



## Sales Battle Card Integration Playground

### **ESB Integration Solutions**

**Purpose:** To provide an API layer to legacy applications, for mobile developers, web developers, 3rd party applications and vendors. This market is controlled by large software companies, which provide a SOA/ESB suite, typically composed of multiple products, that requires very high expertise. These products traditionally were designed to connect any application to any application, and expose web services, and not in an API architecture. Later on these products added API endpoint, but the architecture remained heavy, unasync, service intensive, stateless only, and no context-aware.

Market products/companies:

- IBM message broker
- Tibco
- Oracle Fusion
- SoftwareAG WebMethods

Embedding these suites usually takes from 6 months up to 2 years in implementation services, requires a very large investments and usually needs a specialized integration team, project plan with lots of tailoring and adaptation to organizational work processes. These solutions are comprehensive, but suffer from bad Time-to-Market (“TTM”) mainly due to different roles, integration team bottlenecks, manual mappings and limited customization and tailoring.

OpenLegacy offering supports the following scenarios:

- The customer has no ESB in place (usually SMB's)

- The ESB fails in delivering new services in short timeframe, and business demands a shorter TTM solution, and OpenLegacy is a ESB “bypass” solution
- The legacy environment is proprietary or very tailored, and the ESB lacks connector/products to connect to this solution
- Customers pays too much maintenance for connectivity products (MQ, CTG, EntireX), and are looking to cut costs
- Performance sometimes is not good due to the async nature of the ESB suite. ESB works internally as message driven, and less oriented for online transactions, which requires near real time response.
- Lack of mobile/http context in ESB, legacy stateful sessions. ESB is usually agnostic to the underlying application, and treats any transaction on a stateless fashion.
- Customer understands that an ESB architecture is not needed, as a single data source is used (e.g. Mainframe) and they can use a more simplified API solution like OpenLegacy
- OpenLegacy optionally puts the API power in the hand of the legacy team, as the solution doesn't require advanced integration expertise

OpenLegacy becomes a strategic solution in these cases. As the customer usually pays a lot for such integration solutions, a typical license would be \$250K-\$3M yearly. Integrators are keen for such solutions and willing to pay in advance, just for the opportunity to get the 3-6X multiplier for services.

### **Co-Existence**

Co-Existence is a subset market for integration solutions. It is driven from a business demand and opportunities. This scenario is typically relevant when the enterprise plans to migrate off the legacy system and need the new system to run in parallel to the old system while maintaining data in both systems in parallel.

Competition is similar as in ESB integration solution. OpenLegacy offers in this case the following benefits, in addition to the previous ESB benefits:

- Fast TTM to produce integration end points between the old and new systems
- Ability to interact with screens which is the only legacy integration in many cases
- Ability to perform ETL (batch updates) from/to the legacy via the business process and not just the data

### **Screen Modernization Solutions**

**Purpose:** To provide a new UI to green screens, typically for web, mobile and tablet usage. This is a mature market with commodity products. Products in this market usually have screens only offerings.

The Mainframe offers are led by:

- SoftwareAG Applinx
- SoftwareAG Jacada
- IBM HATS
- Attachmate / Microfocus
- Rocket Software

Price range is usually \$100K-\$300K perpetual licensing. Products are usually 12-16 years old, with



## Sales Battle Card Integration Playground

old or proprietary web technologies, limited flexibility and no API architecture. Some provides a WYSIWYG editor. Some provide the ability to expose web services from green screens. Most of them require manual fields mapping. The AS/400 has additional products in lower price range (\$30K-\$60K).

OpenLegacy differs in this market by providing:

- API based architecture
- Cutting edge technologies and standard based
- Cloud and on premise deployment options
- All-in-one solution for legacy presentation/business logic/data layers
- Built-in API management capabilities

In some cases, customer already has existing modernization solution and looking for a more modern technology and faster TTM capabilities.

### Home Grown Integration Solutions

Many organizations invested organically in home grown infrastructure and practices to perform integration. This is usually implemented using IT architects, and the home grown solution is adapted but usually suffers from bad TTM, high maintenance, and “reinvent the wheel” approach.

Typical use cases:

- Usage of low level RPC API for AS/400, and writing lots of code on top of it
- Using emulation low level API for interacting with screens for exposing legacy processes. This approach breaks often and requires dealing with very low-level screen scraping coding.

- Lots of Java/.NET coding for exposing web service/mobile/web from more common technologies such as Web services, databases and SAP

OpenLegacy offering in this case:

- Much faster TTM
- Reduced maintenance
- Putting standardization in place
- Simplifying IT entry to new common technologies such as REST API and AngularJS

### Do Nothing Competition

Usually SMB’s which didn’t rush to adopt integration solutions. Most of the business logic is on the legacy application (typically AS/400, can be SAP later on). Some peripheral web/mobile application, but disconnected from the legacy applications. Nightly/hourly/etc. synced are performed, using files/data.

OpenLegacy provides value to such customers:

- Enabling by opening up much more legacy and much faster application to web/mobile
- Don’t replicate the Business Logic to new applications, which also slows down development
- Getting more real-time information

### OEM / ISV

In this scenario, which is related to business development, we cooperate with the legacy software vendor, aiming for a bundled solution (OEM), in which OpenLegacy is part of the vendor offering and can be up-sale/upgrade with the vendor product

releases. We work with the vendor to provide OpenLegacy predefined APIs for the vendor’s software: Screens, programs and/or data, as a ready to use API, web or services, with/without customer customization capabilities.

OpenLegacy offering to the vendors:

- Ability to open up the legacy software (e.g. Mainframe Core banking, AS/400 ERP) for API usage, which typically is a demand from it’s customers
- Let the vendor focus on the business logic functionality and not on integration and API capabilities
- Enable the vendor to maintain happy customers, and avoid losing business to more open alternatives

### API Management / New ESBs

The ESB market has evolved with new players providing ESB capabilities. These products rarely deal with legacy platforms and typically provide API to web services (from ESB), REST APIs and database connectivity (SQL/NoSQL).

None of these API management products provide integration to Mainframe. Some provide to AS/400 and SAP but only through 3rd party offerings.

OpenLegacy offers in this scenario:

- A full software stack starting from the legacy system to more modern platforms (DB, SAP, WS, etc.), up to API management capabilities
- Deep caching, deep analytics on the base assets (e.g. program, screens)

## Sales Battle Card Integration Playground

- Distributed transaction from legacy, on-prem, cloud
- High flexibility and customization
- Faster TTM
- Full end-to-end solution including front end web/mobile generation

### Platform Vendor Solutions

In this category, OpenLegacy competes with the vendor self-solutions (IBM, SAP, & Oracle). The motivation typically for the customer for NOT choosing the vendor solution:

- Requires platform upgrade, and lack of support with new solutions on older platform versions
- Expensive license fees for additional software stack on the Platform
- Requires internal platform experts (Mainframe) which is less available, than admins for Open/Java based environments
- Limited flexibility in the platform solution
- High learning curve
- Lack of orchestration between multiples programs
- Lack of orchestration between different platforms
- Lack of expertise on the solution, even by the vendors themselves
- Low traction to developer due to non standard solution

OpenLegacy provide a flexible, standard with fast TTM vs. the platform vendor solutions. In this category, it's possible to find solution like:

### IBM z/OS Connect

Information below is based on comparison from <http://www-03.ibm.com/software/products/en/zos-connect-enterprise-edition>

First of all, this is a first step from IBM in the right direction. It is expected that non-ESB integration will be available from more and more vendors. However, everything we say still applies:

1. z/OS Connect EE is currently in open Beta status, not available to customers
2. In order to use the z/OS Connect EE to expose CICS transactions as secure APIs (most common scenario) you will need to have the following IBM products:
  - WebSphere Application Server
  - RDz
  - z/OS Connect EE server
  - WebSphere z/OS Optimized Local Adapters (WOLA)
  - IBM API Management
  - IBM Datapower
3. Consider that even if you don't have to buy all the above products, even if IBM will give you everything for free, still you will need expertise and experts in all of those different technologies, which most customers do not currently use (IBM has been trying to push RDZ and Eclipse based COBOL programming for years now, without much success).
4. z/OS Connect EE offers no flexibility at all - APIs are as-is. No customizations available. **This would have been a deal-breaker for most of our current customers and engagements.**

5. Only deployment option for the APIs is the mainframe itself - no scalability at all
6. Only deals with CICS/IMS mainframe assets, no solution for other mainframe assets such 3270, DB2 or SOAP services from the mainframe, certainly no solutions for non-mainframe
7. It does not provide a way to generate Java strong type API as OpenLegacy does, for Java integration and customization
8. It does not provide caching capabilities to reduce Mainframe MIPS
9. It exposes the Mainframe via HTTP to developers or at worst, end users to REST, without further protection which needs additional products (Datapower)

Basically this is a nicer way of doing something that was available on the CICS platform for 2-3 years now. Better UI and integration with other IBM products but no big change in architecture or concept. For real-life cases (as opposed to demos) an ESB will still be needed for orchestrations, flows and service.