**TM Forum Open APIs**

**Conformance Certification**

*Company Name:* ***Sterlite Tech***

*TM Forum Open API Name:*

***Shopping Cart API TMF663***

*TM Forum Open API Release Version:* ***19.0 / 4.0***

**Report Date: 25th September, 2021**

1. **What Product or Solution does your API support?**

Shopping Cart API is managed by STL’s digital Engagement Platform (dEP) product. dEP provides Open APIs in line with TM Forum defined specifications that allows unassisted channel specific operations like User Management, Commercial Product Catalog Management, Shopping Cart Management, Payment management. This includes, but is not limited to, the Shopping Cart API, TMF 620 Aligned Microservice of STL. This document focuses on listing some of the capabilities of the Shopping Cart API.

dEP is a next-gen engagement platform that covers a wide spectrum of care and commerce needs of both consumer and enterprise segments. dEP helps reduce Customer Churn, improve Average Revenue Per User (ARPU) and maximize Customer Lifetime Value (CLV) by engaging more customers in a personalized way. It does so by delivering highly personalized and contextually relevant digital experiences across multiple touch-points, channels and devices. dEP helps operators cover a large surface area of customer engagement right from discovery, acquisition, onboarding, monetization, 360° view and Self-care.

1. **Overview of Certified API**

STL’s Digital Engagement Platform (dEP) provides digital commerce and care capabilities to enable telcos address the needs of the end customers in an unassisted manner, with a true omni-channel experience.

dEP provides multiple microservices such as User Management, Commercial Product Catalog Management, Configuration Management, Payment Management & Shopping Cart. This document focuses on the Shopping Cart Microservice, STL implementation of TMF 663 Open API.

Shopping Cart Micro service provides a standardized mechanism for the management of shopping carts for different lines of Businesses. Shopping Cart Microservice includes but is not limited to following capabilities:

**• Create Cart:**

* As an anonymous or logged in user, you can add items to the cart
* It creates the cart when you add the first item to the cart

**• Update Cart**

* Before making purchase, at any time user can update the cart
* Update can be changing the quantity, adding more items or removing existing items from the cart
* Shopping Cart Microservice automatically adds a product offer into cart if the product offer is defined as an auto picked offer in Unified Product Catalog. For e.g. If the customer is adding a mobility plan into the cart , Shopping Cart Microservice automatically adds a SIM card into the cart as it is defined as an ‘Auto-pick’ item, along with a mobility plan.

**• Retrieve Cart**

* System supports Abandoned cart handling
* Returning customers can resume their journey as items are always saved to the art and users can resume from there
* Retrieval of cart is supported based on different search criteria
  + Retrieve all
  + Retrieve specific cart items
  + Retrieve cart items with Specific fields i.e Total fields are 25 for cart Entity But only id needs to be retrieved
  + Support Exclude Operation for Retrieval i.e. Retrieve all fields except the specified ones
* Pagination is supported

**• Delete Cart :** There are two scenarios in which System will delete the cart Entity

* Once a cart is converted to Order, cart is not required to be persisted and hence, deleted
* After “Cart Expiry time” of abandoned cart, it is marked as “Expired” & deleted after configurable time. For example, if Cart Expiry time of abandoned cart is set to 7 days, the system would mark all the carts that are having creation date before 7 days, as “Expired” and Asynchronous Job would delete all the expired carts after configurable time.

• **Check Cart Compatibility, Eligibility & Cardinality Rules**

* **Compatibility**: Shopping Cart Microservice validates the compatibility of cart items based on the rules defined in UPC and does not allow customers to purchase in case of violation of compatibility rules.

* **Eligibility:** Shopping Cart Microservice also validates the eligibility of the customer to complete the order for selected cart items. This can be at the time of adding new items to the cart, and also at the time of purchase. It can be related to an item or number of items for e.g. You might be eligible for a device, but you might not be eligible to add multiple lines of subscriptions with that device.

* **Cardinality:** Shopping Cart Microservice validates the cardinality of Products configured in UPC. for e.g. Customers are not allowed to purchase a bundled iPhone plan without an iPhone. In this case, cardinality for the iPhone is defined as: Minimum Quantity 1. Cardinality indicates both minimum and maximum quality for the product offers that can be purchased within the given context, say bundled product offerings.

**• Do Cart Validation**

* Shopping Cart Microservice validates any business rules that are configured in the system to ensure that the customer is allowed to proceed ahead with the purchase only if certain conditions are met. For example:
  + More than x products are not allowed in a single order
  + Mobility offers cannot be purchased via guest user, Registration is mandatory for that because of different regulations
  + Prepaid & postpaid both are not allowed in a single order
  + Note : Rule configurations are flexible enough to enable/disable it
* Shopping Cart Microservice also takes care of different validations (business validations and product validations) every time a customer is trying to check out the cart. This ensures all criteria are met at the time of purchase.

Shopping Cart entity is used for the temporary selection and reservation of product offerings in e-commerce and retail purchase. Shopping cart supports purchase of both tangible and intangible goods and services (e.g. handset, telecom network services, Mobility Offerings, Fixed line services, Add Ons, Enterprise Offerings, etc..). It also supports Bundled Product Offerings (a Bundled Product Offering consists of multiple Simple or Bundled Product offerings such as Devices, Plans, Add Ons, etc..).

Shopping Cart contains a list of cart items, a reference to party or party role and can be associated with an Anonymous customer as well.

Shopping Cart also contains the calculated total price including the detailed Price Summary Such as :

• One Time charges for e.g. Device price, Accessories price, Activation fees etc.

• Recurring charges such as monthly charges of a postpaid mobility plan

• Deposit for handset or Roaming services

• Tax as applicable

• Discounts if any

For the tangible product offerings, it reflects the stock availability by showing “Out of Stock” or “In Stock” indication. Customers are able to add products into cart only if products are in Stock, but if you have a cart that's a few days old (for a returning customer), it validates stock availability of the items you have added to the card at the time of purchase.

For the tangible product offerings, customers can select the quantity of the product (for e.g. accessories), However for the service offerings, customers can add multiple instances into the cart for the same offer (for e.g. data packs).

**Personas supported:**

• Anonyms ( Unknown) user browsing the website

• Existing Subscriber ( Primary/Secondary user)

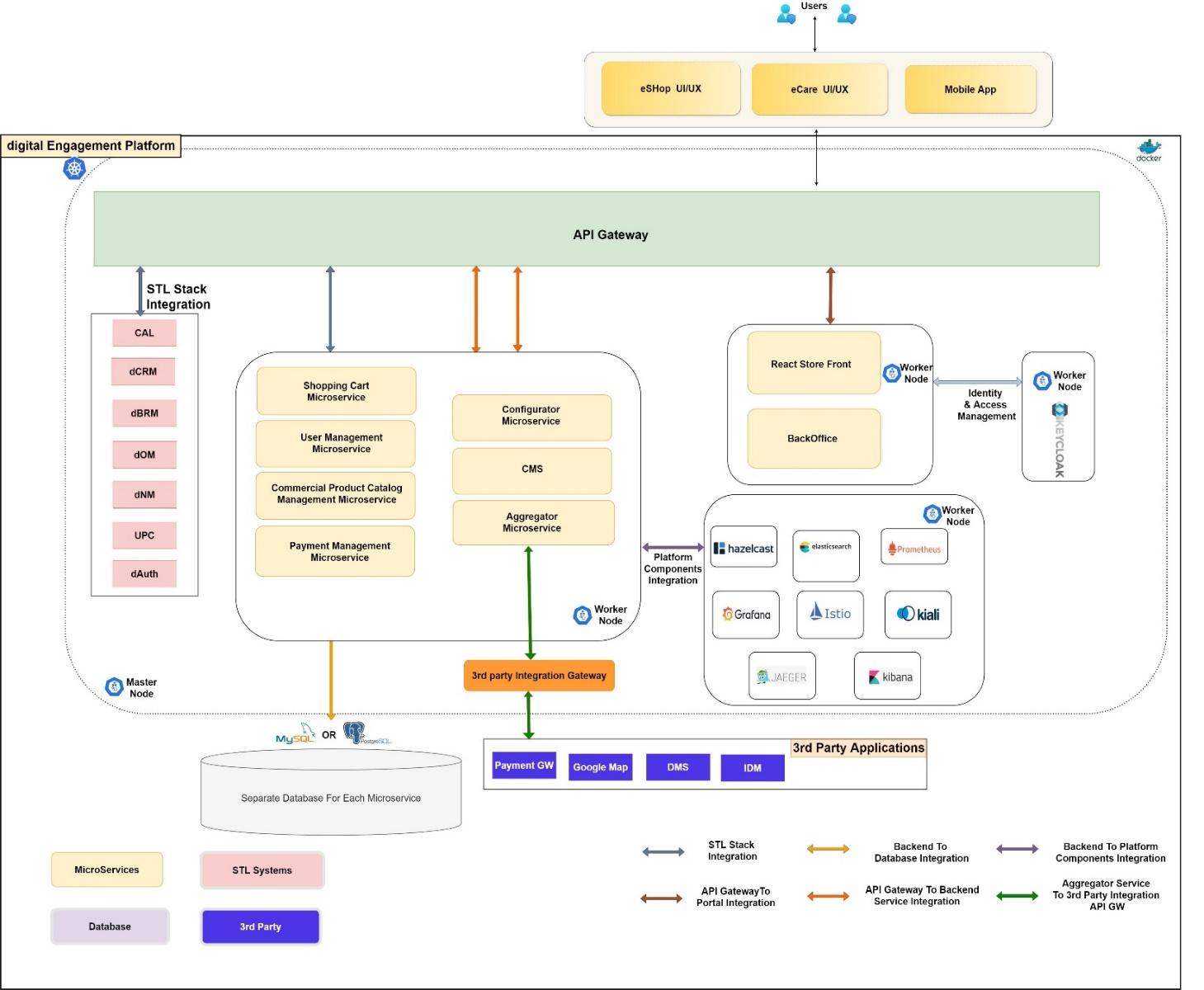
• Employee of an Organization in case of Enterprise Use cases

**Technical Capabilities:**

• Operations in shopping cart micro service are pluggable and customizable

• One can override the business logic by just extending the existing implementation

1. **Architectural View**



Digital Engagement Platform and all other STL products use STL’s Cloud Native Platform for platform services in a centralized manner. for e.g. Logging, Monitoring, Auditing, Cashing, Identity Management, Gateway, Service Mesh, Queue Management, Storage, Cloud native database management etc…

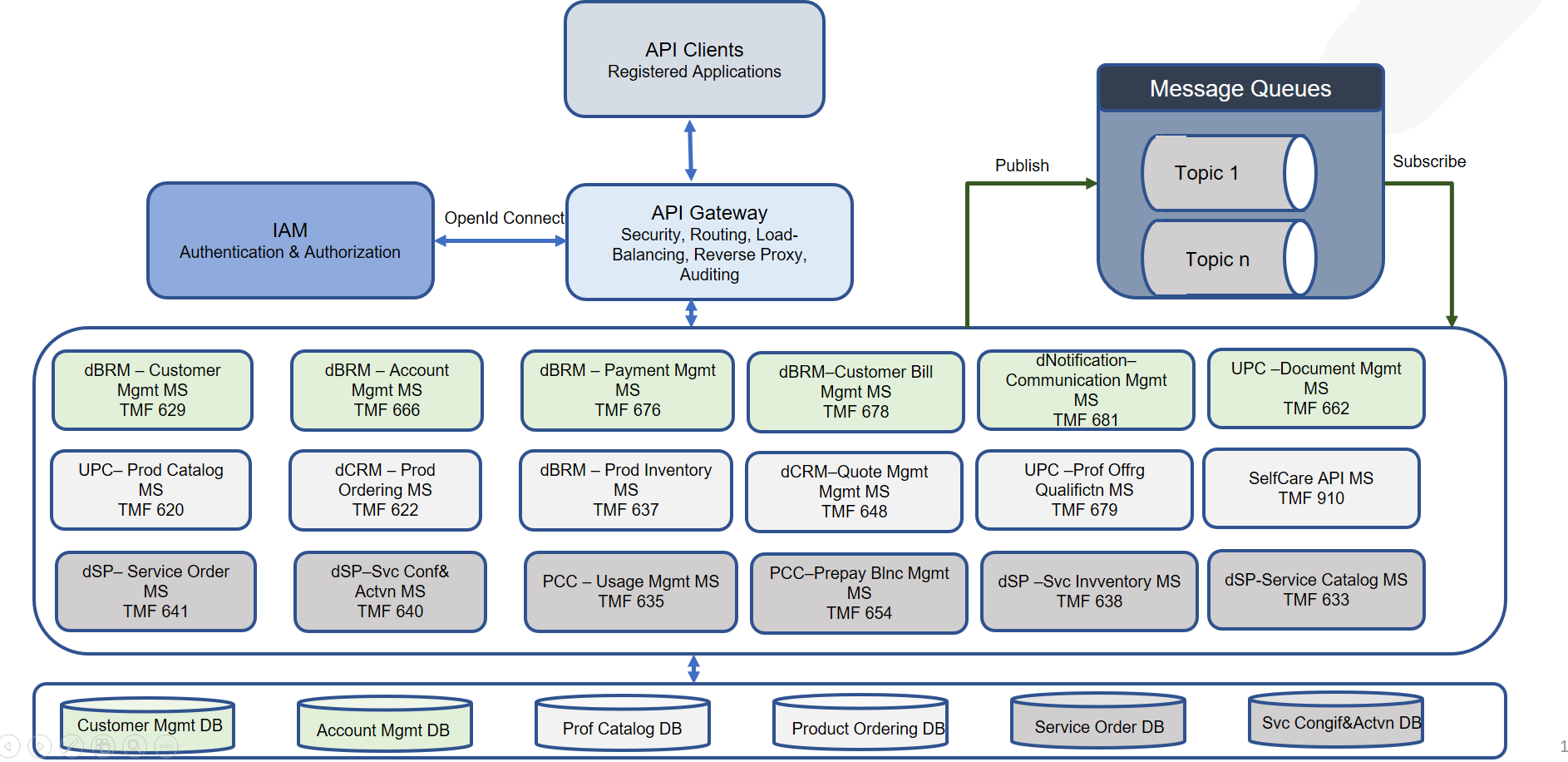
dEP (Digital Engagement Platform) provides TMF aligned microservices as Omnichannel APIs and that can be consumed by any external application/channels such as e-Shop, e-Care, Mobile Apps, Chatbots etc..

dEP integrates with STL Unified Product Catalog to implement eligibility and other checks via TMF679 – Product Offering Qualification, and maintains its own Commercial Catalog for better and faster search. dEP uses modern indexing and caching mechanisms to achieve advanced filtering.

dEP uses its inbuilt Aggregator Microservices to integrate with 3rd Party systems like Payment Gateway, Identity Management, Document Management, Deposit Management etc.. It has pluggable and extensible features that can be used as per external 3rd Party systems’ requirement for seamless integration.  The same Aggregator Microservices are used to integrate with other STL systems as well for e.g.

* dCRM: digital CRM, to retrieve all customer information
* dOM: digital Order Manager, to submit and retrieve order information
* dPCC : digital Policy Control and Charging, to retrieve unbilled usage charges
* dBRM: digital Billing Revenue and Revenue Management, to retrieve all billing information
* dNM: digital Notification Manager to send notifications to customers in centralized manner

STL implements TM Forum APIs as microservices in compliance with Open Digital Architecture. Below APIs are implemented by STL BSS:



APIs Details for respective ODA layers can be found below:

* ODA Party Management Layer :
  + STL Billing and Revenue Management (BRM) implements below TM Forum APIs:
    - Customer Management (TMF 629)
    - Account Management (TMF 666)
    - Payment Management (TMF 676)
    - Customer Bill Management (TMF 678)
  + STL Unified Product Catalog (UPC) implements:
    - Document Management (TMF 662)
* ODA Commerce Management Layer :
  + STL Billing and Revenue Management (BRM) implements:
    - Product Inventory APIs (TMF 637)
  + STL Unified Product Catalog (UPC) implements:
    - Product Catalog (TMF 620)
    - Product Offering Qualification (TMF 679)
  + STL Customer Relationship Management (CRM) APIs include:
    - Product Ordering (TMF 622)
    - Quote Management (TMF 648)
    - Self Care APIs (TMF 910)
  + STL Digital Notification (dNotification) APIs include:
    - Communication Management (TMF 681)
* ODA Production Layer :
  + STL Digital Service Provisioning (dSP) Module implements:
    - Service Order (TMF 641)
    - Service Configuration and Activation (TMF 640)
    - Service Inventory (TMF 638)
    - Service Catalog (TMF 633)
  + STL Policy and Charging Control (PCC) implements:
    - Usage Management APIs (TMF 635)
    - Prepay Balance Management APIs (TMF 654)

1. **Test Results**

