workato

WHITE PAPER

The Cost of Standing Still

Why Legacy ESB Slows Enterprise Growth

RESHMA RADIA, ENTERPRISE ARCHITECT

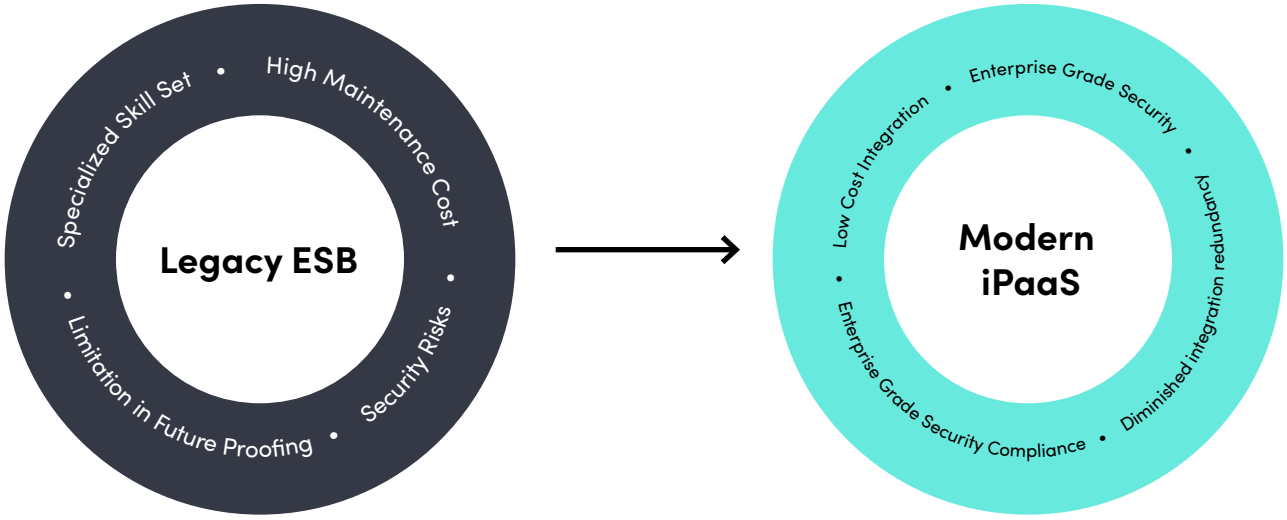
Introduction

In the new modern world, fast-paced, agile technology can make or break an organization's need to keep up with growing demands and respond to market changes to remain competitive. Legacy systems, now considered outdated, pose a challenge to enterprises that need to grow at scale and modernize. Many enterprises have the need and want to modernize their technology but hesitate due to being heavily intertwined with old systems that fulfill critical business processes. Ripping and replacing a system like a mainframe is very difficult and very few enterprises in the world have done it successfully. Ultimately, the solution is trying to co-exist with the legacy system whilst still modernizing the greater solution without disrupting key business processes.

In today's times, startups and organizations can access and implement the latest and greatest technology to integrate their systems. In this paper, we will talk about how these systems were previously integrated and the challenges posed by older integration patterns and platforms as well as how enterprises should take the leap into modernizing their solutions. The organizations that will succeed and be the "leaders" will be defined by those that have the best "interconnections" between their systems.

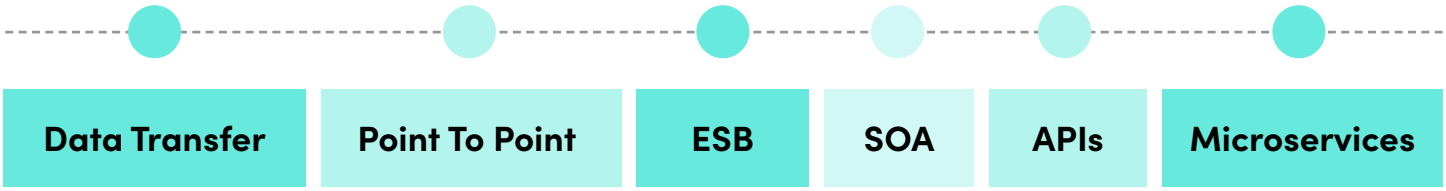
The Legacy Integration Solution

Decades back, new integration patterns and tools emerged to modernize the legacy system architecture such as Mainframe, COBOL, DB2, and other systems built on a single application server. This section covers the evolution of integration patterns for the legacy integration solution and some of the key platforms that were used to connect legacy applications to target systems and the challenges that have been present with this approach.



Evolution of Integration Patterns

Through time, enterprise integration evolved from data transfer between two systems to the Enterprise Service Bus and Service Oriented Architecture to connect legacy applications with target systems. It is important to understand the reason behind the evolution of these patterns.

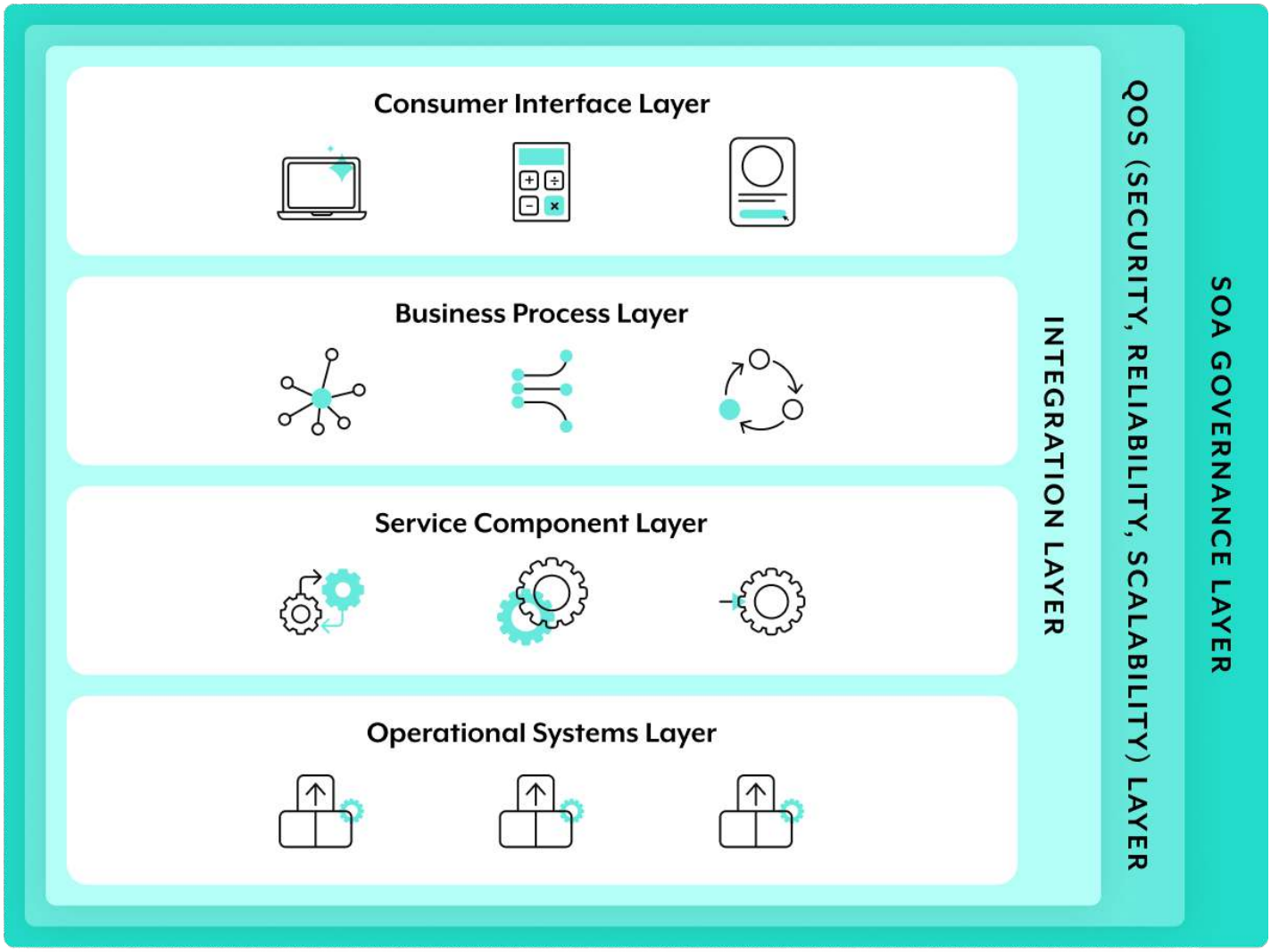


Integration patterns have evolved over time—from data transfer in the 1960s to microservices, which gained public traction in the 2010s.

Data transfer between two applications was the first step in how systems were integrated. In the old days, this was done through punched cards or magnetic tapes, bringing them to the other systems to upload. This was a simple form of file-based integration, where data was shared via files transferred between systems. It was manual, cumbersome, and error-prone. Not only did it hinder the process of integrating data from more than one system, but data sharing was restricted to only internal applications. It laid the foundation for the need for more sophisticated methods of integration.

This is where point-to-point integration, connecting data from one system to another, took off. Each interface was defined as connecting System 1 to System 2. If there was a third system involved a second integration would be needed and so on. It solved the issue of being able to share data between different applications. Whilst it solved issues such as being able to share and transform data in realtime, for larger enterprises that had a plethora of systems, the number of integrations grew exponentially. Furthermore, there was no central visibility to manage the growing number of integrations, and error handling and recovery needed to be built into every single interface. As market needs grew for larger enterprises, point-to-point integration needed a makeover.

That is where the Enterprise Service Bus (ESB) and Service Oriented Architecture (SOA) came into play. Both are two sides of the same coin. While the ESB solved the technical issues, SOA orchestrated business processes. The simplest option was to wrap the current legacy system with a middleware (ESB) that can offer the legacy system interface through a Web service or connector. In this way, legacy functionality is “wrapped” with a service layer and “plugged” into an SOA environment which provides modularized functional building blocks for business functions such as “Payments” or “Shipment’. This solved the issue of interoperability, messaging disparities and data transformations, centralized error handling and logging as well as governance and security.



Service Oriented Architecture and Enterprise Service Bus are two sides of the same coin

However, the high cost, maintenance issues, and scalability issues still posed a challenge and yet another upgrade was needed in the world of Enterprise Architecture. This is where modern-day integration patterns came into play.

Legacy Integration Platforms

As the ESB rose in popularity as the means to connect legacy applications, several integration software vendors emerged to provide tools to make the journey easier with out-of-the-box functionality and drag-and-drop features allowing service providers and service consumers to talk to each other via the ESB.

Two of the leading vendors in this space at the time were:

WebMethods:

Founded in 1996, started as an Enterprise Service Bus specializing in WebServices. Through time webMethods evolved into a hybrid, cloud-optimized iPaas platform consisting of:

- webMethods.io (Integration Server)
- webMethods API Management Gateway
- webMethods EntireX
- webMethods Business Rules
- webMethods Trading Networks

Webmethods was acquired by Software AG in 2007 and most recently by IBM (2023).

Tibco[1]:

Founded in 1997, started as an Enterprise Service Bus. Eventually, Tibco evolved through time into a hybrid, **cloud-optimized software** platform consisting of components such as:

- Tibco Cloud Integration
- Tibco Cloud API Management
- Tibco Cloud EBX (MDM Solution)
- Tibco Cloud Messaging

In 2014, TIBCO was acquired by a private equity firm, an environment often criticized for stifling innovation.

1 Teknekron Software Systems, Inc. was founded in 1986 by Vivek Ranadivé. Reuters acquired the company in March 1994 for $125 million. In 1996, the company was renamed TIBCO Inc., and in January 1997, it was restructured to form TIBCO Software Inc., focusing on markets beyond financial services.

Challenges of Legacy Integration Platforms

When integrating solutions in large enterprises, three fundamental use cases come into play:

- **Data Consistency:** Systems such as databases and systems of records need to be synched so that data is consistent between the systems.
- **Multistep Process:** Applications independent of each other are integrated to facilitate an end-to-end process
- **Composition:** New applications are built by a combination of existing applications, data, and functions

Whilst traditional integration platforms are powerful in their own right and have a good set of features, customers are choosing to migrate away from them due to rising challenges such as:

- **Limitations in Types of Use Cases:** Traditional Integration products can typically **support only one of these use cases** based on their core functionality. When integrating with a modern iPaaS solution, all 3 use cases can be addressed simultaneously without having to purchase multiple platforms for different use cases across the enterprise.
- **Specialized Skill Set:** Heavier integration platforms require a steeper learning curve for developers or a highly specialized skill set. This creates more time and cost for the organization to train current employees or find resources with the required skills for that platform.
- **High-Cost Integration:** Many integration platforms have functionality or add-ons to be able to handle large-scale use cases. Additional memory to scale or infrastructure as well as additional costs for add-on tools can have a significant impact on the total cost of ownership of the tool.
- **Maintenance:** High maintenance costs for upgrades and maintaining the current architecture to scale to modern-day needs can impact both total cost and the amount of resources needed to continue to provide value to the business. Hybrid, cloud-optimized solutions require significant time and effort to upgrade to major releases adding the need for regression testing and/or re-writing current integrations.
- **Limitations in Future Proofing:** Hybrid Integration platforms which were built to be best suited for on-premise environments may not have upgraded all their components to cloud infrastructure. Additionally, newer capabilities such as automation capabilities or AI-driven capabilities may be limited in these platforms.
- **Enterprise Grade Security Risks:** Legacy systems often lack modern protections like strong authentication and encryption, making them vulnerable to threats. Integrating these systems with newer technologies increases the risk of breaches and requires complex, resource-intensive solutions to ensure safety.

Modern-Day Integration Patterns

As market trends changed, the need for faster-paced, agile architecture for large enterprises is growing due to the demands of modern application development cycles that force organizations to respond quickly to evolving conditions, new opportunities, and ever-changing customer preferences. New applications must be quickly developed and deployed, and frequently updated. The following modern-day architecture patterns provide a fast-paced response to real-time events:

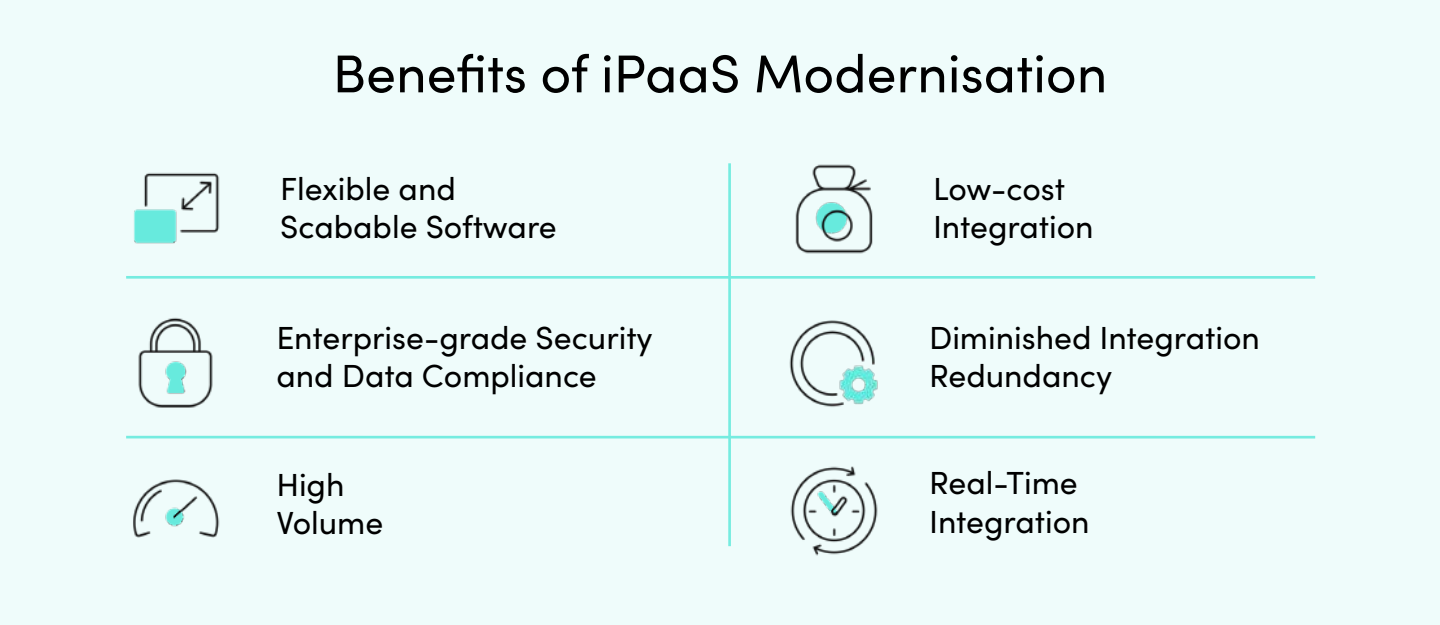
- **Event-Driven Architecture**: Scalable architecture pattern in which decoupled applications can asynchronously publish and subscribe to events via an event broker (modern messaging-oriented-middleware). Event-driven architecture has three key components:
    - **Event Producers:** Publishes an event to the router which triggers the business process. The Event Producer only knows an event has occurred but does not know the outcome of the event.
    - **Event Routers:** The middleman that manages, controls, and routes events from specific event sources to specific event targets. It also plays a vital role in transforming the event format and applying rules to filter and match the events.
    - **Event Consumers**: Listens for specific messages from Event routers and acts on them. There can be multiple consumers for a single event. Event consumers can be downstream applications such as Billing Systems, Microservices, or event Notification Engines.

Producer services and consumer services are decoupled, which allows them to be scaled, updated, and deployed independently.

- **Microservices:** Consists of small modular services that represent a specific task or business function. Each service runs independently and a series of services are tied together via communication interfaces such as APIs. Each service can be isolated, rebuilt, tested, redeployed, and managed independently.
- **Data Hub:** Hybrid pattern of a data lake and a data warehouse. It stores both raw and structured data in a modular and flexible architecture, using different layers and zones to separate the data by quality, purpose, and access. A data hub enables you to ingest, transform, and distribute data across multiple systems and applications, using a metadata-driven and event-based approach.

Benefits of iPaaS Modernization

The first question any enterprise organization asks is *"Why should I spend the money to modernize? What will I get out of it?"* The benefits of modernizing far outweigh the risks.



The key benefits include:

Flexible and Scalable software: When it comes to integration, adding new solutions and services to your network with a cloud-native iPaaS solution is powerful, provides speed and scalability, and eliminates the resourcing, costs, infrastructure, and time commitments required. iPaaS solutions offer you greater flexibility to scale up or down depending on changing business demands.

Low-cost Integration: Older ESB and iPaas solutions require specialized skills and high costs for upgrading to major releases. Organizations spend millions of dollars a year to upgrade, re-architect, and regression test for upgrading their solutions to the latest version of the platform. With a modern, cloud-native solution, upgrades occur automatically with zero downtime, relieving organizations from the high costs of software upgrades and trying to find specialized resources to maintain.

Enterprise Grade Security and Data Compliance: Legacy systems present significant security challenges due to outdated technologies, lack of vendor support, and incompatibility with modern defenses. They are often vulnerable to unpatched exploits, weak authentication protocols, and limited access controls, making them prime targets for attackers. Additionally, legacy systems lack essential security features like encryption, robust logging, and integration with modern monitoring tools. Flat network architectures and insecure protocols further increase exposure to threats, while hardcoded credentials and inadequate data protection exacerbate risks. Compounded by resource constraints and a shrinking pool of expertise, these systems become increasingly difficult to secure.

Modern iPaaS solutions have advanced security and governance features. Modern-day iPaaS platforms include features that satisfy corporate and regulatory compliance standards, Role-Based Access Control (RBAC), and support for the latest enterprise authentication technology.

Diminished integration redundancy: iPaaS connections are efficient for multi-tenant applications because they reduce integration redundancies while at the same time offering rapid scalability for onboarding new tenants.

High Volume, Real-time integration: Some iPaaS technologies are native to the cloud and other cloud-optimized; hence, the integration solution is uniquely suited to handle the high-volume, real-time data integrations, and near-real-time processing demands of a modern cloud enterprise IT architecture.

Workato- The Modern Day iPaaS Solution

When talking about modern-day integration patterns like the above, an iPaaS solution that fulfills these needs but also future-proofing solutions and business processes is required. Workato has been recognized as a Leader with the furthest in vision placement in the Gartner® Magic Quadrant™ for Integration Platform as a Service (iPaaS), Worldwide. Named a Leader for seven consecutive years for our Ability to Execute and our Completeness of Vision.

Workato features a modern-day approach to integrating old systems with a new fast-paced agile architecture. Whilst many vendors are trying to upgrade their current solutions to match new modern-day needs, the elephant in the room still needs to be addressed: ***How do we move to a new way of integrating whilst being restrained by old legacy systems with limited integration capabilities and patterns that are difficult to unpick?***

The Path Forward

Modernizing your integration approach is no longer optional—it's the foundation for agility and growth in today's enterprise IT landscape. With cloud-native, decoupled iPaaS solutions like Workato, you can break free from the limits of legacy and realize real business agility without disruption. Talk to one of our integration experts today, and stay tuned—there's much more to explore in the future of enterprise integration.

Take the first steps

Learn How Modern iPaaS is Evolving to Enterprise Orchestration

Why the path forward is Cloud-Native vs. Cloud-Optimized

The Role of APIs in Enterprise Automation

Download the 2025 iPaaS Buying Guide

Start planning for your future

Talk to our experts or see a live demo to discover how Workato enables fast pivots—and real competitive advantage.

workato

Schedule a Demo