing a request
  - Policies can be applied to all Resources in a Proxy or only to select Resources
  - Policies can be conditionally executed
- For a given Resource, Policies are grouped together into a **Flow**



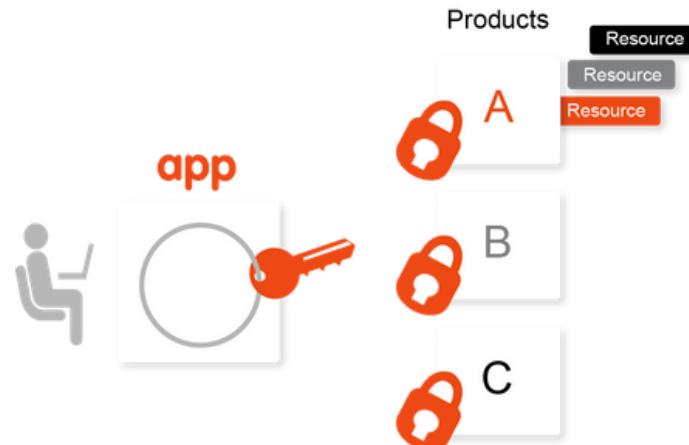
## Developers and Apps

---

- **Developers** are the internal or external partners that create applications that use your API products
  - Developers can be internal or external, and generally represent individuals
  - Developers can be grouped into **Companies**
- Developers are associated with **Applications**, which are developer-written programs that use APIs
  - Applications can also be included in Companies

# API Products and Keys

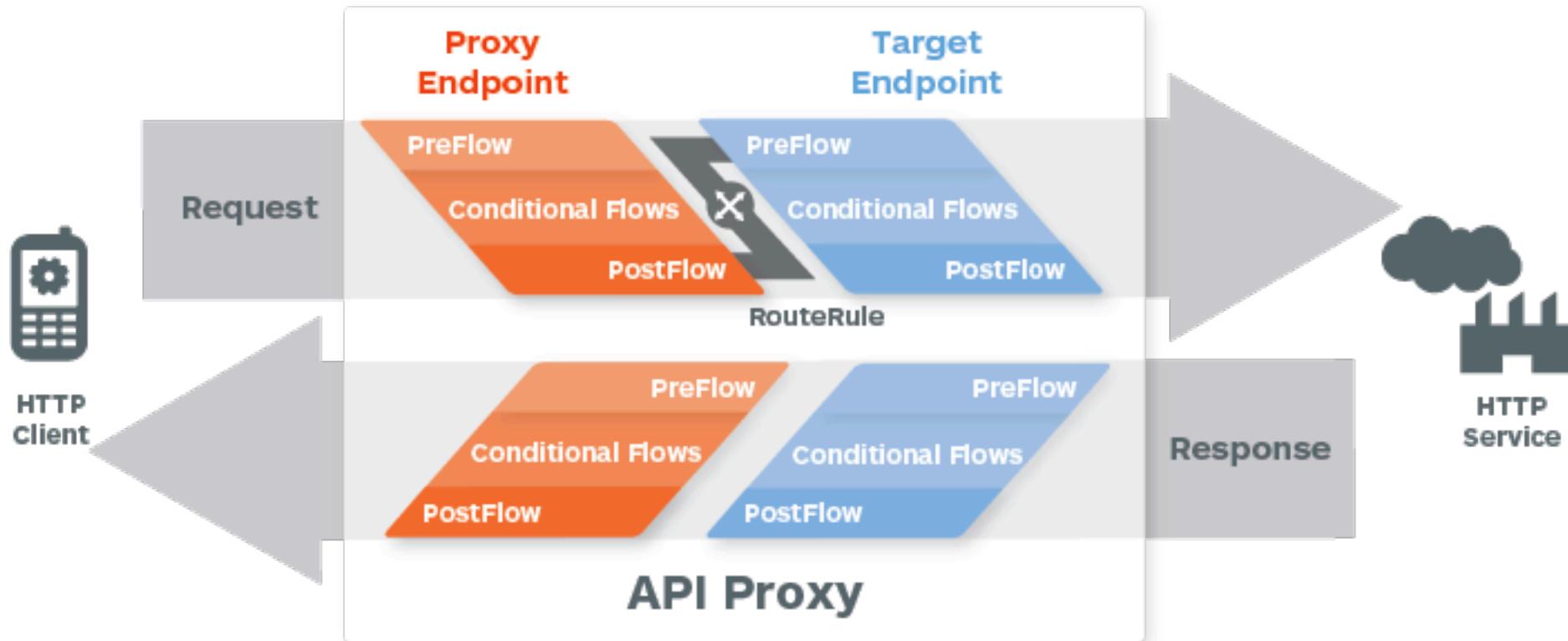
- **API Products** are:
  - collections of API resources – from the same or different APIs - combined with a service plan and presented to developers as a bundle.
  - the central mechanism for authorization and access control. API keys are provisioned for Products, not individual APIs
- **Keys** are:
  - assigned to an application when it is created
  - linked to a Product when the Product is associated with an application



# API Proxy Endpoint and Target Endpoints

- Define an API as a series of resources that access a given target system.
- Client-side interfaces can be accessed using either HTTP or HTTPS
- Targets can be accessed using either HTTP or HTTPS, using either one-way SSL or two-way SSL with mutual authentication
- REST and SOAP targets supported
- “First match” selection: define a set of resource criteria matching incoming requests, and the first match found controls request execution
- Resource matching on path nodes, query parameters, HTTP verbs and other types of conditions
- Determine target service for requests using either static or dynamic routing

# API Proxy Flows



- Flows allow you to define API resources.
- Conditionally execute policies across all flows (resources) or only for specific resources.
- Add policies that execute either before or after backend API request.

# The power of policies

**Manage** interactions with API consumers and optimize performance

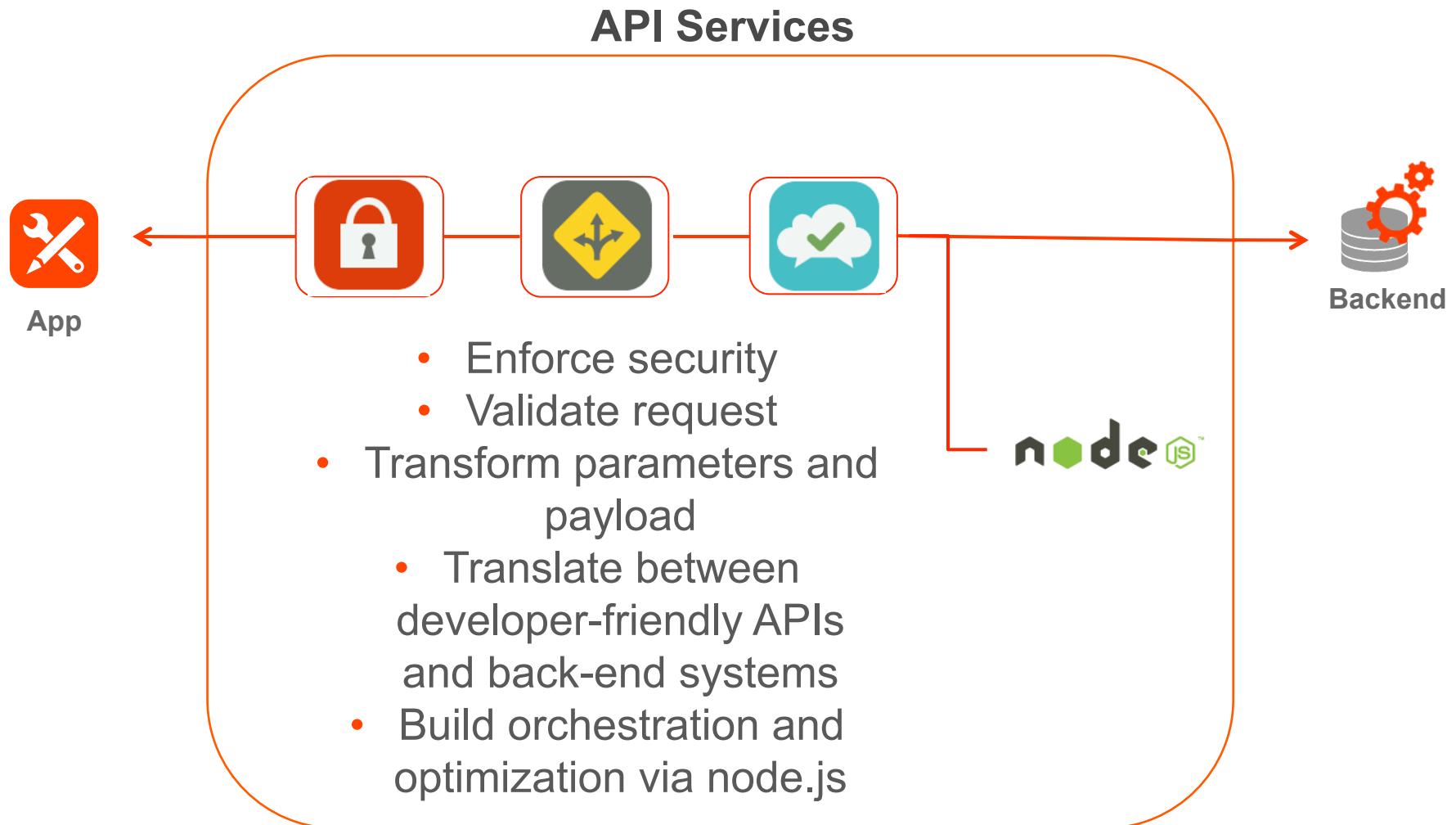
**Transform**, translate and reformat data for easy consumption

Traffic Management	Security
Quota	XML Threat Protection
Spike Arrest	JSON Threat Protection
Response Cache	Regular Expression Protection
Lookup Cache	OAuth v2.0
Populate Cache	Get OAuth v2.0 Info
Invalidate Cache	OAuth v1.0a
Reset Quota	Verify API Key
Mediation	Access Control
JSON to XML	Generate SAML Assertion
XML to JSON	Validate SAML Assertion
Raise Fault	Extension
XSL Transform	Java Callout
SOAP Message Validation	Python
Assign Message	JavaScript
Extract Variables	Service Callout
Access Entity	Statistics Collector
Key Value Map Operations	Message Logging

**Secure** APIs and protect back-end systems from attack

**Extend** with programming when you need it

# The “processing pipeline” is powerful and flexible

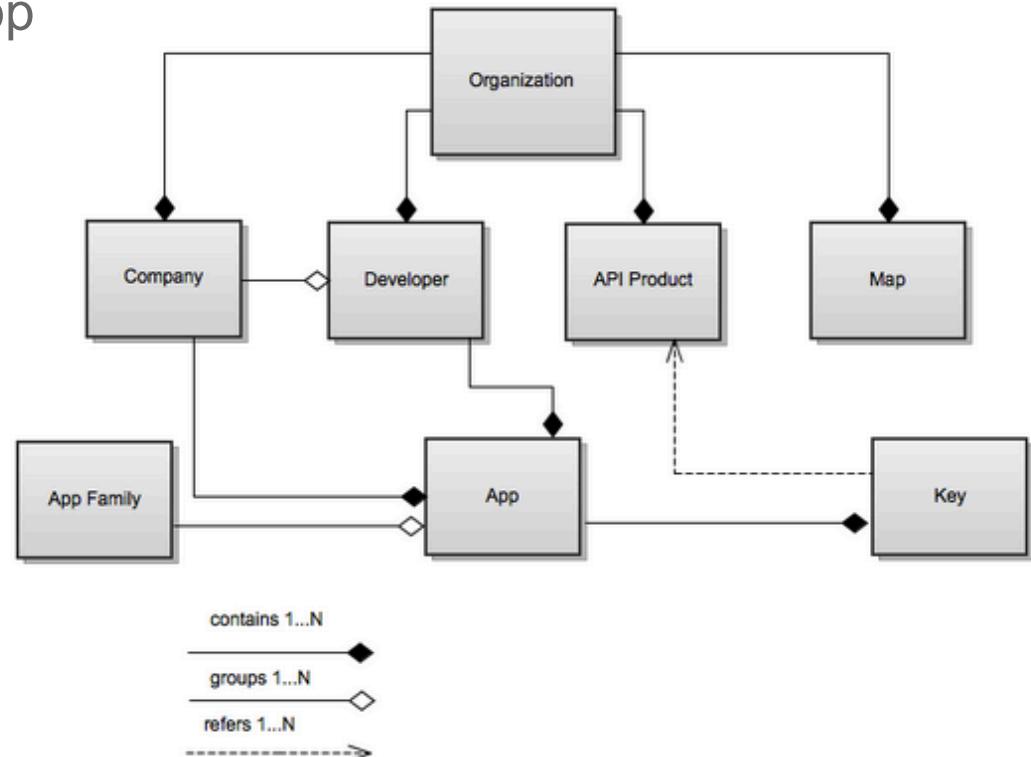


# Variables and Conditions

- **Variables** allow you to store data for use during policy execution
  - Create using the Assign Message policy or from JavaScript/Python/Java
  - Apigee Edge provides an extensive set of pre-defined variables covering areas such as:
    - System (date/time, host name, etc.)
    - Configuration (organization/environment/application name, proxy base path, etc.)
    - Request and response (client IP address, query and form parameters, headers, request body, target host name, timing data, etc.)
    - Policy (variables specific to the individual policy, such as rate limit info)
    - OAuth 1.0a and 2.0 (information related to access tokens, etc.)
- **Conditions** allow you to control when a policy gets executed and which of a number of resource definitions is selected for processing
  - Compare path nodes, HTTP verbs, headers, query parameters, form parameters or variables with each other

# Controlling Apigee Edge using Mgmt APIs

- Create, manage and delete just about anything:
  - API proxies
  - Policies
  - Developers and companies
  - Apps, app families and app keys
  - API products
  - Environments
- Export and import entities
- Manage users within organizations



# Controlling Apigee Edge using Mgmt APIs

- Create, approve, delete, approve and revoke OAuth authorization codes, access tokens and refresh tokens
- Retrieve statistics for environments, APIs, apps, developers, etc.
  - By default, you can retrieve stats on response times, response status codes, request/response sizes, request/response latency, and other dimensions
  - Custom analytics policies allow you to gather and report on data from request paths, query params, headers or payloads
- Start debug sessions and retrieve information

# Thank you

apigee