Integration Architecture Specification

«Customer name and domain context»

# 

# Table of contents

[**Table of contents**](#_e0g4e1sw8vxm) **1**

[**Introduction**](#_zh5e22gtyfh) **5**

[Key business drivers](#_rtc0l1hmmde5) 6

[References](#_2uhzg83f7zti) 6

[Acronyms](#_1ksv4uv) 7

[**Solution Architecture**](#_5dlt3iqqcyjx) **9**

[Integration Platform Capabilities](#_8cfcxl2mx775) 10

[Patterns](#_uwf783flenbe) 10

[Connectivity](#_dx3pybvzrhk) 10

[Technical Capabilities](#_n9r2wkb6h1qx) 11

[**Integration Catalog**](#_85rj2c1guulq) **13**

[Fulfilment Orders](#_2jxsxqh) 15

[**MuleSoft Integration Implementation**](#_tx22ft2f9v8v) **17**

[MuleSoft Anypoint Platform](#_3as4poj) 18

[What is an API?](#_1pxezwc) 19

[MuleSoft Connectors](#_49x2ik5) 20

[API Application Catalogue](#_2p2csry) 20

[Reusable APIs](#_147n2zr) 22

[**MuleSoft Platform Reference Architecture**](#_g5e419dtlvqh) **25**

[Platform Capabilities](#_23ckvvd) 26

[Runtime Manager](#_ihv636) 26

[API Manager](#_32hioqz) 27

[Design Center](#_1hmsyys) 28

[Exchange](#_41mghml) 28

[Mule Runtime](#_2grqrue) 28

[Connectors](#_vx1227) 29

[Platform Services](#_3fwokq0) 29

[Anypoint Security](#_1v1yuxt) 29

[Virtual Private Cloud (VPC)](#_4f1mdlm) 30

[Cloud Messaging (Anypoint MQ)](#_2u6wntf) 30

[Object Store](#_19c6y18) 30

[Dedicated Load Balancers (DLB)](#_3tbugp1) 31

[Extensibility](#_28h4qwu) 31

[Platform Setup](#_nmf14n) 31

[Deployment Regions](#_37m2jsg) 31

[Business Groups](#_1mrcu09) 32

[Organization](#_46r0co2) 32

[Organization Owner](#_2lwamvv) 32

[Business Groups Explained](#_111kx3o) 32

[Business Group Structure](#_3l18frh) 33

[Environments](#_206ipza) 33

[Platform Security](#_4k668n3) 34

[Access Management](#_2zbgiuw) 34

[Identity Management](#_1egqt2p) 34

[Platform Roles and Permissions](#_3ygebqi) 35

[Platform Role Types](#_2dlolyb) 35

[Default Platform Roles & Permissions](#_sqyw64) 36

[Proposed Deployment Architecture](#_3cqmetx) 37

[Environments](#_1rvwp1q) 37

[Dedicated Load Balancers](#_4bvk7pj) 37

[Virtual Private Clouds](#_2r0uhxc) 38

[Virtual Private Networks](#_1664s55) 38

[About MuleSoft, a Salesforce company](#_vfa116sc8tk4) 39

***Instructions***

*Delete all italicized instructional text that is highlighted with a cyan background, including this block. Update all placeholder text blocks that are surrounded by «chevrons» and highlighted in yellow, removing the chevrons and yellow highlight color when done. Replace all example content that is highlighted in green.*

# 

# Introduction

This document describes the proposed integration architecture to support the «domain context» for «customer». The integration solution is split into the following aspects:

1. The proposed to-be logical integration processes and associated mechanisms for getting data into and out of «key applications» in a way that is agnostic of the integration layer; it is assumed that the integration platform supports all the required integration capabilities
2. The mechanism for functionally implementing these integration processes using MuleSoft as the integration platform
3. A proposed CloudHub-based platform reference architecture for deploying MuleSoft as the integration platform

Note that any changes to ways of working as well as other organizational and operating model changes that are required to optimally deliver and maintain an integration platform whilst maximizing business agility are outside the scope of this document.

## Key business drivers

The key business drivers for this initiative that are relevant for the integration platform are as follows:

1. Reduce order processing time from 2.5 hours to 2.5 minutes
2. Improve stock level accuracy

## References

Note that any version numbers are latest at the time of writing.

*Edit this table of references to fit the integration architecture.*

|  |  |
| --- | --- |
| **Salesforce Platform REST API** | <https://developer.salesforce.com/docs/atlas.en-us.api_rest.meta/api_rest/intro_what_is_rest_api.htm>  Authentication:  <https://developer.salesforce.com/docs/atlas.en-us.api_rest.meta/api_rest/quickstart_oauth.htm> |
| **Salesforce Composite APIs** | <https://developer.salesforce.com/docs/atlas.en-us.api_rest.meta/api_rest/resources_composite_composite.htm> |
| **Salesforce Platform Events** | <https://developer.salesforce.com/docs/atlas.en-us.platform_events.meta/platform_events/platform_events_intro.htm> |
| **API-led Connectivity** | <https://www.mulesoft.com/lp/whitepaper/api/api-led-connectivity> |
| **MuleSoft CloudHub** | <https://www.mulesoft.com/platform/saas/cloudhub-ipaas-cloud-based-integration> |
| **Mulesoft Anypoint Platform** | <https://www.mulesoft.com/platform/enterprise-integration> |
| **Mule Runtime** | <https://docs.mulesoft.com/mule-runtime/4.3/> |
| **MuleSoft Salesforce Connector** | <https://docs.mulesoft.com/salesforce-connector/10.9/> |
| **MuleSoft Salesforce Composite Connector** | <https://docs.mulesoft.com/salesforce-composite-connector/2.8/> |

## Acronyms

*Edit to reflect the acronyms relevant to your context.*

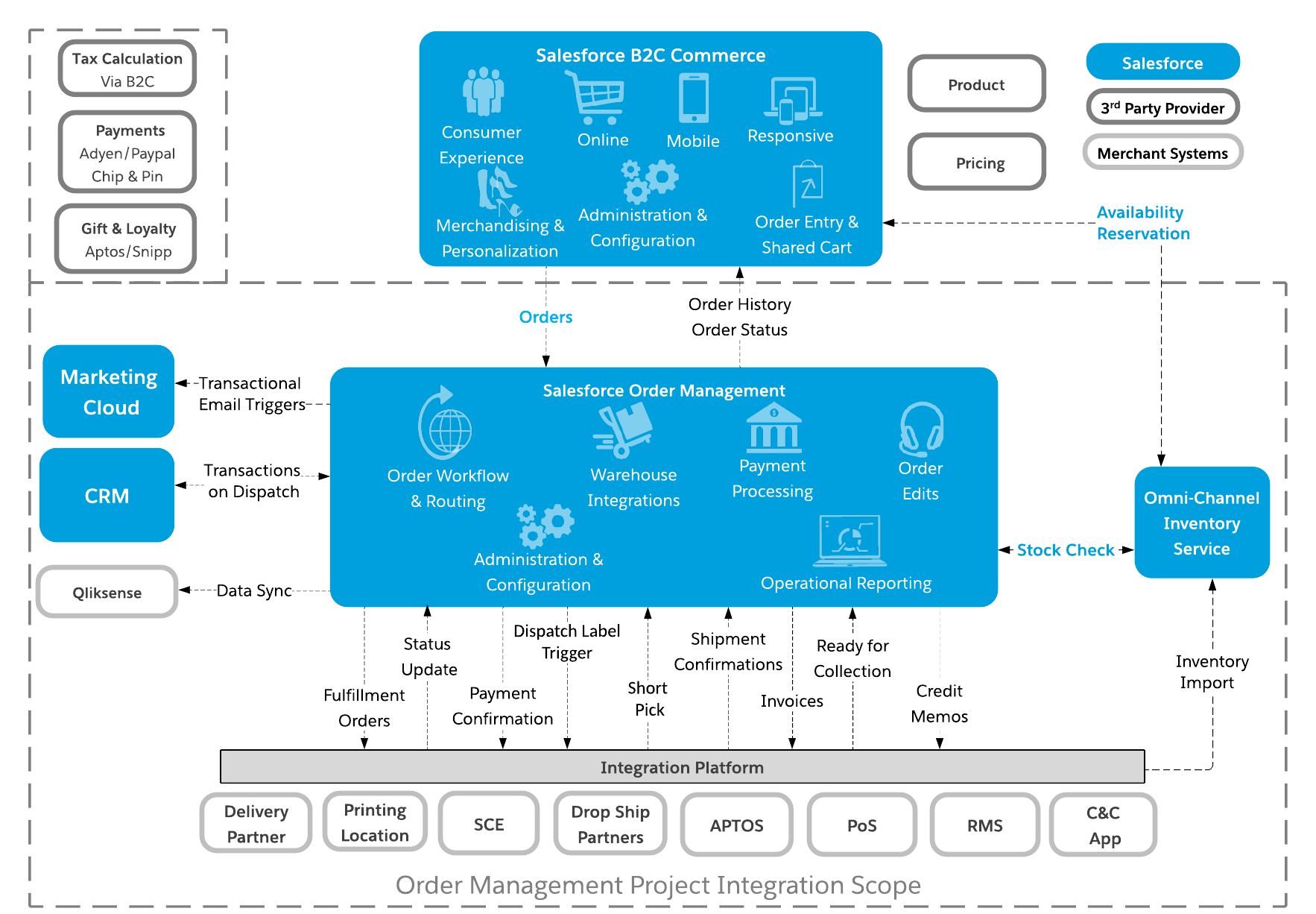
|  |  |
| --- | --- |
| **Acronym** | **Application** |
| RMS | Oracle / Retek Retail Management System (v10) |
| RIB | Oracle Retail Integration Bus (SeeBeyond) |
| SCE | Infor Supply Chain Execution |
| SOM | Salesforce Order Management |
| OMS | Order Management System (aka Salesforce Order Management) |
| OCIS | Salesforce Omni-Channel Inventory Service |
| C&C App | Click and Collect Application |
| ASN | Advanced Shipment Notification - message format |
| PoS | Point of Sale |

# 

# Solution Architecture

The following diagram illustrates the interactions between «main application» and other systems.

*Replace the following diagram with equivalent for your context.*



Note that this document focuses exclusively on integrations that go via the “Integration Platform” and does not describe any other direct system-to-system integrations, e.g., those between the various Salesforce clouds. These are typically provided as “out-of-the-box” integrations.

## Integration Platform Capabilities

The integration platform in this document is assumed to provide and be responsible for the following capabilities:

*Adjust according to your connectivity requirements.*

1. Connectivity
2. API Design and Management
3. Discovery and Reuse
4. Batch Processing
5. Monitoring
6. Enterprise Security

### Patterns

This document assumes that the Integration Platform supports the following integration patterns:

*Adjust according to your connectivity requirements.*

1. Synchronous Request-Reply
2. Fire and Forget
3. Message with Asynchronous Callback
4. Messaging Publish-Subscribe
5. Process Orchestration

### Connectivity

This document assumes that the Integration Platform supports the following connectivity technologies:

*Adjust according to your connectivity requirements.*

1. Salesforce
   1. Platform Events
   2. Streaming APIs
   3. sObject CRUD operations
   4. Custom Apex REST
   5. Bulk API
   6. Composite API
2. Database (poll for changes plus CRUD)
3. SFTP (poll, read, write, move)

### Technical Capabilities

In addition to this, the integration platform is expected to be able to support the following technical capabilities:

*Adjust according to your connectivity requirements.*

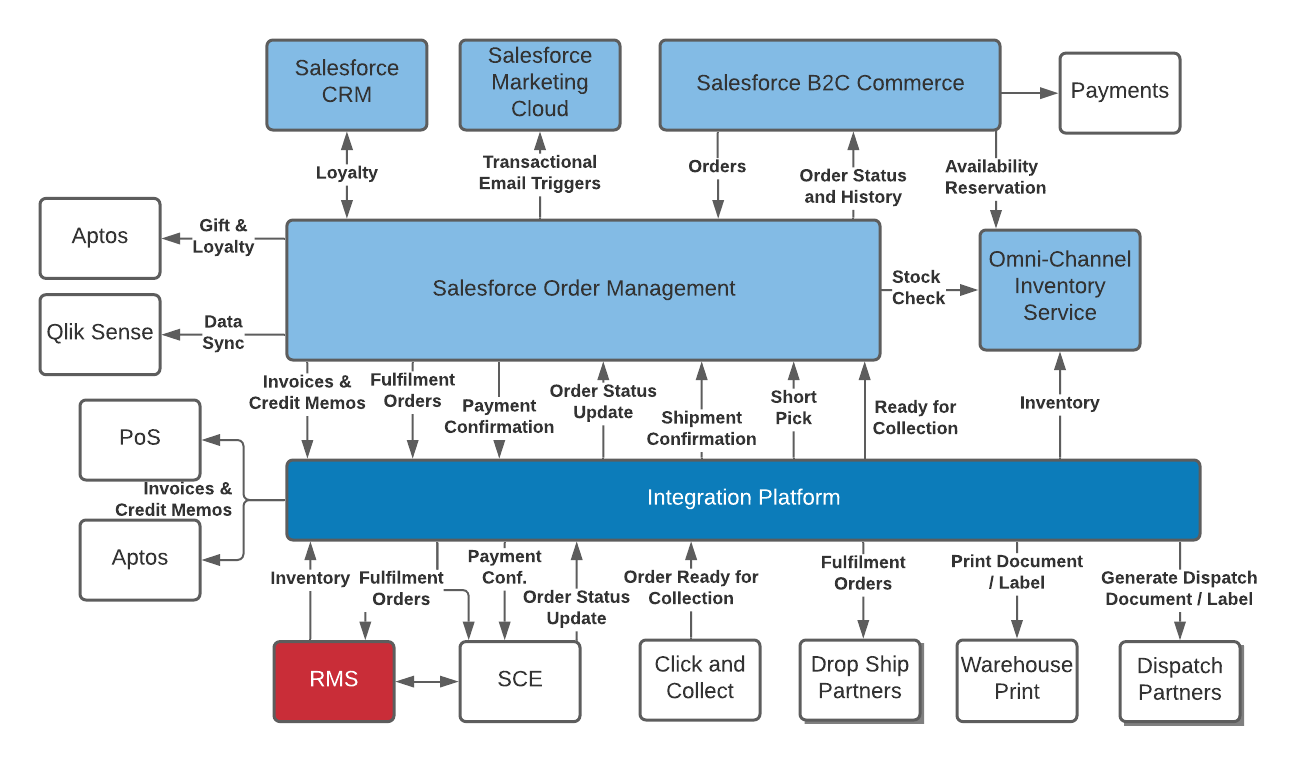
* Data mapping and transformation
* Support for a wide range of standard protocols, including SOAP, REST
* Unit testing, including the ability to simulate external dependencies
* CI/CD integration for automated testing and deployment
* Support for full API lifecycle, from conception to deployment
* Exchange API contracts with consumers to support API development
* Automatic generation of mock APIs
* Routing and orchestration
* Event logging and monitoring
* Analytics

# 

# Integration Catalog

The diagram below provides a high-level overview of the integration flows for this system.

*Replace the diagram below with an integration context diagram to match your domain context.*



The Integration Catalog is driven by the inbound trigger; a single integration flow will result in outbound interactions with one or more target systems. Note that multiple source systems can trigger the same integration flow.

*Catalog all the integration flows - replace all of those in the table below to reflect your integration flow diagram.*

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Source System(s)** | **Target System(s)** | **Comments** |
| Fulfilment Orders | OMS | SCE and RMS; or  Drop Ship Partner | Time critical, ideally less than 2.5 mins from Order submission. The Integration Platform will detect if the Order is for Drop Ship Partners and route the message accordingly. |
| Warehouse Order Status Updates | SCE | OMS | Also includes Shipment Confirmation and Short Pick messages. |
| Invoices | OMS | PoS,  Aptos |  |
| Credit Memos | OMS | PoS,  Aptos |  |
| Order Ready for Collection Notification | C&C App | OMS |  |
| Inventory Position | RMS & Non-RMS Concessions | OCIS | Periodically poll for batches of inventory position updates; assumes polls are performed every 5 mins. |
| Payment Confirmations | OMS | SCE |  |
| Generate Dispatch Label/Documents | OMS | Dispatch Partner | Note that it is currently assumed that the generation and subsequent request to print dispatch documents will be treated as separate integrations that are orchestrated by OMS. |
| Print Dispatch Label/Documents | OMS | Warehouse Dispatch Print |  |

*Replace the example below with details for every integration flow from the table above in the following format.*

## Fulfilment Orders

**Description**: Replicate fulfilment order details that are made in Salesforce B2C Commerce into either RMS and SCE (for non DropShip orders) or the relevant Drop Ship Partner (for Drop Ship orders).

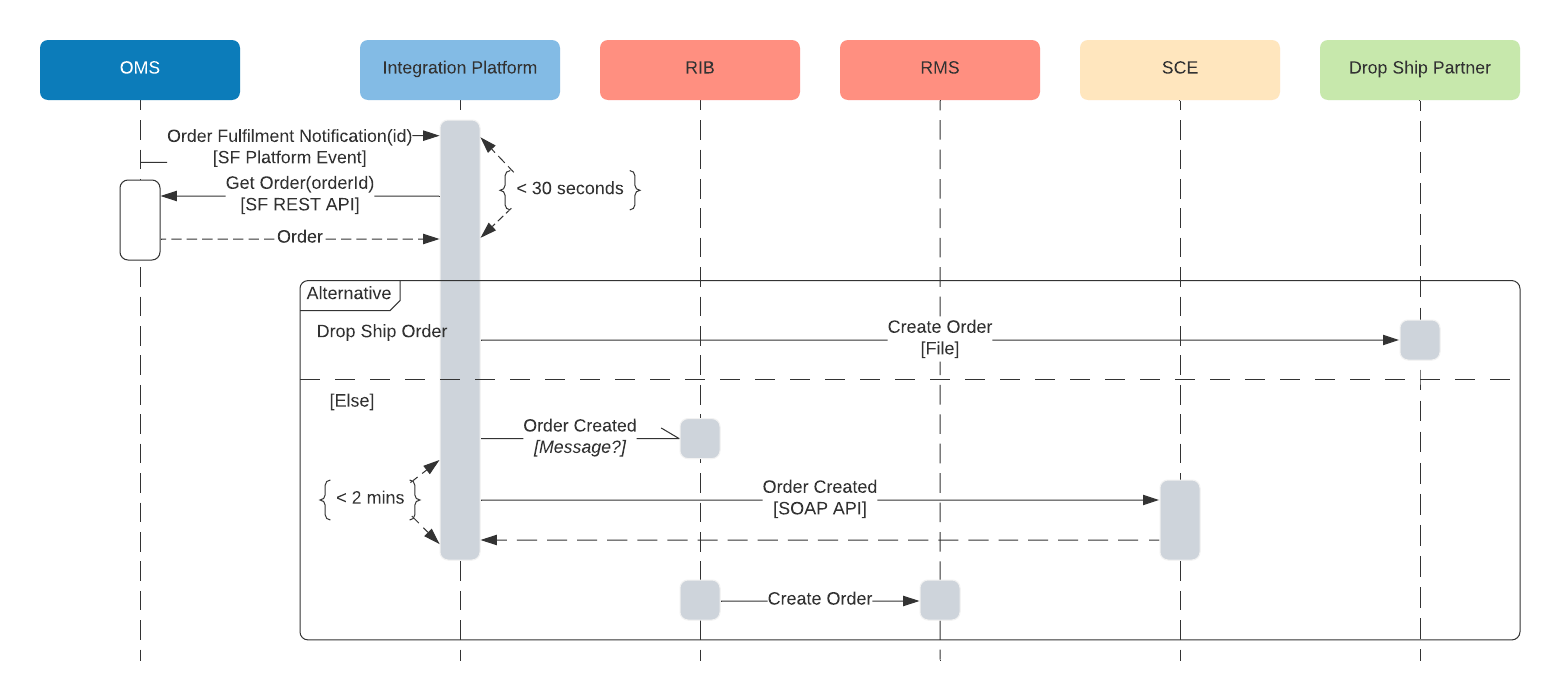
**Timeliness**: Near real-time, ideally less than 2.5 mins delay from order placement to equivalent entities shared with target systems.

**Customer Reference: XX-ABC-123, YY-CBA-345**

**Business Trigger**: An order is created within Salesforce B2C Commerce which is sent to OMS.

**Source System(s) [Protocol]**: OMS [Platform Event]

**Flow**:



Where the Integration Platform is provided by MuleSoft, it is anticipated that on average it will take less than 30 seconds from OMS sending an Order Fulfilment Platform Event to MuleSoft getting the relevant Order object from OMS via the Salesforce Retrieve Object REST API. It is then anticipated that it will take less than 2 minutes in total for MuleSoft to send an Order Created message to both the RIB as well as SCE. It is assumed that there is a message based API that can be used for sending the Order to the RIB. It is currently unknown how long it would take for the RIB to commit that Order Fulfilment message into RMS.

Order Volumes:

* Approximately 2,000 per brand per day.
* Peak was just over 5,000 orders per day around the last Black Friday.

**Outbound Integrations [Protocol]**:

1. Drop Ship Orders
   1. Drop Ship Partner [File]
2. Non Drop Ship Orders
   1. RIB [Message - to be confirmed]
   2. SCE [SOAP API]

# MuleSoft Integration Implementation

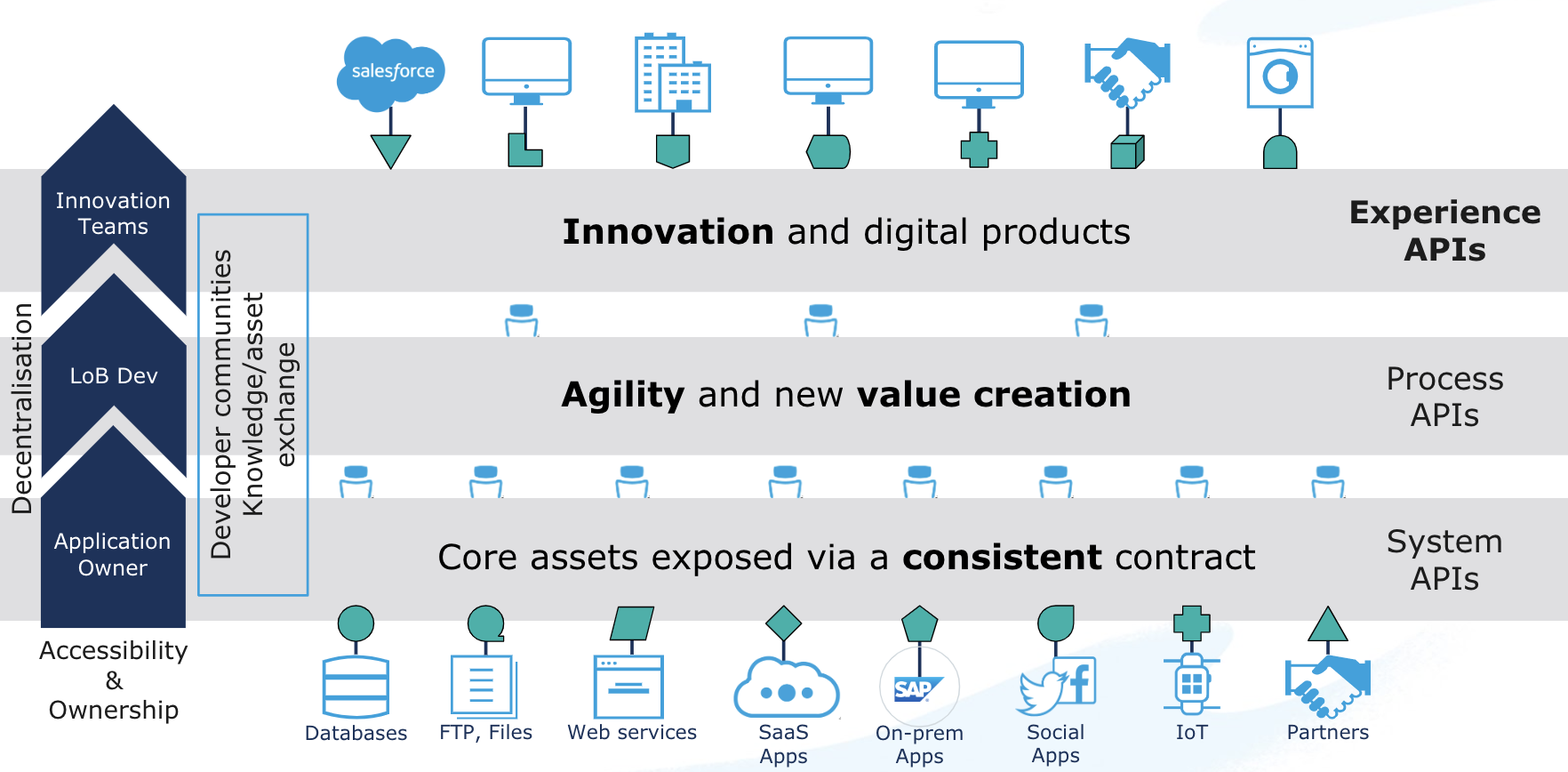
The key goal of the approach recommended by this document is to introduce reusable building blocks that can be reused both during the initial implementation as well as by future projects, resulting in reduced development effort.

**API-led connectivity** is a methodical way to connect data to applications through a series of reusable and purposeful modern APIs that are each developed to play a specific role – unlock data from systems, compose data into processes, or deliver an experience.

The API building block is a product that consists of functionality and simplicity required for the full lifecycle of APIs. This lifecycle consists of the ability to compose the data and connect to any other source of data. And it must provide full visibility, security, governance right from design.

The diagram below illustrates the API-led connectivity approach composed of three main layers:

* System APIs to unlock backend systems through a consistent contract, making use of our extensive connectivity,
* Process APIs providing orchestration and transformation into business domain objects for greater agility and value creation and
* Experience APIs focused on rendering information specific to devices or consuming client applications.

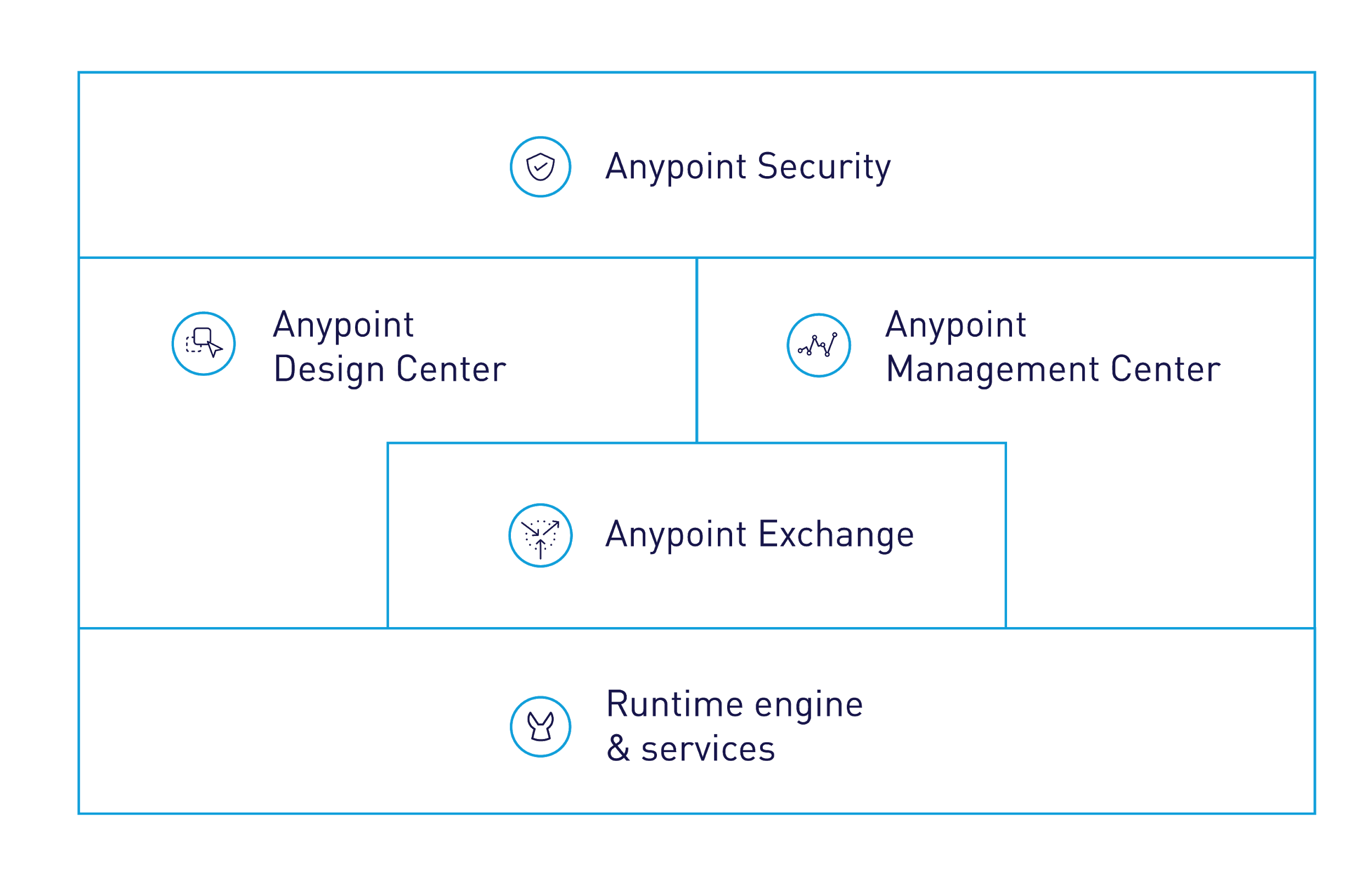


The value of this approach is enabling a flexible, agile architecture built for reuse and consumption, to speed up project delivery with built in governance and security.

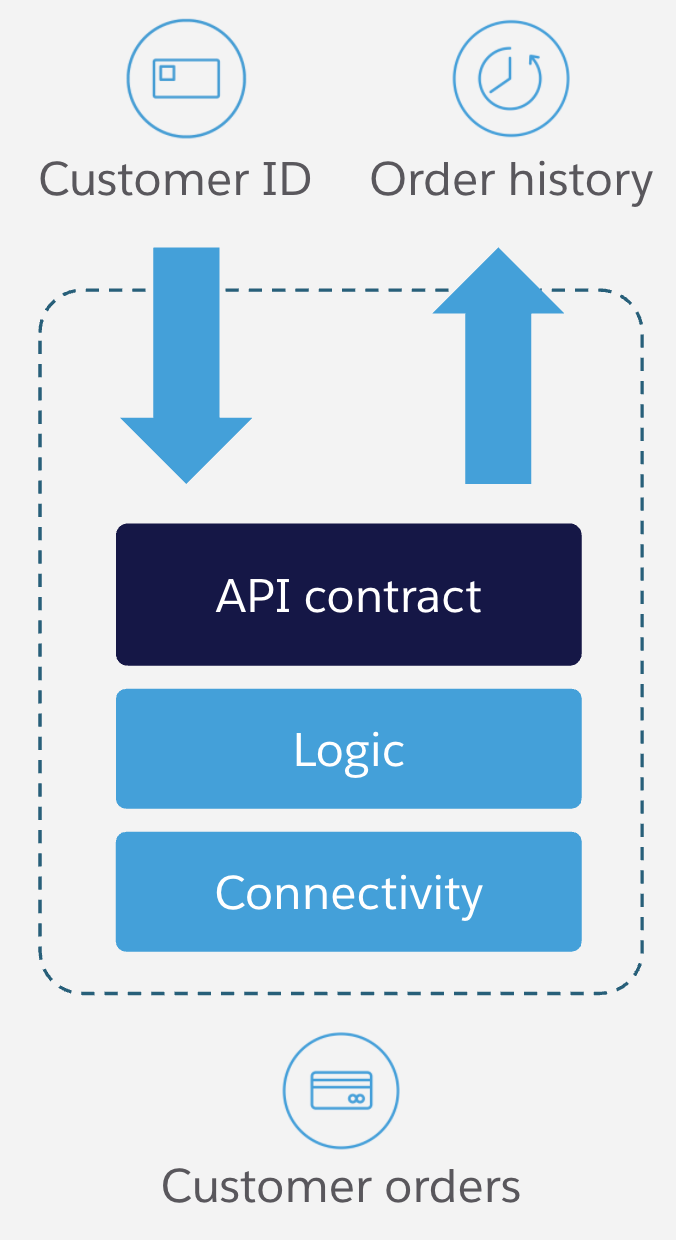
## MuleSoft Anypoint Platform

The Anypoint Platform has five major components and will be the core to future integration capabilities:

* **Exchange**: Marketplace of API and integration assets, promotes reuse of pre-built connectors, templates, examples, and APIs
* **Design center**: Comprehensive tools to develop APIs and integrations faster and easier
* **Mule runtime engine & services**: Single runtime for your Mule applications, easily configurable, performant, and deployable anywhere
* **Management center**: Manage and monitor your applications across the platform in a single place
* **Security**: Safeguard sensitive information with layers of protection



## What is an API?



An API is comprised of the following three aspects:

1. **API Contract**: Describes data formats, transport and protocols that are used to consume this API
2. **Logic**: The implementation of the API, including data transformation, logical flow control, error handling, etc.
3. **Connectivity**: Adapters for translating to external application interfaces, including protocol translation, data format transformation, security, etc.

## MuleSoft Connectors

To meet the connectivity requirements of this solution, the following MuleSoft Connectors will be used:

*Adjust according to your solution context.*

1. Salesforce Connector (Platform Events, sObject CRUD operations)
   1. Salesforce SOAP, REST, Bulk, Platform Events, and Streaming APIs
2. Salesforce Composite Connector (Salesforce Composite API)
3. Web Services:
   1. SOAP
   2. REST
4. Database
5. SFTP

## API Application Catalogue

A list of the proposed deployed API applications is provided below. Note that this list is based on the information provided during the discovery workshops and validated during playback sessions. A single deployed API application can support multiple aspects of the logical integrations as described in section 1 of this document.

To interpret this table, it is important to first understand the MuleSoft API-led Connectivity approach as described in section 2 and in more detail here:

<https://www.mulesoft.com/lp/whitepaper/api/api-led-connectivity>

*Replace all the examples in the table below according to your solution context.*

|  |  |  |
| --- | --- | --- |
| **API Name** | **Type (E/P/E)** | **Function (+ logical integration catalogue references)** |
| OMS | Experience | Entry point for Salesforce Order Manager (aka OMS).  Fulfilment Orders, Invoices, Credit Memos, Payment Confirmations, Generate and Print Dispatch Documents. |
| C&C | Experience | Entry point for the Click and Collect Application.  Order Ready for Collection. |
| Order Fulfilment Notification | Process | Logic for processing Order Fulfilment notifications and routing to the relevant RMS / SCE / Drop Ship System APIs. |
| Get Order | Process | Logic for retrieving Order information, uses the OMS System API. |
| Order Status Update | Process | Logic for processing Warehouse Order Status Update messages and routing to the relevant OMS System API operation (Shipment Confirmation, Short Pick, Status Update). |
| Generate Dispatch Docs | Process | Logic for routing to the relevant Dispatch Partner System API (assumes one for each of the four Dispatch Partners - DHL, DPD, UPS, and Yodel). |
| Print Dispatch Docs | Process | Abstraction of the Warehouse Print Dispatch Docs logical function, routes to the Warehouse Dispatch Print System API. This approach allows for additional Dispatch Partners to be added with minimal impact. |
| Invoices | Process | Logic for fetching the Invoice object (via the OMS System API) and routing to the appropriate Invoice system based on payload data. Invokes the upload invoice operation on either PoS or Aptos Invoice System APIs. |
| Credit Memos | Process | Logic for fetching the Credit Memo object (via the OMS System API) and routing to the appropriate Invoice system based on payload data. Invokes the upload invoice operation on either the PoS or Aptos Invoice System APIs. |
| OMS | System | Provides operations for interacting with OMS, specifically Get Order, Get Invoice, Get Credit Memo, Order Status Update, Shipment Confirmation and Short Pick. |
| SCE | System | Operations for interfacing with SCE, including Fulfilment Orders and Payment Confirmations. |
| RMS | System | Functionality for sending Fulfilment Orders to RMS as well as polling for Inventory Position updates. |
| Drop Ship Partner | System | Sending Fulfilment Orders to Drop Ship Partners. Currently assumes file based, however, REST API integration can be provided in the future. |
| PoS | System | Support for Invoice and Credit Memo integration with PoS application. |
| Aptos | System | Support for Invoice and Credit Memo integration with the Aptos application. |
| Non-RMS Concessions | System | Poll for non-RMS Inventory Position updates. |
| OCIS | System | Functionality for sending Inventory Position updates to OCIS. |
| Dispatch Partner | System | This document assumes that there are four Dispatch Partner System APIs - one for each of the Dispatch Partners (DHL, DPD, UPS, and Yodel). |
| Warehouse Dispatch Print | System | Functionality for low-level integration with the to-be Warehouse Dispatch Print application. |

## Reusable APIs

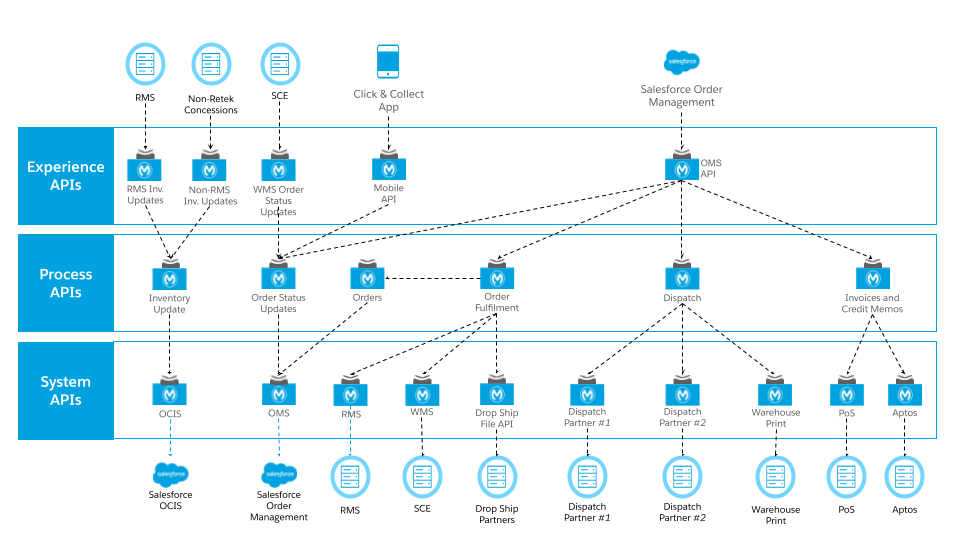
The following core APIs are proposed to maximize reuse:

*Replace the list below according to your solution context.*

1. Order Fulfillment Notification
   1. **Consumers**: OMS
   2. **Endpoints**: RMS, SCE + Drop Ship Partners (possibly multiple variants)
2. Get Order
   1. **Consumers**: Integration Platform (from multiple locations)
   2. **Endpoints**: OMS
3. Order Status Update
   1. **Consumers**: SCE (possibly also Click & Collect App for Order Ready for Collection status updates)
   2. **Endpoints**: OMS (generic Order Status Update), OMS (Shipment Confirmation), OMS (Short Pick), OMS (Ready for Collection - if C&C App)
4. Generate Dispatch Documentation
   1. **Consumers**: OMS
   2. **Endpoints**: Multiple Dispatch Partners (assume that each have their own unique System API)
5. Print Dispatch Documentation
   1. **Consumers**: OMS
   2. **Endpoints**: Warehouse Print Solution
6. Invoices
   1. **Consumers**: OMS
   2. **Endpoints**: Aptos, PoS
7. Credit Memos
   1. **Consumers**: OMS
   2. **Endpoints**: Aptos, PoS

This can be visualized in the following layered API-led network diagram:

*Recreate according to your solution context.*



# 

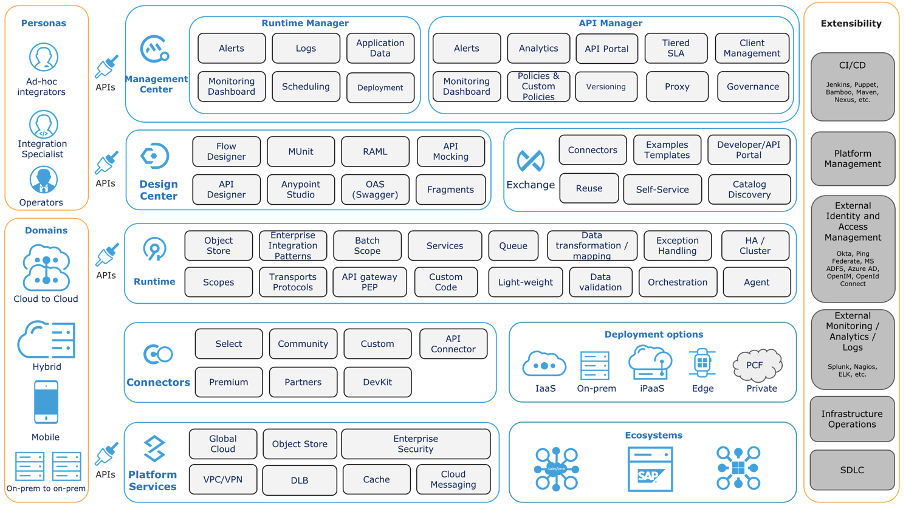
# MuleSoft Platform Reference Architecture

In line with the customer’s cloud-first strategy, the MuleSoft integration platform will be provided by CloudHub. CloudHub is the platform as a service (PaaS) component of Anypoint Platform - a global, fully managed, multi-tenanted, secure, and highly available platform for APIs and integrations. A CloudHub deployment model provides the following benefits:

* Get continuous software updates, no complex hardware maintenance required
* Improve efficiency with multi-tenancy for applications and workers
* Enjoy a globally distributed architecture with 99.99% uptime
* Leverage out-of-the-box cloud security and compliance
* Control access based on complex organizational requirements

This section serves as an initial Platform Reference Architecture that will provide a scalable and performant foundational architecture for ongoing and future integration and API development. The architecture provides common reference specifically around the physical, security, and deployment aspects of the platform.

## Platform Capabilities



As per the diagram above, the MuleSoft Anypoint Platform capabilities can be categorised into high-level groups of capabilities, each of which are briefly described below with links to detailed product documentation.

### Runtime Manager

Runtime Manager provides capabilities to:

* Deploy and manage all Mule Applications from one central location, regardless of where the applications are running
* Monitor applications and raise alerts based on the configurable criteria to detect events of interest
* View and control any Virtual Machine (VM) or Staged Event-Driven Architecture (SEDA) queues within the flows of your deployed applications (only when running on cloud).

See <https://docs.mulesoft.com/runtime-manager/> for additional information.

High Availability:

* High Availability through Servers, Server Groups and Clusters
* High Performance to hundreds of thousands of Transactions per second
* Vertical and horizontal scalability to hundreds of Mule Runtimes deployed in a highly distributed Hybrid Environment
* Automated Failover
* Message Persistence.

See<https://docs.mulesoft.com/runtime-manager/managing-servers> for additional information on managing Servers, Server Groups and Clusters.

Monitoring Dashboards:

The platform’s monitoring capabilities provide the following capabilities:

* Visibility into integrations across your application network
* Event-driven alerts to identify abnormal behavior, e.g. error responses, thresholds exceeded, etc.
* API Function Monitoring to automatically check the health of individual APIs
* Feedback from Mule flows and components in an application network.

See<https://docs.mulesoft.com/monitoring/> for additional information.

Visualizer provides:

* Views of different aspects of an application network graph
* Navigate / explore an application network to support troubleshooting and architectural and policy compliance reviews.

See<https://docs.mulesoft.com/visualizer/> for additional information.

### API Manager

API Manager provides capabilities to:

* Manage APIs through the configuration and application of multiple, ordered, out-of-the-box policies (e.g., Client ID enforcement, Header Injection, etc.)
* Create and apply Custom Policies to meet the specific needs of your enterprise (e.g. Security, Traffic Management, Throttling, Compliance, etc.)
* Manage API consumers (e.g., Authorization, individual Rate Limits, SLA application, etc.)
* Monitor and raise Alerts for individual APIs based on API-specific metrics.

See<https://docs.mulesoft.com/api-manager/2.x/latest-overview-concept> for additional information.

### Design Center

Design Center provides capabilities to:

* Design and implement APIs using either Anypoint Studio or Flow designer tools
* Automate API testing by using MUnit
* Integrate with multiple Source Code Management systems (SCM) such as Git, BitBucket, TFS, or Subversion
* Mock an API before it is built. This allows API consumers to start playing with the API under development and provide feedback.

See<https://docs.mulesoft.com/design-center/> and<https://docs.mulesoft.com/studio/7.6/> for additional information.

### Exchange

Exchange provides capabilities to:

* Publish and discover reusable assets such as APIs, Mule Applications, Connectors as well as API or Application Templates
* Publish and discover reusable API Fragments such as Data Types and Traits
* Create and expose API Developer Portals that include an API Console to allow developers to understand what the API does and send requests to (a Mock / Sandbox version) of the API.

See<https://docs.mulesoft.com/anypoint-exchange/> for additional information.

### Mule Runtime

The Mule Runtime is a lightweight Java-based engine that provides the following capabilities:

* Allows developers to connect applications together quickly and easily, enabling them to exchange data
* Enables the easy integration of existing systems, regardless of the different technologies that the applications use, including HTTP, Web Services, JMS, JDBC, and more
* Can be deployed anywhere, can integrate and orchestrate events in real time or in batch, and has universal connectivity (Anypoint Connectors).

See<https://docs.mulesoft.com/mule-runtime/4.3/> for additional information.

### Connectors

Anypoint Connectors facilitate the integration of Mule Applications with third-party APIs and Systems, providing a means to access Web Services and Resources. Using Anypoint Connectors within Mule Applications allows for sending and receiving data over a protocol or using an API.

A large set of connectors is available for use and new connectors can be created using the Mule Java and XML SDKs.

See<https://docs.mulesoft.com/connectors/> and<https://docs.mulesoft.com/mule-sdk/1.1/> for additional information.

### Platform Services

#### Anypoint Security

The Anypoint Platform takes a layered approach to securing application networks and is split between platform security and API security.

The key capabilities of platform security include:

* Block unauthorized access to systems through Filters and Policies
* Prevent exposure of sensitive data with encryption and Digital Signatures
* Secrets Manager provides secure vaults for TLS certificates and keystores
* Control access to APIs with proven security standards like OAuth2, SAML or LDAP.

See<https://docs.mulesoft.com/anypoint-security/> for additional information.

The key capabilities of API security include:

* Securing application configuration properties
* Cryptography module
* FIPS 140-2 compliance support
* Integration with Spring security
* TLS configuration
* OAuth Authorization
* Secure Token Service

See<https://docs.mulesoft.com/mule-runtime/4.3/securing> for additional information.

#### Virtual Private Cloud (VPC)

VPCs allow for securely partitioning networks and environments within CloudHub. The MuleSoft base subscription includes 2 VPCs, one VPC can be associated with multiple environments. Organizations typically have separate Production and non-Production VPCs, with only the Production environment in the Production VPC, and all other environments (Test, Sandbox, etc.) in the Non-production VPC.

See<https://docs.mulesoft.com/runtime-manager/virtual-private-cloud> for additional information.

#### Cloud Messaging (Anypoint MQ)

Anypoint MQ provides capabilities to:

* Perform advanced asynchronous messaging scenarios such as queueing and publish-subscribe with fully hosted and managed cloud-based Message Queues and Exchanges
* FIFO queues to retain message ordering
* Consume and produce messages using REST APIs
* Supports different environments, Business Groups, and role-based access control (RBAC).

See<https://docs.mulesoft.com/anypoint-mq/> for additional information.

Note that, at the time of writing this document, it is assumed that this functionality will not be required for the «xyz» project.

#### Object Store

Mule Object Store v2 is a mechanism for storing arbitrary objects in Mule using key-value pairs. By using a Mule Object store, applications are decoupled from any specific persistence implementation, allowing you to choose or switch the implementation you want. Object Stores can be accessed either via the built-in connector or via a REST API. Note that entries are API-specific when accessed via the connector.

See<https://docs.mulesoft.com/object-store/> for additional information.

#### Dedicated Load Balancers (DLB)

DLBs provide capabilities to:

* Handle load balancing among the different CloudHub workers that run your application
* Define SSL configurations to provide custom certificates
* Optionally to enforce two-way SSL client authentication
* Configure proxy rules that map your applications to custom domains. This enables you to host your applications under a single domain.

See<https://docs.mulesoft.com/runtime-manager/cloudhub-dedicated-load-balancer> for additional details.

### Extensibility

The Anypoint Platform provides multiple extensibility mechanisms, e.g.:

* CI/CD automation using Jenkins, Puppet, Bamboo, Maven, Nexus, etc.
* Integration with external Identity and Access Management Systems such as Okta, PingFederate, MS ADFS, Azure AD, OpenIM and OpenId Connect
* Integration with external monitoring, log aggregation and analytics tools such as Splunk, Nagios, ELK, etc.
* Integration with external infrastructure operations tools such as Docker, Kubernetes and OpenShift (on-premises runtimes).

## Platform Setup

### Deployment Regions

The Mule control plane region will be the default (US).

The default Mule Runtime region should be set to “«name of region»”.

### Business Groups

Organization hierarchies can be defined in CloudHub to partition CloudHub resource allocations, security roles, environments, and API deployments. The root of the hierarchy is the Anypoint Platform Organization, the Business Group hierarchy sits underneath the root Organization. The Anypoint Platform’s Access Management capabilities allow the creation of a hierarchy of Business Groups to support fine grained configuration of access to individual resources.

The following sections describe the key features of the organizational platform account structure.

#### Organization

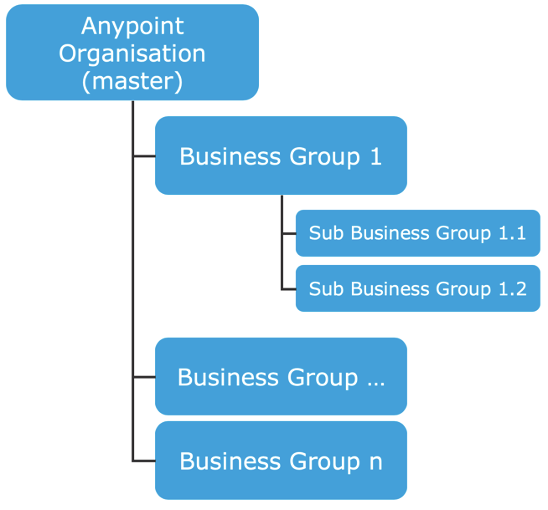
When you create an Anypoint Platform account, a master (or root) organization is created. The organization name is based on the ‘Company’ field in the initial Anypoint Platform signup form.

#### Organization Owner

The user who first signs up for an Anypoint Platform account is designated as the organization owner. This is not a role but an identifier for this single user, who is assigned the Organization Administrators role by default. Organization Administrators can view and configure properties for an organization.

#### Business Groups Explained

Business Groups are also referred to as organizations (or sub organizations). Business Groups reside within the master organization and can contain their own Business Groups. Thus, the master organization and its Business Groups are organized in a hierarchical tree, where the master organization is the root. Each Business Group has one direct parent and can have multiple children, e.g.:



The organization administrator is also the administrator of all Business Groups within the account’s organization hierarchy.

#### Business Group Structure

The following initial Business Group structure is recommended:

* Root Organization: «customer name»
  + Business Group: «solution context»

### Environments

APIs are deployed to Environments; Environments belong to a single Business Group. Anypoint Platform defines three Environment Types:

1. **Design**: Enables API designers to test and run applications at design time. This environment type is used exclusively for APIs developed using Flow Designer within Design Center and consumes Design vCore allocations
2. **Sandbox**: Provides useful environments for development and testing and enables you to safely test an application without affecting the production environment. Environments of this type consume Pre-Production resource allocations
3. **Production**: The production environment where you can deploy applications and APIs. Environments of this type consume Production resource allocations.

The following environments will be created in the “«solution context»” Business Group:

|  |  |
| --- | --- |
| **Environment Name** | **Environment Type** |
| dev | Sandbox (Non-production) |
| test | Sandbox (Non-production) |
| prod | Production |

As stated previously, roles can be applied at the master organization level as well as at the business group level. Within Business Groups some roles and permissions are valid for the entire organization (=all environments) like the API designer role, and some are only valid for a specific environment like the permissions for deploying and starting applications.

## Platform Security

### Access Management

Anypoint Access Management enables Anypoint Platform users to implement and manage resources and assign users, roles, and permissions in a way that makes sense for their unique organizational structure. It facilitates safe testing by creating sandbox environments, restricts access to production, and inspects all resource usage via an audit log that can be queried.

Access Management provides the following Platform and Security capabilities:

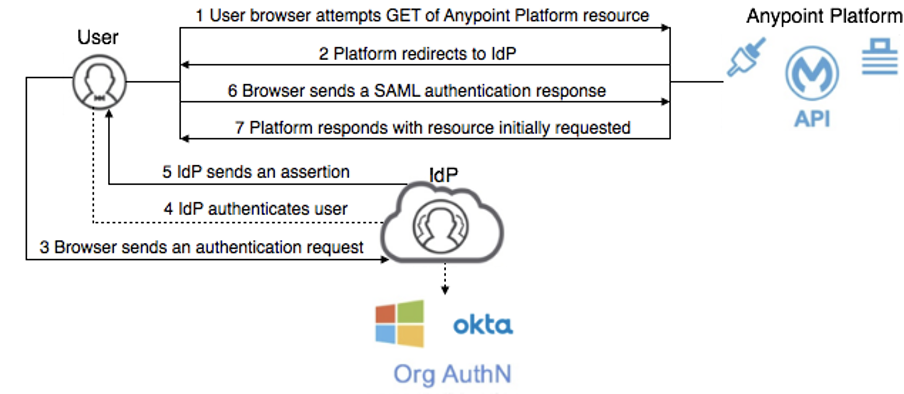
* Federate access and delegate administration to Business Groups
* Assign users to predefined or custom roles with fine-grained permissions
* Isolate sandbox and production environments
* Query a comprehensive audit log of user interactions via UI or API
* Authenticate users and clients with external identity solutions.

### Identity Management

The Anypoint Platform can be configured to use an external Identity Provider using one of the following Single Sign-On (SSO) standards:

* OpenID Connect: End-User identity verification by an authorization server including SSO
* SAML 2.0: Web-based authorization including cross-domain SSO.

The following diagram shows the high level SAML based identity management process:



See<https://docs.mulesoft.com/access-management/external-identity> for further information on configuring external Identity Providers.

### Platform Roles and Permissions

#### Platform Role Types

The Anypoint Platform provides two types of roles:

1. **Default roles**: roles that are created automatically when an organization or business group is created. These roles provide permissions and access to core functionality of Anypoint Platform. You can modify default roles by assigning users or other roles. However, you cannot delete a default role.
2. **Custom roles**: roles that a user can create, customize, and delete. You can assign users or roles to a custom role. You can also associate a custom role to a specific Anypoint Platform product. For example, you can create a custom role called Application Designer and assign that role to the Design Center application. This enables users assigned the Application Designer role to access the Design Center application.

Note that roles and permissions are specific to a Business Group.

See<https://docs.mulesoft.com/access-management/roles> for further information.

#### Default Platform Roles & Permissions

* **Organization Admin**: Enables a user to edit to all versions of all APIs, all registered applications, and all API Portals in the Anypoint Platform. Permits access to the Organization Administration page, where they can add and manage users and roles, view and edit organization details, access API Manager and manage Client Applications and API policies, access the client ID and client secret for the organization, and customize the theme of the Developer Portal. Members of the Organization Administrator role also inherit the role of API Creator by default.
* **API Creators**: Enables a user to create and manage API versions in the Anypoint Platform for APIs. Members of the API Creator role have the ability to add new APIs to the platform on the API administration page. This role does not grant privileges for Runtime Manager.
* **Portals Viewer**: Enables a user to view a list of the Private API Portals to which they have Portal Viewer permissions from the Developer Portal. They can also click to view those API Portals. The ability to view an API Portal does not automatically give a user access to the API.
* **API Versions Owner**: Enables a user to manage, delete, and deprecate any API in the organization. They can edit the portal of any API in the organization.
* **Audit Log Viewers**: Enables a user to access the UI for the Audit Log under Access Management.
* **CloudHub Admin**: Provides access to all Runtime Manager functionality.
* **Exchange Administrators**: Manage Exchange Portal Enables a user to manage Exchange Portals, including customization, manage assets, manage reviews.
* **Exchange Contributors**: Enables a user to contribute Exchange assets and manage versions.
* **Exchange Viewers**: Enables a user to view and consume Exchange assets.

## Proposed Deployment Architecture

### Environments

As described previously, CloudHub has the following environment types:

1. Design
2. Sandbox (aka Non-production)
3. Production

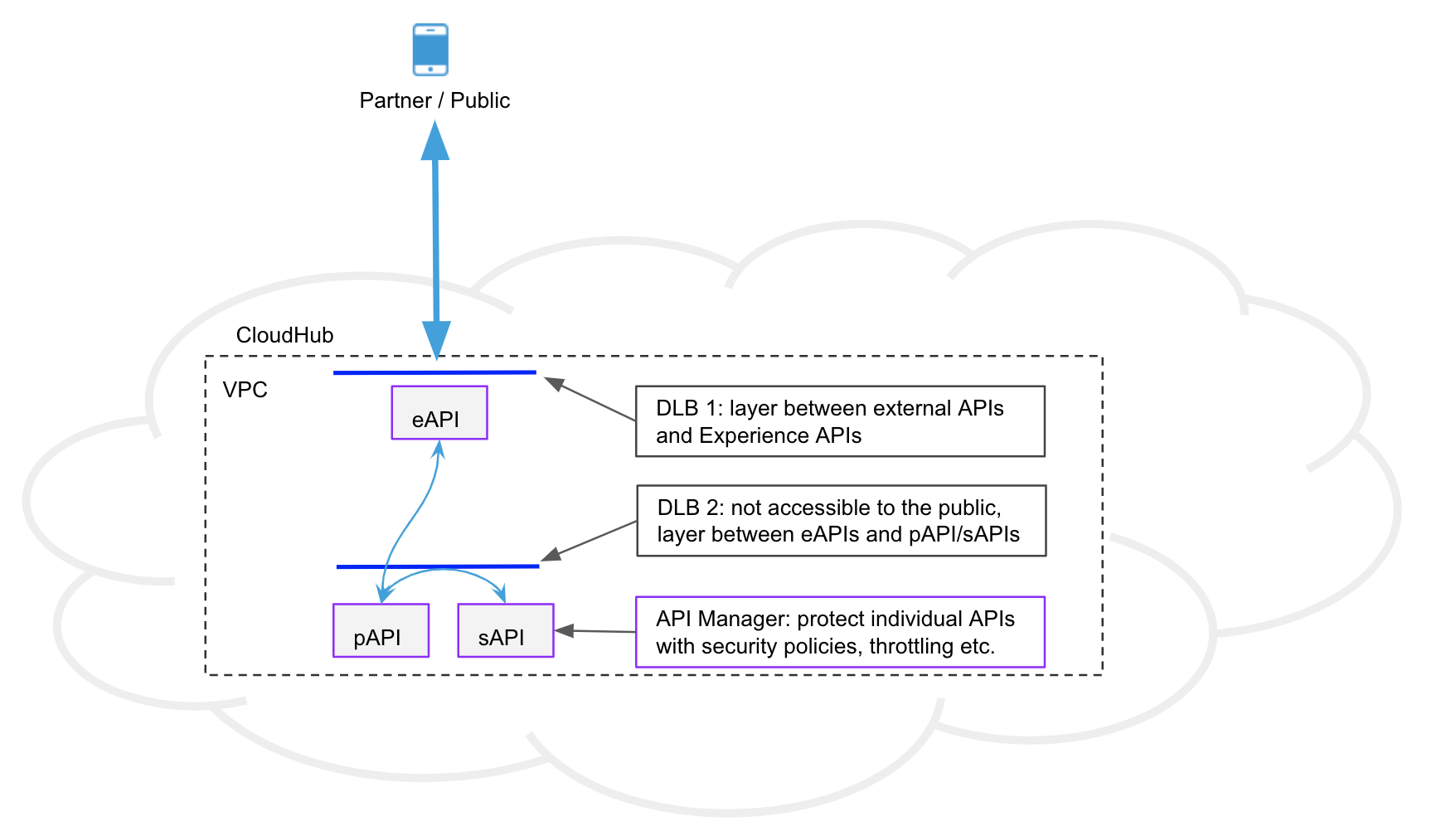
The Design environment type is only used when building APIs using Flow Designer within Design Center and does not support custom VPCs, VPNs and DLBs hence will not be used by the dev, test, and prod environments. This document assumes the following configuration:

|  |  |
| --- | --- |
| **Environment Name** | **Environment Type** |
| prod | Production |
| test | Sandbox (Non-production) |
| dev | Sandbox (Non-production) |

### Dedicated Load Balancers

MuleSoft recommends a pair of Dedicated Load Balancers per environment type, i.e., two for Production and two for Sandbox (Non-production). This provides clean separation and an additional layer of security between APIs that are exposed to external consumers (Experience APIs) and internal APIs (Process and System APIs).

This is illustrated in the diagram below.



### Virtual Private Clouds

Two Virtual Private Clouds (VPCs) will be required - one for Production and one for Sandbox (Non-production). VPCs are required to support custom Dedicated Load Balancers (DLBs).

### Virtual Private Networks

It is assumed that two Virtual Private Networks (VPNs) will be required for CloudHub to connect to on-premises systems, one for Production and one for Sandbox (Non-production).

# 

## About MuleSoft, a Salesforce company

MuleSoft, provider of the world’s #1 integration and API platform, makes it easy to connect [data](https://www.mulesoft.com/integration-solutions/dataweave-integration) from any system – no matter where it resides – to create connected experiences, faster. Thousands of organizations across industries rely on MuleSoft to realize speed, agility and innovation at scale. For more information, visit <https://www.mulesoft.com>.

*MuleSoft is a registered trademark of MuleSoft, LLC, a Salesforce company. All other marks are those of respective owners.*