



Tapash Bishoi

Sr. Integration Manager, Architect
Crane Worldwide Logistics

Integration Types

Enterprise architecture helps to define the standards for managing, acquiring, designing, and configuring technology and services that affect the entire organization. Based on Crane's vision of enabling API led, real time and messaging based integrations , this initiative will focus on following integration architecture types.

Batch Integration

Connect data to applications through batch or scheduled date /time . Empowers enterprise to leverage their best capabilities in delivering applications and projects in control manner based on enterprise application processing methodologies .

Real-Time Integrations

Process and transfer data as soon as it's collected. Ensures faster availability of end-to-end data, better performance of process execution and derive better business decisions

Messaging based Integrations

Transfer of messages between connected applications and systems where producers and consumers are de-coupled. Ensures delivery of messages even if subscriber is down with no data loss



Integration Design Principles

A robust set of integration guiding principles is imperative to achieve Integration standardization, consistency and governance

Loosely coupled integration

Integration design to break complex process into smaller independent sub processes considering fact that any future change will not impact other developed interfaces

Reliable and Resilient Integrations

Design Integrations to ensure Data integrity, Fault tolerance and recoverability across applications to support the service level agreements.

Common API Library or Services Repository

Maintain common services as a separate library to promote maximum reusability and standardization. E.g., services for Error Management and Exception Handling framework, Dev Ops, Audit logging, Alerting

Integration Patterns to Govern Design

Stick to pattern-based Integrations. Follow recommended design templates and guidelines for Real Time, Non-Real Time, Batch and Reporting Integrations

Maximize OOTB connectivity usage

Configure out-of-the box (OOTB) adapters for connectivity with SaaS and applications. Provides prebuilt connection configuration options, saves development efforts for custom connectors, easy monitoring and zero maintenance efforts

Reusable Technical Services

Interface design capable of being used again or repeatedly. Modularity for easier troubleshooting, Reuse of code through inheritance

Security by Design

Where necessary each layer of integration should enact securely, and each integration flow (service) must identify the resources it requires and the valid credentials for it

Abstraction of configuration, rules, etc., from the Core Platform

Any customization, configuration, business rules should be written outside of the main Core Platform to support any changes, upgrades, etc.

Process Orchestration & Process Co-ord

Refers to the practice of unifying individual tasks into end-to-end processes & process synchronization for concurrent execution

API for Partners and Businesses

The architecture should expose API (at experience level) which can be used in the Partner ecosystems or by external applications via secure access points

Error Handling & Reprocessing

Interface will handle all runtime exceptions and have reprocessing mechanism of failed records

Business and IT Monitoring

The design of an integration should support the tracking of completion of a business transaction and support systems level alerts & monitoring

Real-Time Integration Design Principles

Real-Time integration guiding principles serve as foundation for robust interface development and helps to achieve objectives of enterprise architecture

Scalability and Flexibility

Scale to support any increase in the load transaction/ volume, and strategic long-term extensions in functional scope and application architecture. Accommodate changes to the application landscape, functional requirements, and/or technical requirements.

Integration based Interoperability

Design integrations which can serve multiple boundary applications via a common interface.

Promote Data Velocity

Real Time Integrations interfaces will make data available end to end at higher speed. Faster availability of end-to-end data, better performance of process execution

Real time – observability

Establish monitoring and governance of all integrations as close to real time as possible to provide end-to-end traceability from integration source to destination

Application dependencies for Real-Time Design

Consider boundary application capabilities and limitations to design an Integration as Real-Time or otherwise

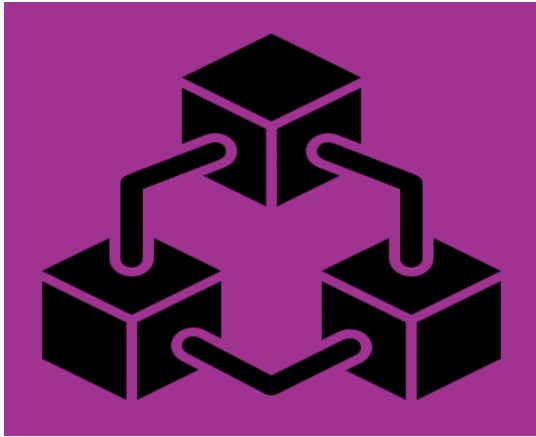
SLA driven Real-Time design

Design real time integrations with API based or Queue based design patterns depending on volume and SLA consideration for data transfer



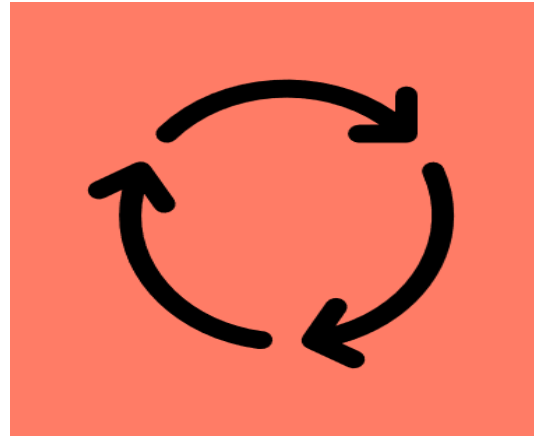
Technology stack – EDI processing framework

Below are the technology stack we have used to build our common framework , router and EDI dashboard



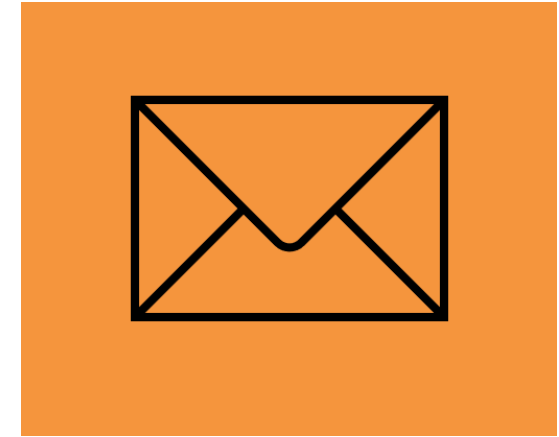
Integrations, EDI/Trading partner

Connect data to applications through reusable Trading partner Integrations.



BOOMI APIM

APIM to built self discovers Apis to be used across enterprise.



BOOMI Event Stream

Transfer of messages between connected applications and systems where producers and consumers are de-coupled. Ensures delivery of messages even if subscriber is down with no data loss



Thank You