**APIC Documentation**

1. API Connect is one of the integration tool of Datapower.
2. To develop API’s we use IBM API Connect.
3. The Run-time of API Connect is Datapower, which means API Connect is always dependent on Datapower.
4. APIC is based on the specific user feedback, which means the features will be available only for that particular user.
5. When providing services with extra features rather than what is required, then its mostly not a use.
6. APIC is customer level, which means we can change our service according to the need of cost. (Eg: Netflix, Aha, Hotstar, Prime etc.)

**APIC Components**

1. **Cloud Manager:**

API connect is a service which connects different API’s.

It contains LDAP, SSL (or) TLS, Gateway, servers, Provider Org., members.

Cloud manager is only accessible for admins / architects

1. **API Manager:**

It contains all API’s, products, members, catalogs and spaces.

The config. That are done in Provider Org. will be visible in API Manager.

It is a tool to define REST API’s. It is a tool to monitor API transactions.

API’s and Products are mapped into Drafts, where we have multiple number of

API’s and plans in Products.

Products

DRAFTS

API’s

Drafts is the collection of Products and API’s.

A Dashboard is a visual display of all the data (or) it is an overview of data

From different resources. In API Connect, dashboard is the overview of all

Catalogs.

1. **Developer Portal:**

* It is the marketplace for API Connect.
* It is used to explore, discover and subscribe to API’s.
* API’s are available in developer portal when API providers publish it in

API manager.

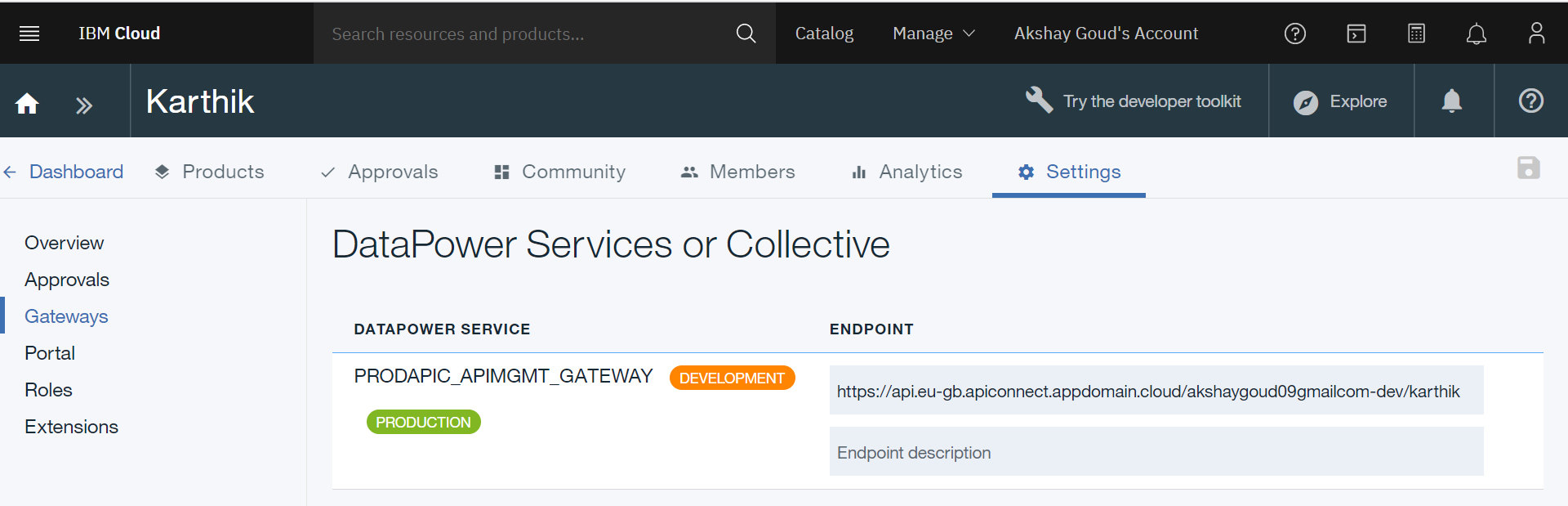
* Each catalog has its own developer portal.
* It also provides blogs, forums and we can review API’s too.
* It consists of APPS, Products (API’s & Plans) and the members.
* Developer Portal will always mapped to Catalog.

1. **Gateway service:**

* It operates between client and backend services.
* It acts as a proxy to accept API calls.
* API connect uses IBM Datapower as its gateway service.
* Whenever we want to call any API service, then we need to call the end

Point.

* It is similar to Datapower.
* Runtime is Datapower.



* If API is deployed, then the below end point is used

<https://api.eu-gb.apiconnect.appdomain.cloud/akshaygoud09gmailcom-dev/karthik/(basepath)/(pathdetails)?(ifAnyQueryParameters)>

1. **Developer Toolkit:**

It provides a command line tool for creating and testing API’s that we can

Run, manage with API connect. It is like command prompt which we use in

Java.

**SWAGGER:**

Swagger is Open API. It is a universal format to define API structure, with .yml and .Raml.

* **Info:** It generally contains basic information like Title of API, name,

Version, description and external doc’s.

* **Schemes:** It defines which transfer protocol you want your API to use.

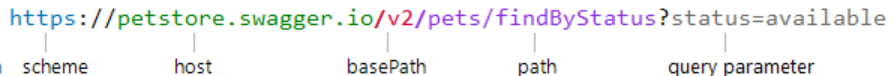
(http, https, ws, wss).

* **Host:** What ever the gateway is in the catalog, will get reflected in

$(catalog.host) -> default

* **Basepath:** It is the URL prefix for all API paths. It should start with a forward

Slash (/) following with path name.



* **Consumes:** The type of data that API takes as input.
* **Produces:** The type of data that API gives as output.
* **Lifecycle:** It consists of 3 phases:

**Identified:** API is in initial phase and is not designed fully nor

Implemented.

**Realized:** It is a default state. API is in implementation phase.

**Specified:** API is fully designed but not yet implemented.

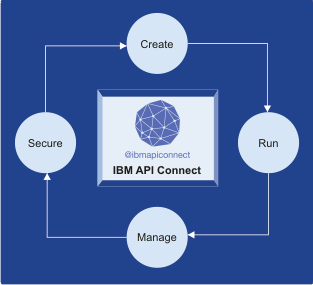
* **Security Definition:** It shows which type of security we want to use.

3 types: API key, Basic, OAuth.

* **Security:** It shows the selection of how many types of security implementation.
* **Properties:** It is used to define values.
* **Paths:** It is used to add parameters by assigning name, location and data type.

Also used to add different types of operations (GET, PUT, POST, DELETE etc.)

**Phases Of API:**



**Product Life Cycle:**

Product lifecycle consists of 5 stages:

* **Stage:** It is the initial stage when you publish the product. It is in development mode

(or) in AUTO Subscription.

* **Publish:** It shows all the subscriptions.
* **Retire:** When you retire a product, the product version can neither be viewed nor

the plans are subscribed.

* **Deprecate:** When you deprecate a Product, the product version is visible only to developers whose applications are currently subscribed. New users cannot subscribe.
* **Supersede:**