Driving Enterprise API Strategy Using Industry Standard and Models

Biao Hao
Executive Architect
IBM Global Solution Center, Dallas
biaohao@us.ibm.com

Annap Derebail
Executive Architect
Financial Services Sector CTO Office, IBM Global Markets
annap@us.ibm.com
Agenda

- Open Banking and other drivers for enterprise API strategy
- Model driven development for APIs and microservices
- Walkthrough
- Summary
<table>
<thead>
<tr>
<th><strong>Open Banking</strong></th>
<th>Evolution of banking using open APIs, leading to transparency, customer choice and control over personal data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Evolution of banking through apps/app stores to enable ecosystems to create new user interfaces, channel applications, digital banking solutions; extend accessibility to functions, data and products</td>
</tr>
<tr>
<td><strong>Open API</strong></td>
<td>For developers outside of one’s organization, includes standard agreements beyond technology</td>
</tr>
<tr>
<td></td>
<td>Application Programming Interface that should be scalable, re-usable, and secure</td>
</tr>
</tbody>
</table>
Fintechs and non-traditional entrants are impacting the banking industry

<table>
<thead>
<tr>
<th>Non-bank entrants</th>
<th>New banks</th>
<th>New payment models</th>
<th>New lending models</th>
<th>New depository</th>
</tr>
</thead>
<tbody>
<tr>
<td>Banking services</td>
<td>Licensed startups</td>
<td>New entrants offering new types of services in all or part of the payment value chain</td>
<td>New entrants offering credit products, stand-alone or bundled offerings</td>
<td>Aggregation of value on platforms, other than banking products (e.g., savings account)</td>
</tr>
<tr>
<td>(deposit, lending, payments) offered by other industry players</td>
<td>offering banking services with new business models</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Tesco** offers range of banking services. Originally joint venture with RBS now a 100% Tesco owned Bank

**Fidor** Bank built on a cloud ecosystem of financial product providers. API-centric core banking platform

**Apple Pay** has inserted itself into existing payments value chain, providing added value (usability, reduced fraud)

**Alibaba - Zhao Cai Bao** Platform, a 14 bn Yuan marketplace allows customers borrow from investors directly

**Starbucks** mobile payments for use in Starbucks’ own stores Customers loaded $1.6 bn in 4Q14, up 17% YTY

*2/20/17*
Ecosystem to produce rapid innovation and new capabilities

Catering to a range of customer needs beyond merely traditional banking services

Extending a bank’s range of capabilities and scalability to its operations
Regulation as a driver for Open API

PSD2 is a European Banking regulation mandating banks that trade in Europe, provide API access to their payments processing systems to registered Third Party Providers (TPPs), in order to improve payment efficiency, market competition, and enhance security across Europe-wide payments.

TPP = Third Party Payment Provider
XS2A = Access to account
APIs decouple product/service distribution and manufacturing
APIs in bi-modal (hybrid) environment

Hybrid Integration

Systems of Engagement
Business logic

Empowering Digital teams

Core Business Operations

On-Premise

Cloud affinity

API & Event Gateway

API Composition
Events
Data Synchronisation

“Low level” connectivity

API & Event Gateway

API Composition
Events
Data Synchronisation

“Digital” connectivity

Mobile
Partners
API Economy
IoT
SaaS
Offerings

SoR
SoR
SoR
SoR
SoR
SoR

XaaS

8

2/20/17

API & Event Gateway

SoR

NoSQL

SoR

NoSQL

Mobile

Partners

API Economy

IoT

SaaS

Offerings

XaaS

“Low level” connectivity

API Composition
Events
Data Synchronisation

“Digital” connectivity

SoR

SoR

SoR

SoR

SoR

SoR

8

2/20/17
Model driven development for APIs and microservices

- A unique **model driven API development approach** combining IBM Banking Industry Models and API Connect
- A set of **banking APIs as microservices**, that can be quickly deployed to create a **sandbox on Bluemix** for explorations and hackathons
Model driven APIs & microservices – architecture overview
Steps in model driven development and variations

Step 1
Scope APIs using BIAN service domain interactions; Create initial REST Resources using BIAN to REST mapping guidelines

Step 2
Model APIs using RADM, with Resources, and Data Objects (RDOs); Apply extensions targeted for API Connect

Step 3
Generate API specifications (in Swagger format) from the RADM models with extensions for API Connect as target platform

Step 4
Generate API applications using API specifications, with automated implementation; Develop custom code

Step 5
Deploy APIs to create sandbox on Bluemix using scripts to automate services provisioning and deployment

Variations - meet-in-the-middle using models discovered from existing applications

Model discovery

Top-down approach

Github: radm

Swagger files

RADM models

Loopback apps

Loopback apps

Github: sandbox

Provisioning & deploy

API Sandbox on Bluemix

Existing Applications

Exis8ng Applica8ons

RADM models

Swagger files

Loopback apps

Swagger files
BIAN - Business Architecture

BIAN is a global, open, independent and unique community where banks, software providers and system integrators are collaborating to define a common yet exceedingly flexible Business Architecture for the banking industry with the goal of establishing a common language.
Service domains have asset types, whose lifecycles they manage
- Asset types translate into individual and collections resources
- Depending on the functional capabilities provided by the domain, there may be multiple resources associated with a domain
- Additionally, fulfilling certain domain actions can translate into algorithmic resources (e.g., invocation of a process or execution of analytic algorithm)
- The service domain to resource mapping is cataloged in a reference spreadsheet currently
- Intend to automate using an evolving BIAN wireframe tool in the future

https://workbench.bian.org/sl
IBM Banking Industry Models

IBM Industry Models

- **Business Vocabulary, Requirements, Capabilities**
- **Process and Service Analysis Models**
  - Process Design Models
  - Services Design Models
  - Data Warehouse Design Models

IBM Industry Data Model

- Data Model provides a structured data dictionary that defines the business terms and phrases used within the banking industry.

IBM Industry Process Models

- Process Models provide pre-defined analysis-level processes, used to ensure consistency and reuse of processes and activities within the Financial Institution.

IBM Industry Service Models

- Service Models provide the pre-defined analysis and design level structures to enable more consistency and reuse in the creation of SOA and RESTful Services.

- **Key Capabilities**
  - 180 predefined business components
  - 600 predefined process using 3000 predefined activities
  - 40 predefined components with 225 predefined business objects and 350 predefined interfaces
  - 4000 predefined service operations 100 composite services and 5000 predefined services message definitions
- Model consumable APIs using graphical UML modeling in RSA
  - REST applications, resources, data objects, paths, query parameters
  - Simple data structures: do not use polymorphism, and not deeply nested

- Derive from existing model elements in IDM and BOM
  - Traceability: SOA Service Models -> Business Glossaries & Data Models

- Generate API specifications for API/app developers without knowledge of the modeling tool

- Model APIs separately for implementation as microservices

- Capture specifics for targeted API platform in UML profiles
IBM API Connect

- Automated, visual and coding options for creating APIs
- Automated discovery of system of records APIs
- Node.js and Java support for creating microservices
- Integrated enterprise grade clustering, management and security for Node.js and Java
- Lifecycle and governance for APIs, Products and Plans
- Access control over API’s, API Plans and API Products
- Advanced API usage analytics
- Customizable, self service developer portal for publishing APIs
- Policy enforcement, security and control
Models in Loopback represent back-end data sources such as databases or services such as REST/SOAP.

All models are derived from the base model, with methods for adding hooks and validating data.

Each model translates into a predefined REST API with a full set of CRUD operations.

Loopback has a set of built-in models for concepts such as Application, User, and Role.

In a typical application, the developer creates custom models that are specific to the application.
Mapping models – RADM → Swagger → Loopback

- A Loopback public model is created to group the RADM operations, using the tag as the name for the model.
- A RADM Resource Data Object (RDO) is mapped to a Loopback private persistent model.
- A RADM operation can specify Loopback implementation by applying a parameterized template, which is defined in a UML profile.
Walkthrough

- Step 1 – Scope APIs using BIAN
- Step 2 – Model APIs using RADM
- Step 3 – Generate API specifications using RADM
- Step 4 – Generate Loopback applications
- Step 5 – Create API sandbox on Bluemix
Summary

What we provide

- A unique **model driven API development approach** combining IBM Banking Industry Models and API Connect
- A set of **banking APIs as microservices**, that can be quickly deployed to create a **sandbox on Bluemix** for explorations and hackathons

Why this is important

- Brings enterprise rigor (lightweight) to API scoping, design, development, and management
- Pre-built industry models bring rapid development, accelerated implementation and reduced time to value
- Easy to comprehend graphical representation of API design following the best practices
- Better communication and traceability among API lifecycle participants
- Custom UML profile for Loopback extensions enables banks to take advantage of Create capability in API Connect
- Specification/code generation brings development efficiencies
  - API specification with Loopback specific extensions
  - Implementation templates automate code for operations
- Scripts to automate API deployment to user’s account on Bluemix
Acknowledgement

Core team members

- **Annap Derebail** – Co-lead, architecture, BIAN
  - Executive Architect - Financial Services Sector, IBM Global Markets
- **Biao Hao** – Co-lead, architecture, project execution
  - Executive Architect, IBM Global Solution Center, FSM
- **Endemecio Santana** – API deployment on Bluemix
  - IT Specialist, IBM Global Solution Center, FSM
- **Raymond Feng** – Loopback tooling with APIc
  - STSM, API Connect Architect, IBM Cloud
- **Gunjan Pandya** – API development with APIc
  - Software Developer, IBM Cloud
- **Niall Jordan** – API modeling with RADM
  - Banking Process & Services Models, IBM Analytics
- **Shane Morgan** – API specification with RADM
  - Industry Models Development, IBM Analytics

Sponsors

- **Chae An**
  - Vice President, CTO, Financial Services Sector, IBM Global Markets
- **Rod Fleming**
  - Technical Sales Manager & Senior Architect - IBM Global Solution Center, Financial Services Market
- **Juan Carlos Soto**
  - Hybrid Cloud Integration and API Economy, IBM Cloud
- **Andrew Hoyt**
  - Director of Development: Strongloop, API Management, & Gateways, IBM Cloud
- **Savio Rodrigues**
  - Director, StrongLoop & Interaction Services Offering Management, IBM Cloud
- **Elaine Hanley**
  - Offering Manager, Banking, Financial Markets & Insurance Industry Models, IBM Analytics
Notices and disclaimers

Copyright © 2017 by International Business Machines Corporation (IBM). No part of this document may be reproduced or transmitted in any form without written permission from IBM.

U.S. Government Users Restricted Rights — use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM.

Information in these presentations (including information relating to products that have not yet been announced by IBM) has been reviewed for accuracy as of the date of initial publication and could include unintentional technical or typographical errors. IBM shall have no responsibility to update this information. This document is distributed “as is” without any warranty, either express or implied. In no event shall IBM be liable for any damage arising from the use of this information, including but not limited to, loss of data, business interruption, loss of profit or loss of opportunity. IBM products and services are warranted according to the terms and conditions of the agreements under which they are provided.

IBM products are manufactured from new parts or new and used parts. In some cases, a product may not be new and may have been previously installed. Regardless, our warranty terms apply.

Any statements regarding IBM’s future direction, intent or product plans are subject to change or withdrawal without notice.

Performance data contained herein was generally obtained in a controlled, isolated environments. Customer examples are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual performance, cost, savings or other results in other operating environments may vary.

References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business.

Workshops, sessions and associated materials may have been prepared by independent session speakers, and do not necessarily reflect the views of IBM. All materials and discussions are provided for informational purposes only, and are neither intended to, nor shall constitute legal or other guidance or advice to any individual participant or their specific situation.

It is the customer’s responsibility to insure its own compliance with legal requirements and to obtain advice of competent legal counsel as to the identification and interpretation of any relevant laws and regulatory requirements that may affect the customer’s business and any actions the customer may need to take to comply with such laws. IBM does not provide legal advice or represent or warrant that its services or products will ensure that the customer is in compliance with any law.
Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products. IBM does not warrant the quality of any third-party products, or the ability of any such third-party products to interoperate with IBM’s products. IBM expressly disclaims all warranties, expressed or implied, including but not limited to, the implied warranties of merchantability and fitness for a particular purpose.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents, copyrights, trademarks or other intellectual property right.