



Edge

Unit Testing

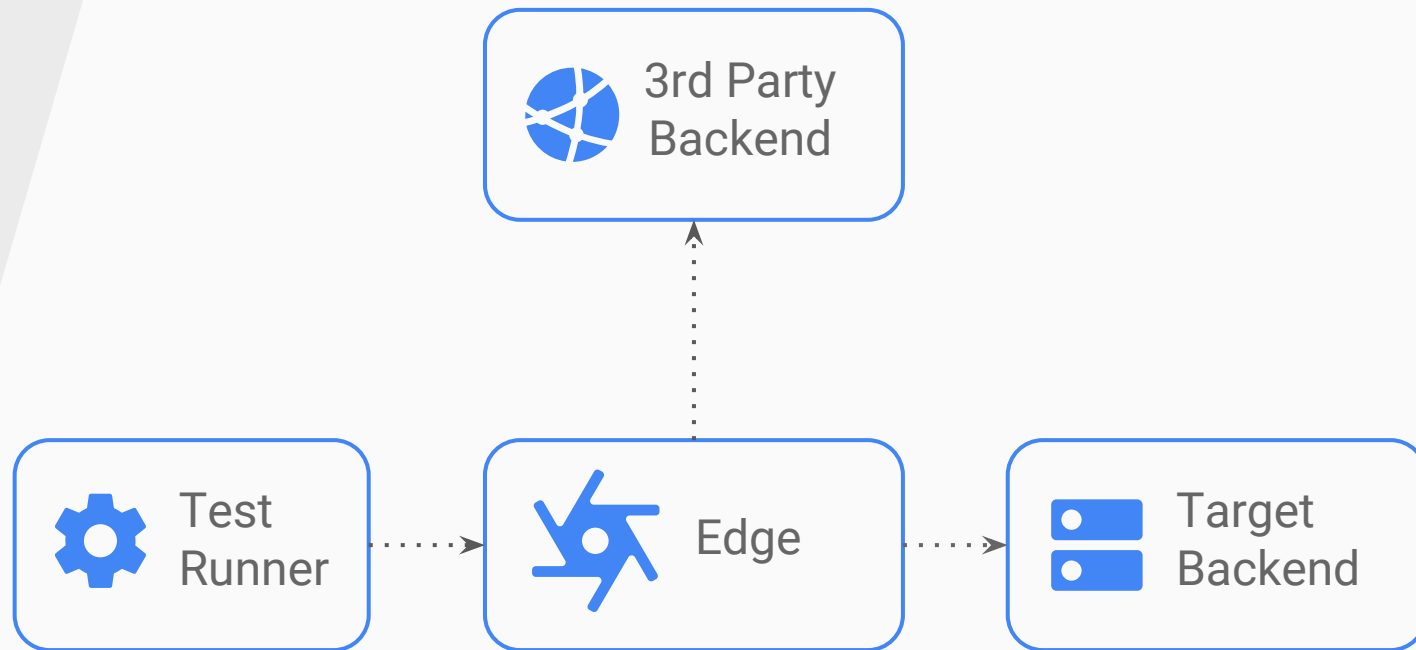
API Proxy Testing



“ Do you think it is possible to test an API proxy just with integration testing? ”

Is Integration Testing Enough?

Service Callouts?
Script Callouts?
Async Operations?



Main Use Cases for Unit Testing



- Operations that our integration testing cannot intercept and therefore cannot assert.
 - Service callouts to external APIs (e.g., Loggly).
 - Testing IP blacklisting / whitelisting - need to simulate different IP addresses coming in to Apigee with many proxies in the middle.
- Very important code (e.g. security code, encryption code, signature generators/validators)
- Where coverage is extremely important, e.g. security code (again)

Unit Testing Benefits

- Code can be tested locally without deployment to Edge first.
- Can create hooks to enforce testing during commit.
- Much faster than integration testing.



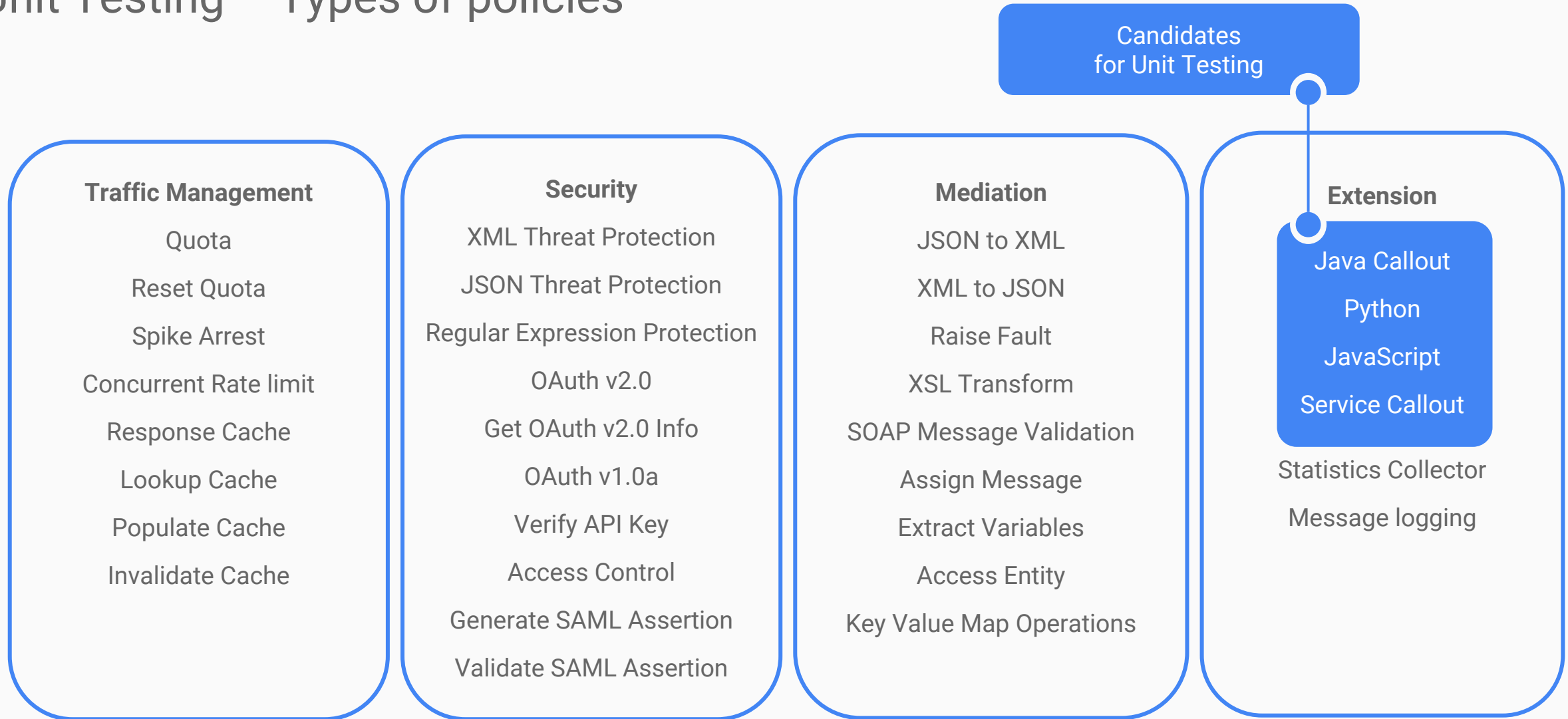
Unit Testing – Boundary Principle



“ Test within your boundaries, don’t test libraries you don’t control ”

The Edge Team

Unit Testing – Types of policies



Tools



Sinon.JS

JUnit

PyUnit