

Platform integration

What it is, why it matters, how to do it



Index

Index	1
Introduction	2
What is platform integration?	3
Why platform integration matters	5
The benefits of platform integration	7
How to integrate your platforms	8
How to prepare for platform integration	12
How to implement platform integration	17
Platform integration best practices	19
How to measure success and ROI	21
Key challenges and barriers	23
We are here to help	25

Introduction

The technology landscape in most large enterprises is a vast, complex beast.

It's analogous to a sprawling city that has grown organically over centuries. Initially, there were only simple structures and roads. But, with each passing decade, new layers of buildings, roads, and bridges are added.

These new layers were often added to meet short-term needs, without considering how they meshed with the wider cityscape.

As a result, the modern-day cityscape is a strange mix of ancient buildings, narrow medieval streets, shiny skyscrapers, and tangled spaghetti junctions, each from different eras, serving different purposes and built with different standards.

When these different elements aren't integrated, traffic becomes chaotic, navigation is confusing and the infrastructure is inefficient.

This is exactly how legacy technologies in a business can become an unharmonious and inefficient mix of disparate systems.

This lack of integration between systems has a very high cost both in terms of the financial cost of overcoming the limits and the business cost of sluggish innovation.

Platform integration in this analogy is like a major urban planning initiative. It aims to integrate the various parts of the city—aligning old roads with new ones, ensuring different areas of the city are well-connected, and modernizing outdated structures—to create a cohesive, efficient, and modern urban environment.

The goal is to bring harmony to the disparate systems, making them more efficient, scalable, and suited to meet current and future needs.

In this playbook, we will outline everything you need to know to successfully approach a major platform integration project, helping to turn these pain points into business value.

This is intended for IT and business decision-makers who know they need to integrate but want to make sure that they maximize the business value of their platform integration program by taking the right approach.

What is platform integration?

Platform integration is the process of connecting different software and technology platforms to work together as a cohesive system, including applications, data, clouds, APIs, processes, and devices.

The work of integrating vast, disparate systems is complex and can be viewed through various key concepts, each of which highlights a different facet of the transformation.

Use cases



API platform

Merge and expose data from different sources, such as CRMs, ERPs, supply chain management and multiple technology platforms to create a unified view or consolidated platform for executing business functions.



Application integration

Integrate 3rd party tools into your existing applications such as mail services, payment services or CRM.



Process optimization

Focuses on aligning business processes and optimising them across various platforms. For example, automating the flow of information from a sales platform to an inventory management system.



Exposure to marketplaces

As businesses increasingly operate in interconnected ecosystems, integrating platforms with external partners, suppliers, and customers becomes important. This can involve exposing your platform to 3rd party marketplaces or platforms as well as consuming services from them.



Migrating to the cloud

Many enterprises are moving to cloud-based solutions. Integrating these cloud platforms with existing on-premises systems is a crucial part of the integration process. Ensuring that this is done in a cost effective approach is essential.



Pre & post incident support

Ensuring that integrated platforms adhere to regulatory requirements and maintain high standards of data security and privacy.



Product market fit

Achieving product market fit requires you to iteratively test with customers to find their killer features and solve for their pain points.

Why platform integration matters

Platform integration is a valuable opportunity to turn your technology estate from a set of disparate and siloed systems that each churn out a particular product or service into a deeply interconnected ecosystem of powerful capabilities and building blocks upon which innovative new products and services can be more easily built.

The deepest benefit of platform integration is much greater than the sum of the various technical advantages. Helping one system talk to another to save a bit of time and money is just the tip of the iceberg.

By connecting different products, tools, teams, data, and so on, it makes your technology capabilities highly accessible to your people.

This democratizes the power of technology, taking it out of the ivory tower and putting it in the hands of non-specialists.

Rather than being the only domain of specialist engineers or data scientists, business analysts and product owners can start leveraging capabilities like analytics or AI on their own.

Returning to our city analogy, we want to not only integrate the tangled mess of different buildings and roads...but to democratize the capacity to create new flows of goods and services within the city.

In this way, the city will be able to flourish in whatever direction it needs, enabling new capabilities and possibilities that can't even be imagined yet! And democratizing these to the ordinary citizens, not just the urban planning gurus.

It's about laying the foundation for the city of the future!

By failing to integrate your disparate systems and bring them in line with modern standards, you are setting yourself up with a serious competitive disadvantage.

The technical and business downside of a lack of integration are significant, as broken down below:



Technical downside

- **Data limitations:**

inefficiencies, manual work and a fragmented understanding of business operations limits business-critical innovation, analytics, reporting, feedback, and customer-facing functions.

- **Inefficient manual work:**

your people need to spend their time devising workarounds, manually tracking down data, and badger colleagues to help them in the face of integration challenges.

- **Scalability:**

non-integrated systems may struggle to handle increased loads or complexity, limiting the growth of your business.

- **Compliance and security risks:**

different systems may have varying levels of security and compliance standards that make it more difficult to maintain an up-to-date security posture across the business.

- **Supply chain:**

[85% of companies](#) are losing money due to integration issues related to their supply chains, emphasizing the critical role of efficient integration in supply chain management.



Commercial downside

- **Spiraling overheads:**

managing multiple standalone systems (and investing in short-term workarounds) can be significant. The average enterprise spends around [\\$3.5 million per year](#) on integration-related IT labor costs.

- **Collaboration and communication:**

collaboration between different departments or teams can be limited due to the lack of shared information and tools.

- **Innovation:**

non-integrated systems can slow down a company's ability to innovate and adapt to market changes, slowing down digital transformation initiatives for 87% of organizations, according to [Mulesoft](#).

- **Customer experience:**

it becomes challenging to develop a 360° view of the customer or a single source of truth for customer data. This can lead to inconsistent customer experiences, as different customer data are used across customer touchpoints and not shared.

The benefits of platform integration

By integrating your platforms and connecting your systems you open the door to a range of benefits, both technical and commercially-orientated.



Technical benefits

- **Streamline IT complexity:**
reduce the complexity of the IT landscape, making it easier to manage and maintain. This helps in reducing IT workload and improving system performance.
- **Tech stack consolidation:**
eliminate redundant technologies, optimize the use of software and hardware resources, and reduce licensing and maintenance costs.
- **Data synchronization:**
data across different systems is in line with governance standards (format, schema, etc.) and up-to-date, eliminating discrepancies and silos. This leads to more accurate, trustworthy, and available data for operational and analytical purposes.
- **Scalability:**
integrated systems are better equipped to handle increased loads or expand functionalities, supporting business growth without the need for extensive restructuring of the IT infrastructure.
- **Build an ecosystem of capabilities:**
rather than IT being the gatekeepers of specific services that are delivered to the business, they become the curators of a suite of business capabilities, which are available for the business to use on demand.



Commercial benefits

- **Operational efficiency:**
streamline workflows, reduce manual processes, and enhance data flow across various systems, leading to significant improvements in efficiency and productivity. The average organization generates 27% of its revenue from APIs, showcasing the economic benefits of integration.
- **Cost savings:**
by optimizing processes and reducing manual interventions, integration leads to significant cost savings from reduced labor costs, lower error rates, and more efficient use of resources.
- **Accelerate innovation:**
more effective data sharing, communication, and democratization of technology facilitate better decision-making, drive innovation, and allow businesses to adapt more rapidly to market changes and new opportunities.
- **Improved customer experience:**
seamless integration of customer data, touchpoints, and interfaces makes for a vastly superior customer experience that can be highly personalized and responsive to their needs.

How to integrate your platforms

Platform integration is a very complex process that involves countless applications, data sets, people, teams, departments, tools, processes, and so on from every corner of your business.

As such, there are many different practices, methods, and approaches that can be applied. This section aims to cover some of the main approaches, best practices, and most common challenges, helping you determine which is best for your unique business context.

Approaches to platform integration

Given the complexity of enterprise technology and the unique context of each business, there are many different ways that integration can be achieved, depending on the objectives and the depth of integration required.

This section will outline the three major approaches to platform integration.

- Integration platform as a service (iPaaS)
- Custom integration (ESB, middleware, SOA)
- New platform (building a new, overarching platform)



Integration platform-as-a-service (iPaaS)

iPaaS is a cloud-based platform that offers a suite of tools and services to connect various software applications deployed in different environments (cloud, on-premises, hybrid). It allows businesses to integrate systems and manage data flows without the need for on-premises hardware.

Typically, an iPaaS tool provides pre-built connectors, business rules, maps, and transformations that facilitate the development of applications and orchestrate integration flows.

Advantages

- **Cost-effective:**

generally less expensive than building a new platform.

- **Flexibility and scalability:**

easier to add or remove integrations as business needs change.

- **Speed of implementation:**

iPaaS solutions offer quicker deployment compared to other approaches.

Disadvantages:

- **Potential for integration limits:**

some iPaaS solutions may have limitations on what can be integrated.

- **Vendor dependency:**

there is reliance on the vendor for support, updates, and security.

- **Complexity in coordination:**

manual API integration requires managing multiple integrations, which can become complex and challenging to maintain.



Custom integration

Custom integration involves directly connecting specific applications and systems through bespoke interfaces and data exchange mechanisms.

This involves determining which systems, data and applications need to be integrated and developing new, custom APIs to 'glue' these together while testing to ensure that the data, security, compliance, and so on are in line with the relevant standards.

Custom integration can be effective for limited, straightforward integration needs but requires significant developer resources to build and maintain. As a result, it often becomes impractical for larger, more complex enterprise environments.

Advantages

- **Specific customization:**
tailored specifically to the exact needs of each integration point.
- **Direct control:**
full control over the integration logic and data flow.
- **Initial simplicity:**
simpler to implement for a small number of applications with straightforward requirements.

Disadvantages:

- **Scalability issues:**
becomes unmanageable and complex with an increasing number of integrations.
- **Maintenance overhead:**
requires significant effort to maintain and update each integration point.
- **Lack of flexibility:**
harder to adapt or modify as business needs change, especially with tightly coupled systems.
- **Higher long-term costs:**
cumulative costs of maintaining multiple point-to-point integrations can be substantial over time.



New platform

This approach involves building a new platform that acts as an overarching layer on top of existing applications, data, and systems.

This could be a front-end portal or platform that provides a unified view and cohesive functionality across multiple different systems.

This typically involves developing new or using existing APIs that allow the new platform to interact with each of the existing systems. These APIs serve as a bridge for data exchange and process execution.

Depending on the complexity and needs, middleware, an API gateway or an Enterprise Service Bus (ESB) could be used to facilitate communication and data transformation between systems.

Advantages

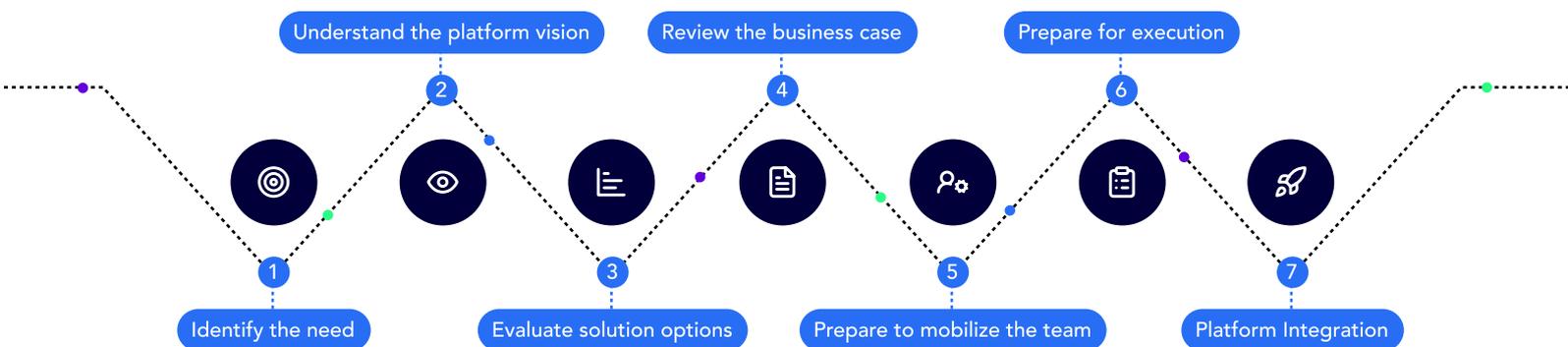
- **Customization:**
tailored specifically to the enterprise's needs, ensuring a better fit for specific processes and workflows.
- **Unified control:**
centralized management and monitoring of all integrated systems allows for improved transparency and oversight.
- **Consistent user experience:**
offers a consistent interface for users across various integrated systems.

Disadvantages:

- **High initial cost:**
development and implementation can be costly and time-consuming.
- **Complex implementation:**
requires significant time and resources for development, testing, and deployment.
- **Maintenance responsibility:**
the enterprise is responsible for ongoing maintenance and updates.

How to prepare for platform integration

esynergy has created a seven-step process for platform integration. Following each step will take an organization from the initial stages of starting to grapple with the problem, all the way to executing their platform integration strategy and continuous improvement.



Stage 1: Identify the need

This stage is twofold, on the one hand, you must consider the needs of your customers, which can be both internal and external. On the other, you should consider your own business operations. This stage consists of identifying the needs and determining the requirements of your users.

This is the time to ask questions, run workshops, conduct research, user surveys, user interviews & market analysis to ensure that you fully understand your customer's needs and their 'jobs to be done'.

Secondly, you must assess your current business operations to identify inefficiencies and areas for technological improvement. Consider leveraging data analytics to obtain insights into operational challenges, and deliver assessments. This is to get a clearer picture of the problem and the kinds of business outcomes that are desired.

Questions to ask:

- Who are your users?
- What are their pain points?
- What are your pain points?
- What are your business objectives?
- What changes in the market will impact your business?
- Is there an opportunity to improve your efficiency or reduce costs?
- What are your key operational inefficiencies?



Stage 2: Understand the platform vision

This stage is to craft a vision for the new technology platform and get the rest of the business to back it.

Here the user's pain points are translated into concrete features and alignment is established on a clear end state for the platform. Features can be product features or related to the infrastructure. You should conduct focused workshops to define and refine your vision for your new technology platform. You should include a diverse stakeholder group to ensure that you receive comprehensive input.

Use methods like MoSCoW (Must have, Should have, Could have, Won't have) to prioritize features that align with user pain points and business objectives.

Questions to ask:

- Do you have an understanding of the scope/features and functionality of what you want?
- Do you understand the business value?
- Do you have stakeholder buy-in?
- What does good look like?
- What is the priority?

"In the realm of platform development, aligning closely with business needs is paramount. As an API platform product manager, I stress the importance of APIs as they speak the native digital language of transformation. They are the building blocks that allow us to seamlessly integrate and extend our platforms, bridging disparate systems and unlocking new capabilities. In this way, we ensure that our development efforts are not only technically sound but also deeply rooted in driving business efficiency and customer satisfaction, reflecting the true spirit of digital transformation."

David Holliday

Senior Product Manager APIs

Munich RE 



Stage 3: Evaluate solution options (buy vs build)

In this stage, the value, costs, and risks of different solutions are considered and managed.

In the context of platform integrations, this would be the model that you would choose such as using iPaaS, custom APIs, or a completely new platform.

Perform a detailed cost-benefit analysis of each of the solutions you review, including a view of long-term maintenance and scalability. Evaluate your current technology stacks and infrastructure to determine how compatible they are with new technology.

Questions to ask:

- What type of architecture are you looking at?
- Is there an existing solution that could address the needs of your platform?
- Do you understand your total cost of ownership?
- Will you be able to maintain this in the future?



Stage 4: Reviewing the business case

With strategic choices made, it's time to make a business case for the project to secure the required budget and development resources.

Develop an ROI (Return on investment) analysis for each selected solution to justify your business case. Identify the potential risks and challenges and how they can be mitigated.

Questions to ask:

- Have you got a resource and program plan?
- Have you got a prototype?
- Have you got a budget sign-off?
- Have you validated there is a need for these changes?
- How long will it take to generate a return on the investment?



Stage 5: Preparing to mobilize the team

In stage five, the teams are mobilized in preparation for execution.

You'll need clear roles and responsibilities as well as incentives and metrics that are aligned with business objectives. You'll need to know if your current team has the skills required for the project and identify any training needs. If you don't have the skills then you'll need a plan for hiring.

Finally, you'll need a change management plan to ensure a smooth transition and adoption of your new platform.

Questions to ask:

- Have you decided if you're delivering this internally (review internal capacity vs external resources)?
- Have you established project governance?
- Have you identified internal systems that will be integrated into the platform?
- Have you got the 'right' team?



Stage 6: Preparing for execution

In this stage, the final preparations are made for organizing team structure, allocating development resources, and aligning behind a methodology.

Start by choosing a methodology such as agile that aligns with the project scope and the dynamics of your team. Create a plan for your resources, who will work where, and what technology resources will be used.

Questions to ask:

- Have you decided on your ways of working?
- Have you identified your software engineering capabilities?
- Do you know what resources are available?



Stage 7: Platform Integration

Launch the project! Start integrating your applications, building continuous feedback into your approach, and integrating lessons learned over time.

Implement strong CI/CD (Continuous Integration / Continuous Deployment) practices to efficiently and reliably release code. Establish robust feedback loops which include talking regularly to customers.

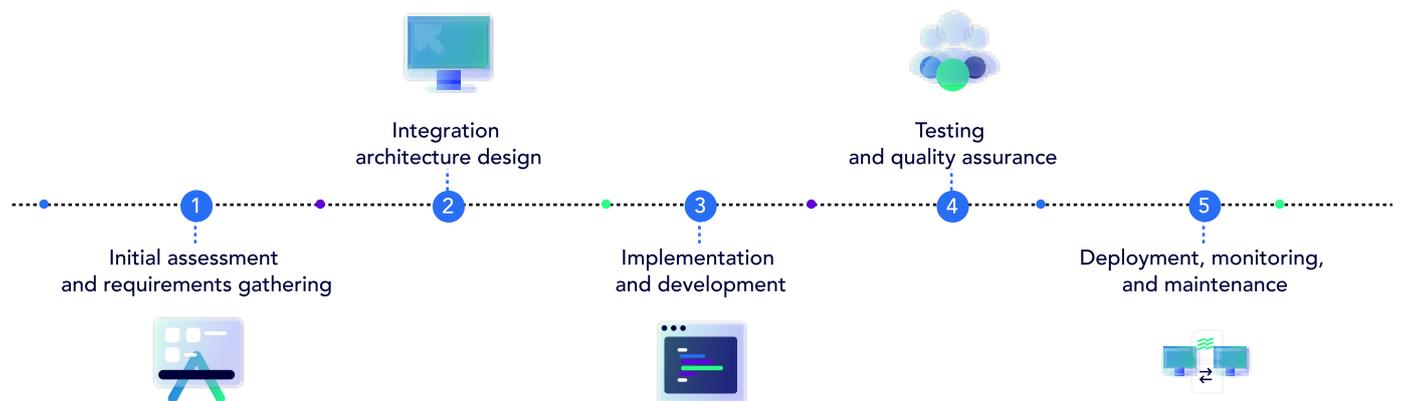
This project doesn't have a discrete endpoint. You can keep evaluating your application landscape and adding new applications to the modernization queue indefinitely.

Questions to ask:

- How can we continually improve?
- How can we get user feedback?

How to implement platform integration

In the era of digital transformation, platform integration is a critical aspect for any organization seeking to streamline operations, enhance efficiency, and drive innovation. esynergy's strategic approach to platform integration involves a comprehensive 5-step plan that ensures seamless integration of various systems and platforms, aligning them with your business objectives.



1: Initial assessment and requirements gathering

Objective:

To evaluate the existing IT infrastructure and gather detailed requirements.

Activities:

- Conduct an audit of existing platforms, systems, and data sources.
- Identify business processes that will be impacted by the integration.
- Gather technical and business requirements from stakeholders.
- Establish clear integration goals and success metrics.



2: Integration architecture design

Objective:

To design an optimal integration architecture that meets business needs.

Activities:

- Develop an integration architecture that supports scalability, security, and performance.
- Select the appropriate integration tools and platforms.
- Define data flow and integration points between different systems.
- Ensure compliance with data privacy and security standards.



3: Implementation and development

Objective:

To develop and implement the integration solution.

Activities:

- Develop custom integrations or configure existing integration platforms.
- Integrate disparate systems, databases, and applications.
- Implement API-led connectivity for seamless data exchange.
- Test integrations in a controlled environment before going live.



5: Deployment, monitoring, and maintenance

Objective:

To deploy the integration solution and ensure its ongoing effectiveness.

Activities:

- Deploy the integration solution in a live environment.
- Monitor the integration for performance and issues.
- Provide ongoing maintenance and support.
- Continuously optimize and update the integration as needed.



4: Testing and quality assurance

Objective:

To ensure the integration solution works as intended.

Activities:

- Conduct thorough testing including functional, performance, and security tests.
- Validate data integrity and consistency across integrated platforms.
- Resolve any issues or bugs identified during testing.
- Involve end-users in user acceptance testing (UAT).

Platform integration best practices

This section lays out some common best practices that we have found are critical in ensuring the success of platform integration projects.

1: Clearly define the scope

One of the most common mistakes when approaching a project like this is failing to properly define the scope of the integration, i.e. exactly which systems, data, etc. are to be integrated.

This is when 'scope creep' starts to emerge, when the edges of the project keep growing but without proper strategic thinking or oversight.

This is why it's critical to clearly define the systems, data, and services that need to be integrated and the resources required. And to defend this boundary strictly.

2: Move from product to platform thinking

Product thinking is about finding ways to identify and prioritize customer needs and systematically build solutions for those needs. But there is a step beyond this.

Instead of thinking in terms of specific products, you want to think in terms of platforms: interconnected ecosystems of business capabilities and building blocks, upon which new products can be built.

Such platform thinking helps you to harmonize your existing applications and data while leaving room to add or subtract new applications, tools, and data as your business grows and changes, without having to start from scratch or bother with messy future integrations and migrations.

The resultant ecosystem is a highly scalable, highly composable suite of capabilities that is at the service of the wider business needs.

3: Create an Integration Centre of Excellence (ICoE)

An ICoE is essentially a mini-consultancy inside your business that helps to disseminate knowledge, establish standards, and support different departments on their integration journey.

This unit can consist of a cross-functional team of integration experts that promote platform integration as a necessary discipline within each corner of your business, supporting these teams with the resources and expertise to help them adopt new platforms and deal with the waves of change.

4: Take your people with you

There's a temptation to approach platform integration as a technology-centric issue.

But your people are the reason you are integrating platforms in the first place. And their needs must come front and center. Without proper support for adoption, your platform integration will fail!

Firstly, conduct user research to figure out what the biggest issues, limitations, and bottlenecks are as a guide for the scope of your integration.

Secondly, when the integration is being deployed, ensure you are just as invested in supporting them to adopt any new tools, processes, interfaces, front-ends, and so on as you were in getting the technology working.

For example, if there is a new front-end portal for accessing a host of previously individualized applications, they will need support to navigate the new portal and be walked through any new features, etc. Don't assume they won't 'figure it out' on their own.

"In the fast-paced world of tech platform development, the key to swift success lies in harnessing the power of experienced teams and deep customer involvement. By focusing our efforts on lean, agile practices and maintaining a relentless customer-centric approach, we can rapidly translate complex technical challenges into tangible business solutions. This streamlined synergy between expertise and customer insights ensures not just speed, but also precision in delivering impactful results."

Matt Lockyer

Matt Lockyer - Platform Practice Lead



5: Take your data with you, too!

When different systems of engagement (customer-facing products and services) are integrated there's a focus on making the resultant platform faster and more agile, to unlock new business value.

But your newly integrated platform will still be intertwined with a massive legacy estate of systems of record, such as accounts, ledgers, order books, and so on, which tend to be very strongly data-focused.

However, when you accelerate one side and not the other, you end up creating an unstable split within your platform with two halves operating in different ways and at different speeds.

To get optimal outcomes, you need to fundamentally change how you work with data, moving to more agile, decentralized, distributed approaches, such as data mesh.

How to measure success and ROI

How you measure success and track the ROI of your platform integration program will depend on your initial business goals and objectives.

In this section, however, we will give an overview of the five main areas where businesses typically develop Key Performance Indicators (KPIs), along with example metrics.

1: Operational efficiency and agility

How has the integration impacted your business's operational efficiency and agility?

- Time-to-market
- Deployment frequency of integrated systems
- Reduction in manual processes

2: Costs and resource utilization

Did the cost savings and improved utilization from the integration outweigh the initial investment?

- Change in support and maintenance costs post-integration
- Resource efficiency
- Operational cost savings

3: User and customer satisfaction

Has the platform integration improved the experience for users and customers?

- User and customer satisfaction ratings
- User and customer engagement metrics
- Reduction in customer service issues

4: System performance

How has the integration affected the performance and reliability of your systems?

- System response times
- System availability and uptime
- Scalability improvements

5: Data and technology democratization

How has the integration improved your people's access to key data, tools, and technology?

- Data quality and accessibility
- Tool adoption rates
- Reduction in customer service issues

"When building out platforms, especially service platforms, I've found it critical for success to balance out the technical capabilities, with a strong business value proposition and an efficient go-to-market engine to deliver return on investment. In today's modern world, platforms require understanding and investments from cross-functional teams, usually being led by an aspirational product and technology vision."

Ryan Clifford

Platform Group Product Manager



Key challenges and barriers

In a complex organization, the key challenge is one of balance.

How do you weigh the urgency of unleashing new business value (through integration) against current resource demands, business continuity, limited skills, security considerations, and so on?

Here are the top challenges that must be carefully weighed and considered:



People

- **Skills:**

you may lack in-house expertise in modern technologies or methodologies required for the integration process, which can be resolved either by calling on third parties or an internal training program.

- **Resource allocation:**

carefully consider the balance of development resources required for your integration project versus other value-creating projects and existing customers.

- **Stakeholder management:**

If the business and the key stakeholders aren't with you (e.g. the teams that run your apps), your modernization project will grind to a halt. Ensure that you include stakeholders from the business (and third parties externally) at every stage.



Business

- **Business continuity:**

critical data and applications need to be continuously available, and the prospect of downtime during integration can be a significant barrier to major changes. Utilize strategies like phased rollouts and backup systems to maintain the continuous availability of critical data and applications during integration.

- **Regulation and compliance:**

enterprises in regulated industries (e.g. finance, healthcare) need to ensure that any changes do not breach compliance mandates. Regularly consult with compliance experts to ensure all integration activities adhere to industry-specific regulatory mandates.

- **Balancing speed and value:**

finding the sweet spot between minimizing disruption and time-to-value versus ensuring you are delivering real value is difficult. Implement agile methodologies to balance quick delivery with delivering substantial business value.



Technology

- **Data transformation:**

It's challenging to integrate data that are siloed, in different formats, etc., but if you don't transform how that data is managed and operated you will have problems with quality, lineage, and access. Invest in robust data management tools and practices to ensure data quality, lineage, and accessibility across diverse formats and systems.

- **Security:**

whenever modifying, moving, or integrating applications there is a possibility of creating new vulnerabilities (e.g. moving legacy apps to the cloud). Embed security considerations into every stage of the integration process, from planning to testing and deployment.

- **Complex technology landscape:**

Integrating various systems with different architectures, technologies, and data formats is difficult, particularly if they have accrued significant technical debt in the form of workarounds and hotfixes. Adopt an integration strategy that accommodates diverse architectures and technologies, focusing on modular and interoperable solutions.

We are here to help

esynergy's approach to platform integration is designed to be holistic, scalable, and aligned with your business's specific needs. We focus on creating an integrated ecosystem that not only streamlines operations but also unlocks new opportunities for growth and innovation. Our goal is to ensure that your integrated platforms are not just technically robust but also strategically advantageous, providing a solid foundation for your digital transformation journey.

We've helped many organizations integrate their platforms to achieve better performance, scalability, and resilience, view our platform integration case studies:

- [CPS modernizes Case Management System](#)
- [Leading fintech company integrates payments system](#)
- [Leading wealth management provider integrates acquisitions](#)

If you are struggling with any of the key challenges of platform integration:

- Making a business case
- Winning over key stakeholders
- Deciding what approach to take
- Determining the most appropriate tools and technologies
- Measuring success and key metrics
- Continuous improvements

Get in touch with our
Head of Platforms and
Platform Integration:

Matt Lockyer
matt.lockyer@esynergy.co.uk





New London House, 6 London Street, London, EC3R 7LP



info@esynergy.co.uk



0207 444 4080