MULESOFT ARCHITECTURE:

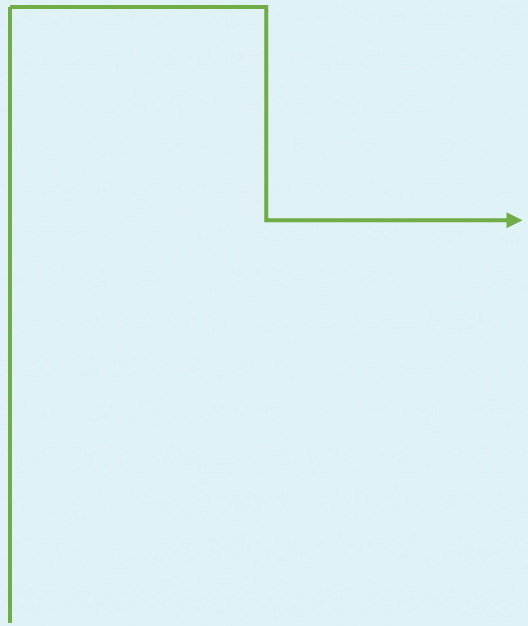
API Lifecycle and SDLC

PRACTIC LEARNING

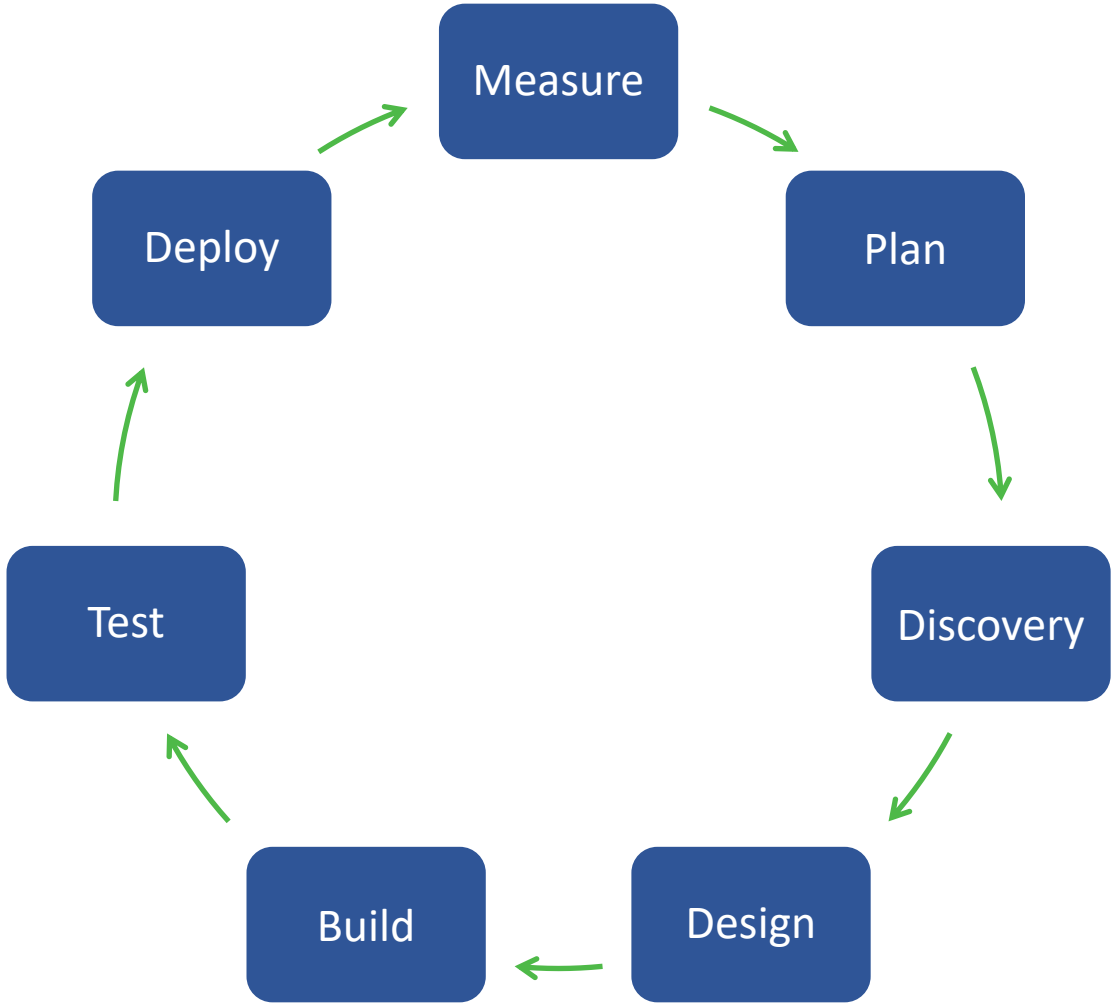
API Design Introduction

1. Service development lifecycle (SDLC)
2. MuleSoft API lifecycle
3. MuleSoft components related to API lifecycle phases

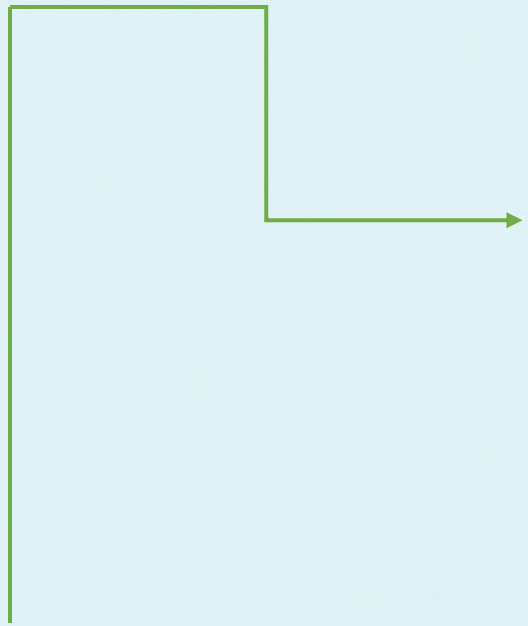
SDLC



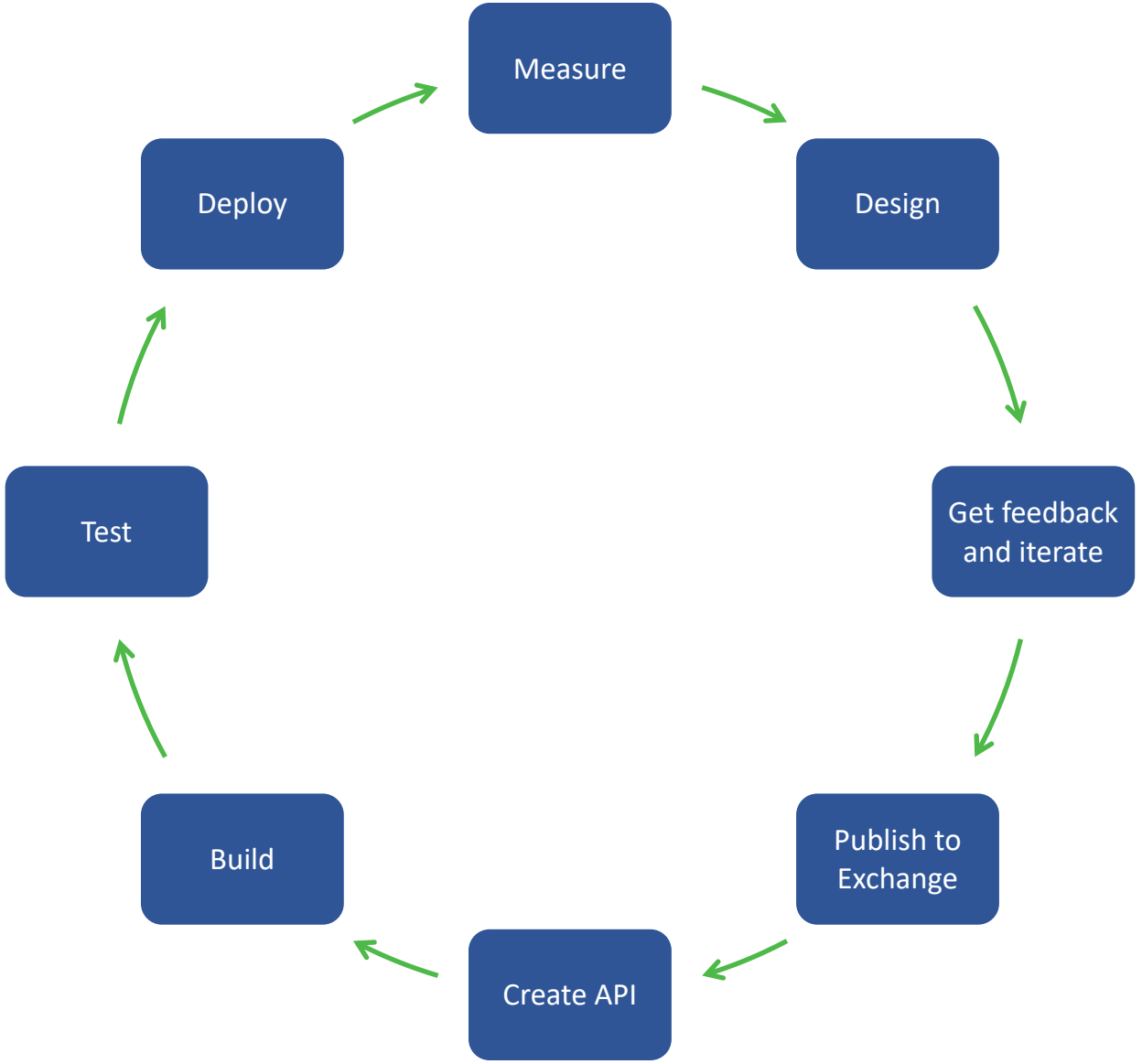
SDLC Phases



API Lifecycle



MuleSoft API Lifecycle



API Lifecycle: Measure

- Critical to establish a baseline of the current state
- Can use 2 great pre-built tools to measure if you already have Anypoint Platform
  - MuleSoft Metrics Accelerator
    - MuleSoft accelerator built as a part of Catalyst containing well documented, common metrics of the Anypoint Platform
  - Big Compass KPIs for APIs Dashboard
    - Great dashboard with many KPIs built by Big Compass (MuleSoft partner), many focusing on ROI and executive-level KPIs such as reuse of APIs
- If setting up Anypoint Platform for the first time, use metrics and KPIs from the current/previous solution to establish a baseline
- For greenfield deployments, set reasonable target metrics to strive for, and establish your baseline after deployment to Production

API Lifecycle: Design

- Occurs in Design Center if building an API
  - Use Mocking Service to demo the API to stakeholders and consumers
    - Iterate based on feedback
- In addition to the API's RAML definition, produce documents such as flow diagrams, sequence diagrams, network architecture diagrams, etc
- Critical phase that can take as long as the build phase
  - This is the primary role of the architect
    - Align stakeholders, development team, and other architects in preparation to develop

API Lifecycle: Publish to Exchange

- API published to Anypoint Exchange
  - Consumers can request access and view API documentation and definition
- Publish the API from Design Center to Exchange to allow for discovery, centralized documentation, and consumption of the asset
- Begin communication to the organization

API Lifecycle: Create API

- Occurs in API Manager
- From the Exchange asset, create the API in API Manager
  - Allows for implementation of API security policies, API governance, SLA enforcement, and API consumer management
  - Never deploy an unprotected API – creating the API in API Manager allows developers to connect to the API from the beginning of the build phase

Basic API Endpoint vs. API Proxy

- Recommended to use basic endpoint in almost all use cases so the API can be customized, including logging
- Only use an API proxy when
  - Dealing with the System API or Experience API layer
  - No data transformation or special authentication/authorization is needed
  - HTTP(S) can be used to connect to the data source (other protocols such as JDBC will not work)
  - Secrets Manager can be used to accept HTTPS requests on the API
  - Logging is not essential (custom, standardized logging is almost always recommended)

API Lifecycle: Build

- Occurs in Anypoint Studio and source control of choice
- Typically where developers take over with support from the architect
- Scaffold API endpoints from the asset in Exchange to jumpstart the implementation
- Use API Autodiscovery to connect the API to API Manager
- Develop to the architectural designs and API definition using established development best practices
- Unit test in MUnit
- Check in code to source control

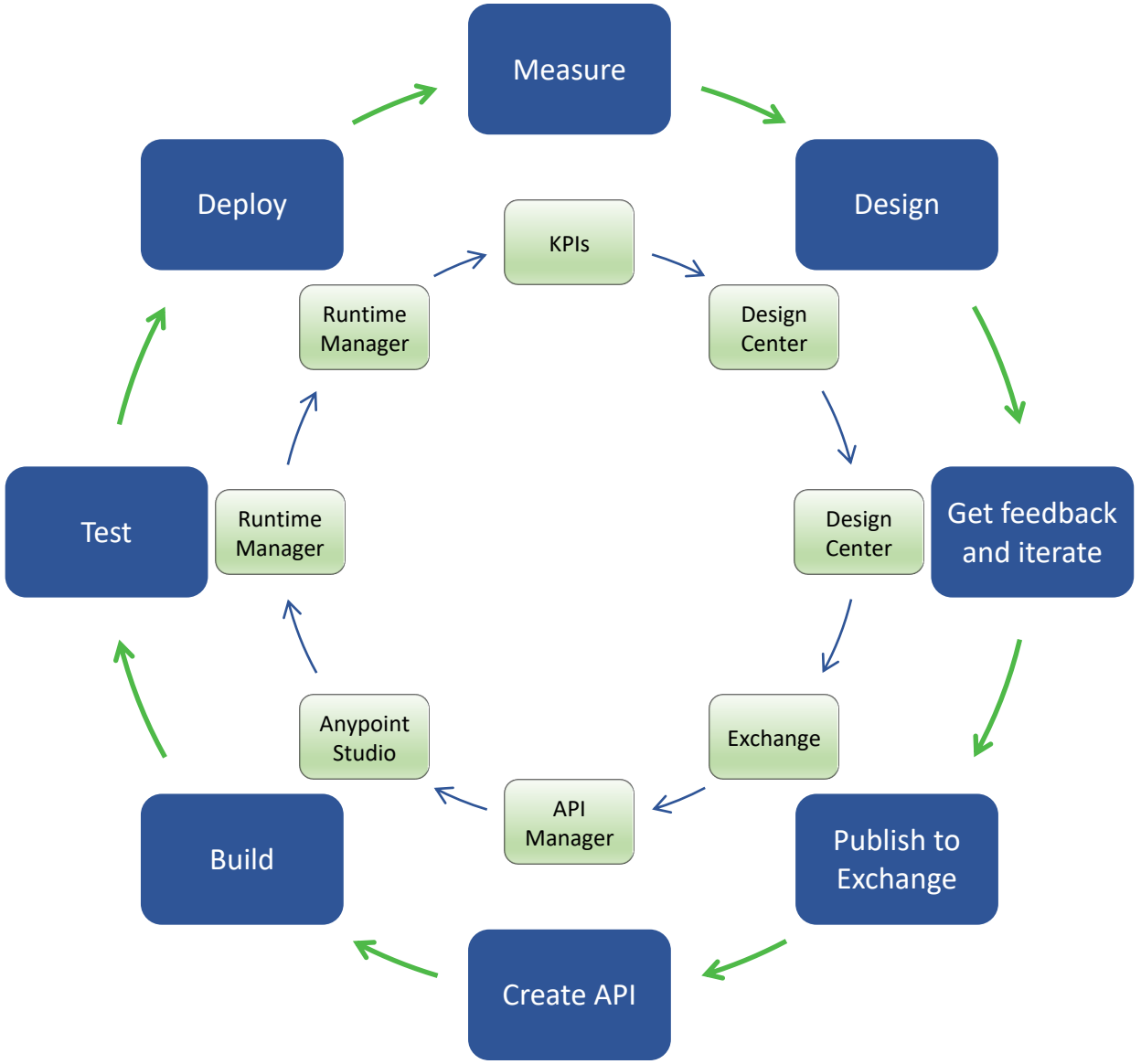
API Lifecycle: Test

- Occurs in Runtime Manager or your server depending on if you deploy to the cloud or not
  - Typically occurs in the QA/Test environment
    - Best practice to deploy using CI/CD
- QA testing team may take over
  - Test cases need to be built, often times by an analyst with support from the architect and developers
  - Execute on test cases to prepare for Production deployment

API Lifecycle: Deploy

- Occurs in Runtime Manager or your server depending on if you deploy to the cloud or not
  - Deploy to the Production environment
    - Best practice to deploy using CI/CD
- Architect, developers, and/or support/operations support the Production deployment at the time of deployment and on an ongoing basis
- Once the solution is deployed for a certain amount of time to make an impact, re-establish your metrics and KPI baseline

MuleSoft API Lifecycle in Practice



API Lifecycle and SDLC Summary

- SDLC
- API Lifecycle
- Anypoint Platform components relating to each phase of the API lifecycle
- Architect’s role in each phase of the SDLC and API lifecycle

Additional Reading

- https://mulesy.com/mulesoft-api-lifecycle/
- https://docs.mulesoft.com/runtime-manager/deployment-strategies
- https://docs.mulesoft.com/runtime-manager/deploying-to-your-own-servers
- https://docs.mulesoft.com/runtime-fabric/1.8/deploy-resource-allocation
- https://github.com/mulesoft-catalyst/metrics-accelerator
- https://meetups.mulesoft.com/events/details/mulesoft-chicago-presents-kpis-of-apis-with-new-mulesoft-metrics-accelerator-mulesoft-chicago-virtual-meetup/
- https://www.bigcompass.com/insights/kpidashboard
- https://wp.nyu.edu/developers/2018/03/29/mule-development-best-practices/
- https://www.mulesoft.com/platform/api/multi-cloud-integration-runtime-fabric