Universal Product Catalogue – The Foundation of Catalogue-Driven OSS/BSS Architecture

Robert Mei
Senior Enterprise Architect, Bell Canada

Raj Cherchattil
CTO, Telecom, Media & Entertainment, IBM Canada
Agenda

- Current Business Problem
- Objectives and Business Benefits
- Conceptual Solution
- Overview of Reference Architecture
- Digital Transformation Enabler
- Conclusion
Current Business Problems

- Qualification
  - Location Intelligence
  - Service Footprint
  - Qualification Policy

- CRM and Self-Serve
  - Order Capture
  - Pricing & Configuration
  - Product Catalogue
  - Customer Asset

- Customer Order Manager
  - Customer Order decomposition
  - Order Orchestration
  - Exception Jeopardy & Status

- Service Order Manager
  - Service Order decomposition
  - Service Assign & Design
  - CFS/RFS Orchestration

- Billing
  - Billing Account
  - Inquiry
  - Rating

Business Problems, Challenges and Limitations

- Product information manually entered in multiple systems during NTI/NPI potentially causing inconsistency
- Changes to products, services or resource specifications need to be manually implemented into multiple systems
- Transformation and mappings at system-to-system interface level
- Data modelling is not based on open standards
- Low re-usability of defined services limited by product-specific implementation
Business Drivers, Benefits and Architecture Objectives

**Business Drivers and Benefits**
- Establish foundation for future business transformation
- Prevent duplication, error and inconsistencies in product specifications, attributes and characteristic data across multiple systems
- Cost avoidance from a single product catalogue implementation such as integration (export and import)
- An interactive and collaborative product environment to support catalogues, access rights and workflows
- Provide a balance between domain and business unit independence and sharing
- Achieve the objective of the enterprise catalogue driven architecture
- Enable cross-system dimensional product analysis

**Enterprise Architecture Objectives**
- Develop a cross-domain and business unit PSR-based product catalogue across the enterprise
- Provide an interactive and collaborative product design environment and manage product lifecycle through governance process
- Support the UPC PSR model extraction, federation and integration with the client systems
- Develop product models following TMF PSR model to allow product specifications to be shared cross market segments

**End-to-End Product-Service-Resource (PSR) Modelling & Catalogue-driven Architecture Benefits**
- Decoupling of Product to Service to Resource relationship, enabling marketing or technological changes while limiting the system impacts
- Enables changes to products or offerings while limiting or avoiding impacts to fulfillment, and vice-versa
- Facilitate implementation of seamless access services (separation of technology from product)
- Prevents fragmentation and asymmetry between PSR components, and across different products
Catalog / Metadata Driven Architecture

Why?
- **End-to-end view of what is being sold** to the customer in a single place
- **End-to-end view of what has to be provisioned** and configured from a network perspective
- **Clear definition of products, services and resources** including their attributes, relationships and dependencies
- **Single place for definition** of prices, discounts and agreements across multiple products
- **Source of truth** for various applications including order management, customer billing, order capture, etc.
- Ability to quickly and efficiently introduce new, regional specific and personalized product offerings to the market place – **improve the NPI process**
- **Achieve maximum re-usability** of elements of the model, instead of rebuilding product-specific solutions

**Product offering**: One or more products for sale

**Services** are characterized as either being possibly visible and usable by a customer or not (Site Internet, Class 5 POTS, GPON, DNS, etc.).

There are Customer Facing Services (CFS) and Resource Facing Services (RFS).

**Resources** are capabilities required to deliver the Services. Can be physical (CPE device, Set-top box, etc) or logical (IP address, Channels, Phone #)
Conceptual Solution

**TMF SID-Based PSR Meta Model**
- The meta model to describe the various products in the future UPC implementation

**Product Models**
- The products which need to be defined in the MDM CE implementation
- The product model is enriched with attributes required to publish data to represent the complete product offering, pricing and product/service/resource specifications

**Distribution / Exports**
- The functionalities necessary to implement a distribution mechanism that will disseminate all relevant UPC data to downstream systems which consists of Trigger Mechanism, Transformation, Transfer, Audit/Backup/Log

**Product Lifecycle Management**
- The processes that need to be implemented in the UPC to support the end-to-end product lifecycle management

**Governance**
- The policies, rules and processes that ensure the effective use of systems/tools for maintaining a cohesive and holistic view of both the UPC Meta Model and the various product models

**Reporting**
- Offers the necessary reporting capabilities
Use of UPC with Order Management

Proposed End-to-End Architecture Benefits

- Enables re-use of system functions (and data) across multiple product lines and function-centric system specialization, as opposed to duplication in product-centric silos
- Centralization of customer information, order state information and unification of customer experience across channels
- De-couples the products/offers from the services required to realize them
- Enables gradual fulfillment automation of services while minimizing the impacts to CRM/self-service areas
- Reduction of fulfillment cycle time, and provide visibility to key metrics and KPIs
- Fosters easier and incremental changes to fulfillment business rules and processes (leveraging a business rules engine)
A Hybrid Architecture

Enable the Transformation to Next-Gen Services

- De-couples the product offers and customer-facing services from the network resources required to realize them
- Associate service fulfillment flows at decoupled customer-facing service level to enable hybrid architecture for both next-gen services and classic services that conduct the product offering
- Enables gradual transformation and automation of services fulfillment while minimizing the impacts to the product offering
- Allows collaborative OSS/BSS functions between next-gen and legacy technologies to maximize the return of investment
- Leverages catalogue-driven target architecture to enable transformation to full automation for next-gen network, service and user experiences
UPC Delivery Strategy

The UPC platform will be based on and extended from the IBM MDM Collaboration Server solution:

**Infrastructure and Platform**
- Will include the software platform solution, infrastructure and governance process models
- Will support product model design, product lifecycle management, collaborative IDE environment, team and business unit collaboration framework, governance models, version control capability and repository management

**End-to-End PSR Data Models**
- Will support high priority Product-Service-Resource [PSR] models for the initial phase
- Will include the PSR model subtraction, federation, and transformation capability to support legacy or non-SID complaint target systems for their native business operation functions
- The business rules model and design principle will be defined and applied to the meta data model

**Client Systems**
- The client systems with active catalogue will be rolled out gradually, and synchronize the exported product catalogue data entities from UPC via various integration techniques
- Manual process to full automation integration with client systems can achieve the objective of the enterprise catalogue-driven architecture with significant business value
A catalogue-driven solution was achieved through pre-defined product/offer templates and categories, driving repeatability...

Templates enable data-driven (no configurator changes) updates from UPC by re-using existing product rules and associations. Repeatability is key.
Conclusion

As one of the major components of the catalogue-driven architecture, the UPC solution based on the TMF SID data model enabled us to:

- Drive significant benefits for the business by enabling a competitive cost structure and quicker time to market
- Enable simplification of NPI/NTI/PLM business processes
- Achieve enterprise architecture objectives with expected business benefits
- Be well positioned to complete digital transformation to next-gen telecom services and customer experiences