onebridge
a marlabs company

What to do when integration tools are too expensive or not available

Client

A data and artificial intelligence company based in Indianapolis, IN

Company Size

400+ employees

Location

US

Onebridge, a Marlabs company, needed to integrate several internal applications, but traditional integration platforms were too expensive, inflexible, and unsupported by their architecture. The internal IT team had strong development skills but lacked access to affordable and compatible integration tools that could support scalable, long-term growth.

Our team designed and implemented a lightweight, modular integration framework using the .NET platform. This custom solution replicated key features of enterprise tools, allowed for future expansion, and could be managed in-house. The result was a cost-effective and modern integration capability that eliminated licensing constraints and empowered internal teams.

Enterprise Architecture

Integration Strategy

Custom Application Development

API Development

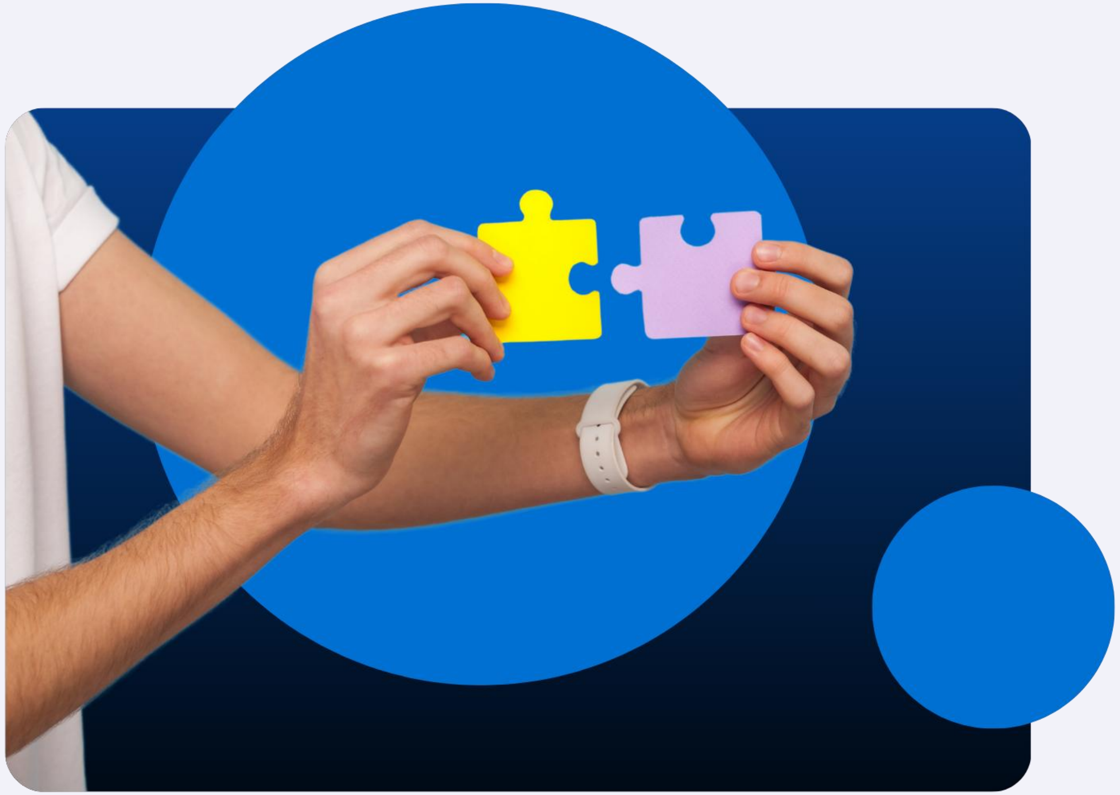
The Challenge: Solving integration without expensive tools

Objective: Deliver a flexible integration framework that internal teams could maintain and scale.

Existing Issues: Commercial integration tools were too expensive, incompatible with legacy systems, and unsupported by internal architecture.

Solution Needed: A custom-built integration solution that avoided licensing costs and enabled long-term modernization.

Outcome: A sustainable and scalable .NET-based framework that connected critical systems without relying on third-party platforms.



The organization saw the need to modernize and unify data across multiple legacy systems. However, they didn't want to purchase expensive out-of-the-box solutions with traditional integration platforms that couldn't meet all their needs.

The Solution: A custom integration framework built to last

To address the limitations of commercial tools, we created a tailored solution using the Microsoft .NET framework. This modular architecture replicated the core capabilities of expensive integration platforms while remaining lightweight and adaptable to future changes. The result was a powerful and cost-conscious integration strategy that fit the organization’s environment and skill set.

Phase 1: Architecture Assessment & Planning

The team evaluated existing systems and architecture to identify integration points and technical constraints.

Workstreams:

- System audit
- Integration mapping
- Constraint identification

Phase 2: Framework Design & Development

We built a custom .NET integration framework that emulated key features of traditional tools.

Workstreams:

- Modular architecture design
- API structure creation
- Exception handling setup

Phase 3: Validation & Deployment

The team rigorously tested and deployed the framework to ensure reliability and performance.

Workstreams:

- Functionality testing
- Load testing
- Production deployment

Phase 4: Knowledge Transfer & Support Enablement

To ensure long-term maintainability and independence, the team created documentation and trained staff.

Workstreams:

- Documentation development
- Team training
- Support strategy creation

Services and Technologies Used:

Services:

- Enterprise Architecture
- Integration Strategy
- Custom Application Development
- API Development

Technologies:

- .NET Framework
- RESTful APIs
- Microsoft SQL Server
- JSON & XML
- Legacy Application Platforms

The Results: Impact on the client organization

The engagement resulted in a cost-effective integration solution that allowed us to consolidate systems without the constraints of third-party tools. The framework offered a future-ready architecture that could evolve alongside business needs, and internal teams gained the knowledge and tools to maintain it independently. This approach proved that integration can be powerful and scalable without breaking the budget.

Lowered Integration Costs: Avoided licensing fees by replacing commercial tools with custom, in-house solutions.

Improved Internal Enablement: Equipped internal teams with the training and documentation needed for ongoing maintenance and self-service.

Increased Scalability: Built a modular system that could grow and adapt to future integration needs, architectural upgrades, and digital transformation.

Reduced Vendor Dependence: Eliminated reliance on external integration platforms, enhancing system control.

Streamlined Data Flow: Enabled consistent and reliable data movement across legacy and modern platforms.