celigo

Definitive Guide:

Building an Application Integration Strategy to Support High Growth

Whitepaper

Table of Contents

Definitive Guide:

Building an Application Integration Strategy to Support High Growth

Introduction

When you think of ways to accelerate growth, developing an application integration strategy probably isn't the first thing that comes to mind. You might think about how to improve your data sharing across marketing, sales, operations, and support or how to automate those burdensome manual processes, or how to free up your technical resources so they can work on making your product better.

Developing an integration strategy can help you do all those things and more. It might not be sexy but we've seen lots of companies skip or delay this step in their preparation for high growth and always regret it. They find themselves quickly overrun with data silos, disconnected processes, inconsistent customer experiences, and the need to add resources to counter the resulting operational gaps.

Why Integration Matters to Business Leaders

84%
say integration challenges slow business process innovation

83%
say data silos create business operations challenges

64%
say they cannot provide a connected user experience across sales, marketing, services, and support

In the following sections, we'll discuss how to create an integration strategy that meets your cloud data integration needs, aids growth, and minimizes technical debt. We'll walk through evaluating different integration approaches, including pros and cons. Lastly, we'll go through four best practices to follow when implementing an application integration platform to help free up your IT resources. We've gathered data from hundreds of other high growth mid-sized companies and found that with the right app integration solution and strategy in place, the stage is set for your technology to support rapid and continuous growth.

The difference between integration types

Researching integration tools can be confusing as multiple terms are used interchangeably to describe integration. Before we get further, let's set some context and properly define these terms and how they are related.

Application Integration

Application Integration is the process of bringing resources from one application to another and often uses middleware.[1] In most modern organizations, this generally means connecting multiple cloud apps together. This can also be referred to as Cloud Data Integration. In organizations where legacy systems exist, application integration also includes connecting on-premise and cloud apps. The applications addressed are typically core applications like ERP, CRM or e-commerce platforms where many business processes rely on data from other cloud-based apps.

Who uses these solutions:

Cloud application integration solutions are typically tailored towards a tech-savvy business user or IT staff, but don't usually require specialized skills.

On-premise application integration solutions are typically tailored towards large enterprise use cases and used by IT staff with specialized skills.

1. *https://www.techopedia.com/definition/16559/application-integration*

Data Integration

Data Integration is a combination of technical and business processes used to combine data from disparate sources into meaningful and valuable information. Data integration goes beyond simply moving data from point a to point b by doing things to the data to make it more usable. Legacy data integration platforms are traditionally thought of as on-premise middleware solutions. They are often referred to as ETL solutions, meaning they extract, transform, and load data to another system. Modern cloud data integration solutions follow the same concepts as on-premise data integration tools but, obviously, are used for integrating cloud apps or cloud to on-premise applications.

Who uses these solutions:

Cloud application integration solutions are typically tailored towards a tech-savvy business user or IT staff, but don't usually require specialized skills.

On-premise application integration solutions are typically tailored towards large enterprise use cases and used by IT staff with specialized skills.

Integration Platform as a Service (iPaaS)

Integration Platform as a Service (iPaaS) is a modern application integration solution that borrows key data integration concepts. iPaaS allows you to not only integrate multiple applications but manipulate, augment, and validate data quality as data is being moved. You can think of iPaaS as the offspring of app integration and data integration parents. If deployed on-premise, they follow the more traditional models of enterprise application integration (EAI) or ETL tools.

Who uses these solutions:

iPaaS platforms can be tailored to both tech-savvy business users and IT staff. This is dependant on the complexity of the use cases the platform supports and if the solution is cloud-native vs. an on-premise solution running in the cloud.

In general, relatively straight-forward uses cases can be accomplished by tech-savvy business users as most iPaaS solutions provide guided configuration options. For more complex use cases, typically IT or DevOps staff are implementing the solution.

API Integration

API Integration allows two applications to talk to each other through software called an Application Programming Interface (API). APIs power most modern applications and enable others to develop interfaces to those applications. For example, if you buy something on Amazon using your phone, you're using an API. In the context of app integration, APIs enable the integrations to happen. Most application integration tools and iPaaS solutions provide an abstracted view of an application's API by offering pre-built connectors or adapters and some level of API management. API integration platforms allow you to work with the APIs natively, and offer tools to make it easier and faster to build integrations. They also typically have more robust API management capabilities necessary for those building APIs for others to use.

Who uses these solutions:

API integration platforms are built for and used by developers.

How to create an integration strategy to power growth

Most conversations we have with growing companies start with one or more of the following statements:

"Our business teams need to log into multiple systems to get the data they need"

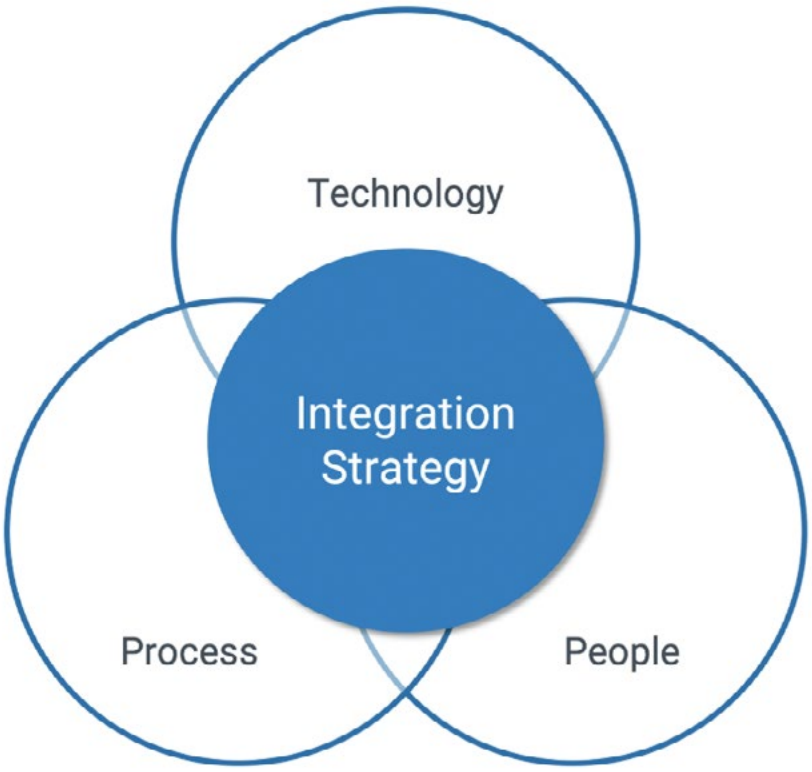
"We can't scale with all the rekeying we are doing"

"We have multiple manual processes"

"We're adding a lot of new technology and it's taking too long to integrate it into our existing systems"

To grow your business, you must scale business operations. To scale operations takes people, process and technology. Your integration strategy sits at the cross-section of these three pillars. The typical goal for any integration strategy is to:

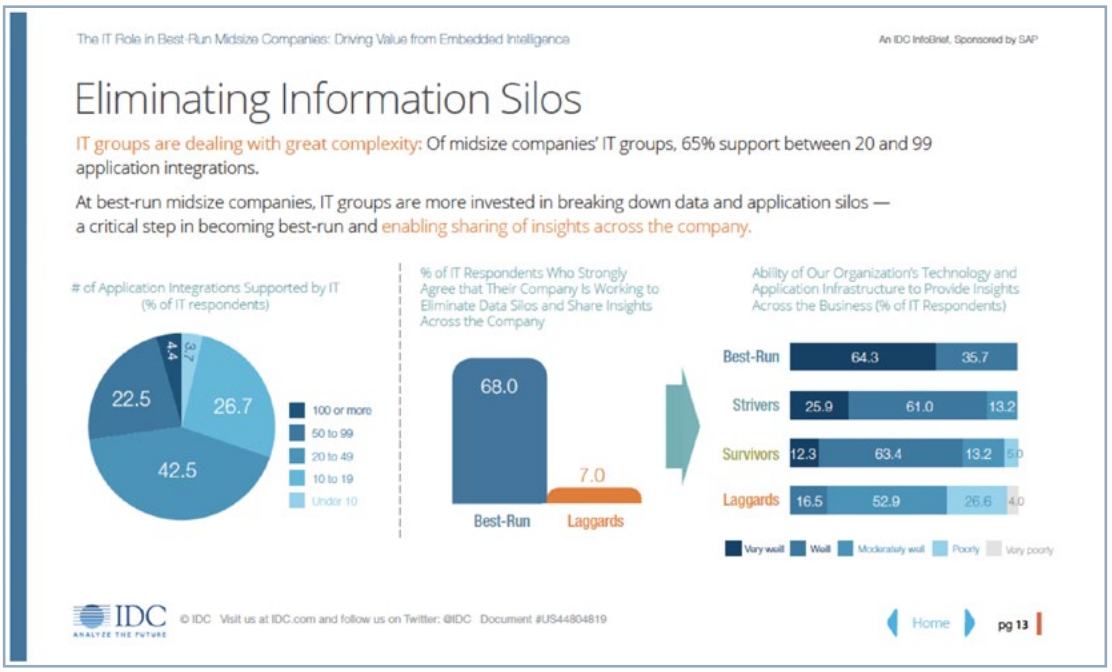
- Enable business teams with data in the systems they use
- Innovate and automate processes
- Free up resources to work on high-value projects and tasks
- Fully utilize your technologies



Three Steps to Get Started

Step 1
Determine is your system of record. Meaning, what system is the source of truth? For some organizations, it's their ERP but for others, it's their CRM. There is no right answer, but determining the system of record is a critical starting point for your integration strategy. The system you choose is your data management system and the anchor for the bulk of your integrations. Without clearly determining your system of record, you will find it impossible to trust your data which will make it difficult to decide what data to move to what systems and when.

Step 2
Map your applications. A recent survey by IDC of mid-sized companies showed that 65% use 20-99 different cloud applications to run their business[2]. Of those companies, the best-run ones were actively working to eliminate data silos while the laggards were barely making a dent in the problem.



Step 3
Identify the apps that need to be integrated, and remove the apps that don't add enough value. Try to simplify your application landscape as much as possible so your teams are not managing business applications and building integrations that bring no real value.

***Tip**: Be harsh.*

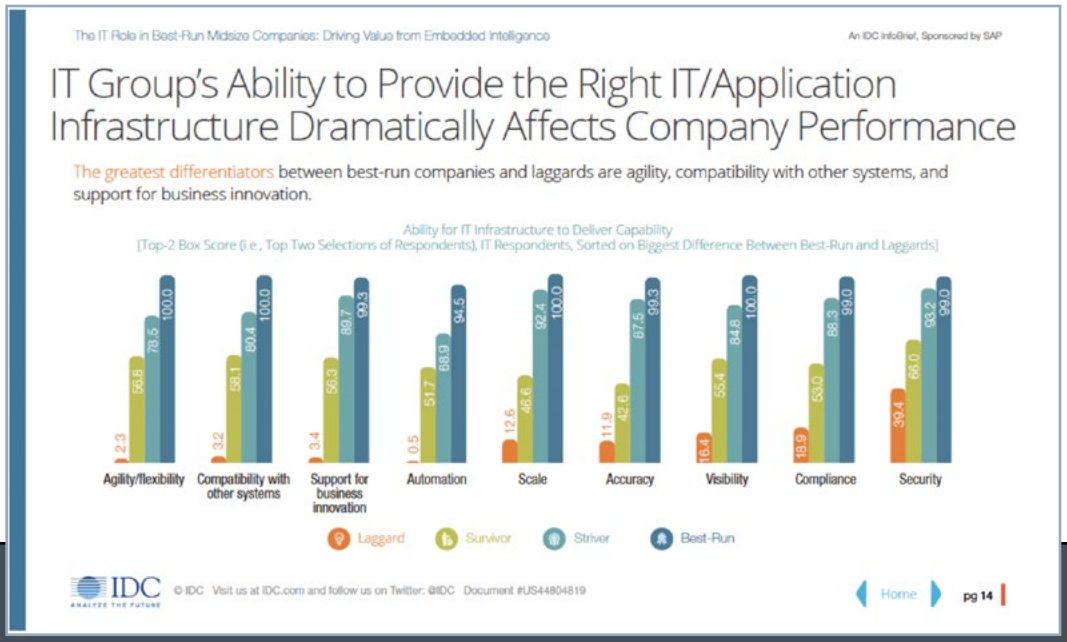
2. The IT Role in Best-Run Midsize Companies: Driving Value from Embedded Intelligence
https://www.sapphiresystems.com/download_file/force/1985/4483

Start Small but Don't Think Small

Integration requirements tend to be fluid in high growth companies. Your integration strategy needs to be flexible enough to accommodate changes without forcing a massive IT project or forcing you to rewire your integrations when you add new technologies. A common mistake is to think you just need to build a couple of integrations and all will be good. That might be the case initially but it rarely remains that way.

As you think about your integration needs, look beyond immediate requirements to what might be needed in 3-6-12 months time. Giving equal weight to current and future requirements plus a healthy nod to the unknown reduces the risk that you'll need to rework your strategy and technology decisions in a few months.

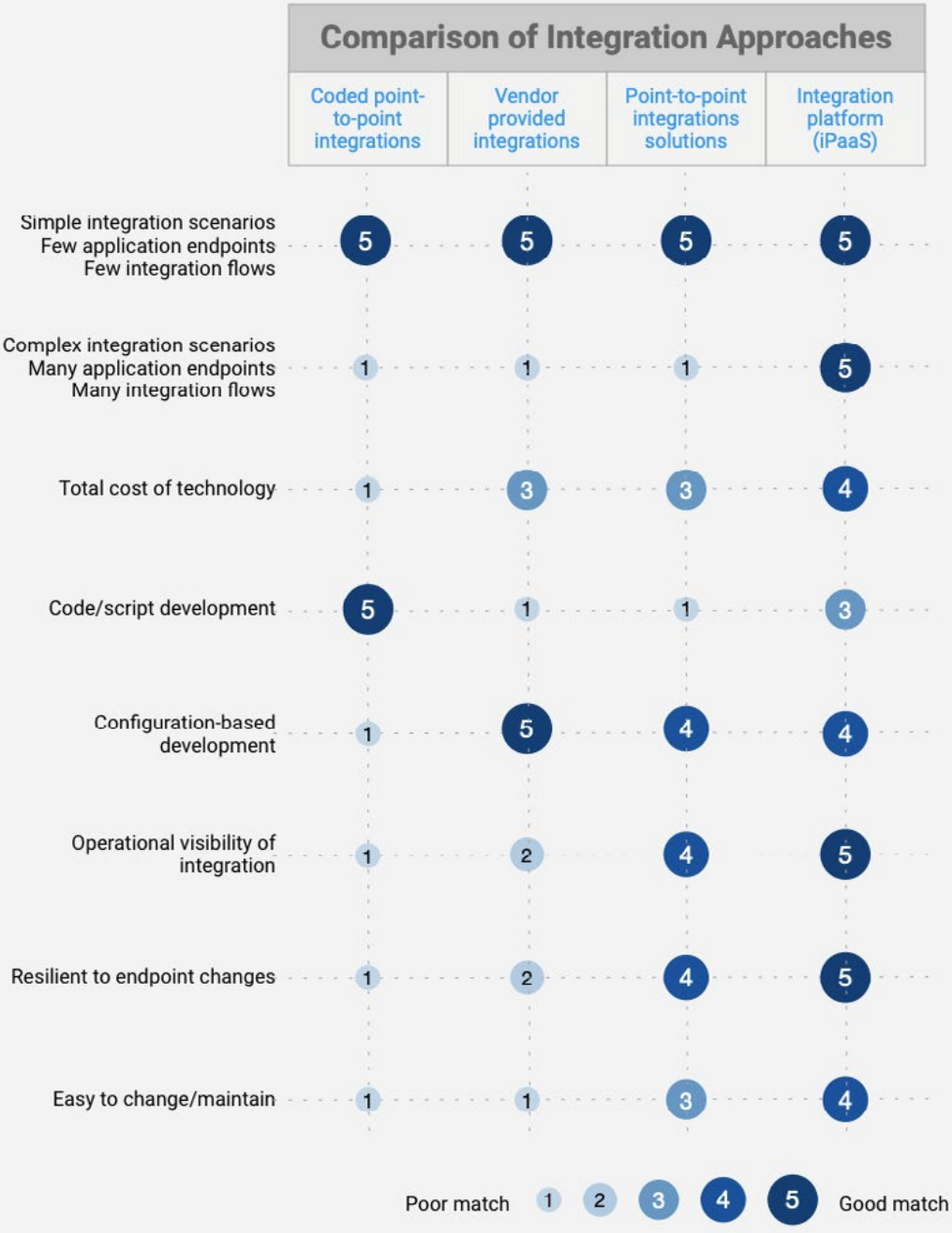
As the IDC chart below shows, the greatest difference between best-run companies and laggards is the ability for IT to deliver an integrated application infrastructure that supports business agility and innovation.



Free Advice: Don't try to boil the ocean and integrate everything all at once. It will take too long to execute and see the value. Instead, create a staggered execution plan so you are consistently delivering value and incremental improvements. In development terms, think Agile not Waterfall.

If budget is a concern, most integration vendors provide tiered pricing so you can start small and expand as needed. Once you've nailed implementation of your mission-critical integrations and realize value, you can usually get the funding needed to expand to secondary systems and processes.

Selecting the Right Solution: Comparing Integration Approaches



Coded point-to-point integrations

In this approach, a developer is writing code to integrate APIs. This is done by an internal or external development resource. An API integration tool, like Cloud Elements, may be used to help expedite the integration development or developers just code directly to the API.

When to consider this approach

- You only have a couple of applications to integrate to a hub app, i.e. 1-2
- You already have IT developers on staff with bandwidth to implement, manage, and maintain the integrations
- You don't want to leverage third-party solutions

Cautions

- If you have more than a couple of applications to integrate, writing custom code for each integration may take too long to meet business needs (average of 41 days per integration)
- Custom integrations require a dedicated resource(s) to maintain and manage as business teams will have no visibility to errors or any ability to make changes without using IT
- IT will need to resolve any integration errors
- If integrations are mission-critical and time-sensitive, you will need to staff for 24x7 365 rapid response support
- You will need to create a process to be made aware of endpoint (API) changes to avoid unintended downtime

Vendor provided (native) integrations

This approach is relatively common as many apps provide native integrations to large hub applications like Salesforce.com. In this approach, the vendor has built an integration that typically a business system admin configures to exchange data between their app and one other. For example, HubSpot to Salesforce.

When to consider this approach

- You don't have any unique or custom integration requirements
- You don't have a need to send the same data to multiple other applications

Cautions

- You may have to buy a higher tiered version of the respective apps to gain access to the APIs and integration services
- It is hard to customize these integrations outside of what the vendor provides out-of-the-box
- Determine who is responsible and a process for managing errors as typically only system admins have the necessary visibility
- Visibility and error handling is basic often with no ability to recover without resending the record from the source system or manually re-keying

Point-to-point integration solutions

These solutions are typically built for individual tech-savvy business users to enable relatively simple integrations between the systems they use. They provide a highly abstracted view of the underlying APIs that enable users to configure integrations so no specialized skills are needed. Users typically need admin access to the systems they are trying to integrate or access to API keys. Example solution providers are FarApp and Zapier.

When to consider this approach

- You have individual or department level integration needs
- You only have a couple of apps to integrate with a handful of simple integration flows
- Business teams own apps and integrations

Cautions

- Because of the high abstract interface, It is hard to customize integrations outside of what the solution provider has built
- These solutions do not scale well to organization-wide use
- Pricing is “task” based which can get expensive if you have large numbers of small data packets moving between systems
- Visibility and error handling is basic
- There are no checks to ensure integrations are not stepping on each other, i.e. different integrations are mapping different values to the same field
- It is very difficult to perform audits when there is a proliferation of point-to-point integration

Integration Platform as a Service (iPaaS)

This class of integration solutions is relatively new to the market, coming up in the past 3-5 years. The are designed to eliminate the problems that often result from a proliferation of point-to-point integrations by providing centralized integration management. Like point-to-point solutions, they abstract APIs to enable configuration of connections from apps to the iPaaS. Beyond that, the user experience can vary widely. Some iPaaS solutions are built for large enterprise use cases so they are more like integration "toolkits" better suited for IT use, such as Dell Boomi, Jitterbit, and Mulesoft. Others offer more pre-built integrations designed to help organizations get started quickly and the ability to build custom integrations to serve any use case. These solutions tend to serve midsize companies better as they can be used by both business and IT teams to build simple to complex integrations. Celigo and Workato are good examples of iPaaS solutions that cater to the mid-market.

When to consider this approach

- Your integrations are mission-critical
- You have a mix of complex and simple integrations
- You have multiple applications to integrate
- You want business teams to manage and maintain integrations, not your IT team
- Your integration requirements are fluid and change often
- You need to build custom integrations
- You need advanced operational visibility into your integration performance, including error handling and auto-recovery

Cautions

- Not all iPaaS solutions are the same, so develop a checklist of your specific requirements
- Ensure you are not paying for enterprise-level features you will never use

4 Rules to Follow to Free Up Your IT Resources

IT in mid-sized companies are always pulled in ten different directions at once which constantly puts you in a “reactive” mode. The idea of implementing an integration platform, like an iPaaS or an Integration Hub, might sound like too big of a bite to take with the limited resources you have. The reality is quite the opposite.

Implementing an integration platform is a proven way to free up your IT resources by eliminating low-value, labor-intensive and error-prone processes from their day-to-day. All you have to do is follow these four rules in your integration strategy:

Standardize and reuse integration templatesthe data they need”

Enable business self-service management and maintenance

Use third-party solutions more extensively

Pick a solution with a low total cost of technology

1 Standardize and reuse integrations templates

Standardization ensures the same integration process is used across departments and common integrations are reused. This prevents teams from reinventing the wheel for every new integration. For example, if various systems need customer data from your ERP, you build a predefined "template" that defines the right fields and tables to use so there is no question the right data is moved. The "template" is then reused for the next system that needs customer data, cutting the time to create the integration by up to 80% and potentially pushing the integration work out of IT to the business teams.

2 Enable business self-service management and maintenance

Business teams understand the priority of their integrations and usually know better than IT what is causing a specific integration issue, especially related to data errors. By pushing the management of integrations to the business, you actually will reduce errors because the business teams are often better equipped to find the root causes and fix them. Look for solutions that provide visibility into the performance of integrations through a non-technical UI, auto-recovery capabilities, and an ability to fix issues on the fly.

As mentioned earlier, business requirements are often fluid at high growth stages. Enabling business teams to do their own change management, such as modifying a mapping, adding a new field, etc., benefits everyone. IT resources are not diverted to maintain integrations and business teams have control of their integrations. Many modern cloud data integration platforms provide an option for wizard-like user experience to build integrations and enable tech-savvy business users to become integration specialists.

However, with self-service, it is important that IT establishes integration "best practices" and, where possible, creates standard templates and processes for business teams to follow.

3 Use third-party solutions more extensively

Yes, this is a bit self-serving but let me explain why this is here. If you are not using third party tools for integrating apps, then your IT team is coding to the APIs. It takes, on average, 41 days for a developer to integrate an API with advanced functions. If you have a couple of systems that need to be integrated then having your in-house team build and manage the integrations might be ok. However, as mentioned above, mid-sized IT departments are managing somewhere between 20-99 different apps. As an alternative to internal resources, many midsize organizations overpay for short-term external resources to fill the resource gap. However, once those external resources leave, your in-house team must take over the support and maintenance of an integration they did not build. A better long-term approach is to invest early in third-party tools that minimize integration development time, maintenance and support. By starting early, you can scale at speed without incurring additional costs or take time out to rework integrations.

4 Pick a solution with a low cost of technology

There are many factors that go into determining the cost of technology or the total cost of ownership for a technology solution. They include:

Implementation complexity:
Watch out for solutions that require a significant amount of professional services/consulting to get up and running. Challenge vendors if implementation timelines extend beyond 3 months for the initial integrations. Both can be indicators of operational complexity that may require multiple dedicated resources to manage.

Feature richness designed for midsize:
The needs gap between midsize vs. enterprise companies is shrinking but not closed. Many vendors who sell to large enterprises often fail to meet the needs of midsize companies. They trim down features to reduce costs, therefore, limiting functionality. Depending on what has been limited, IT or business teams may need to fill the gap with people or pay a similar price as large enterprises to gain access to necessary features.

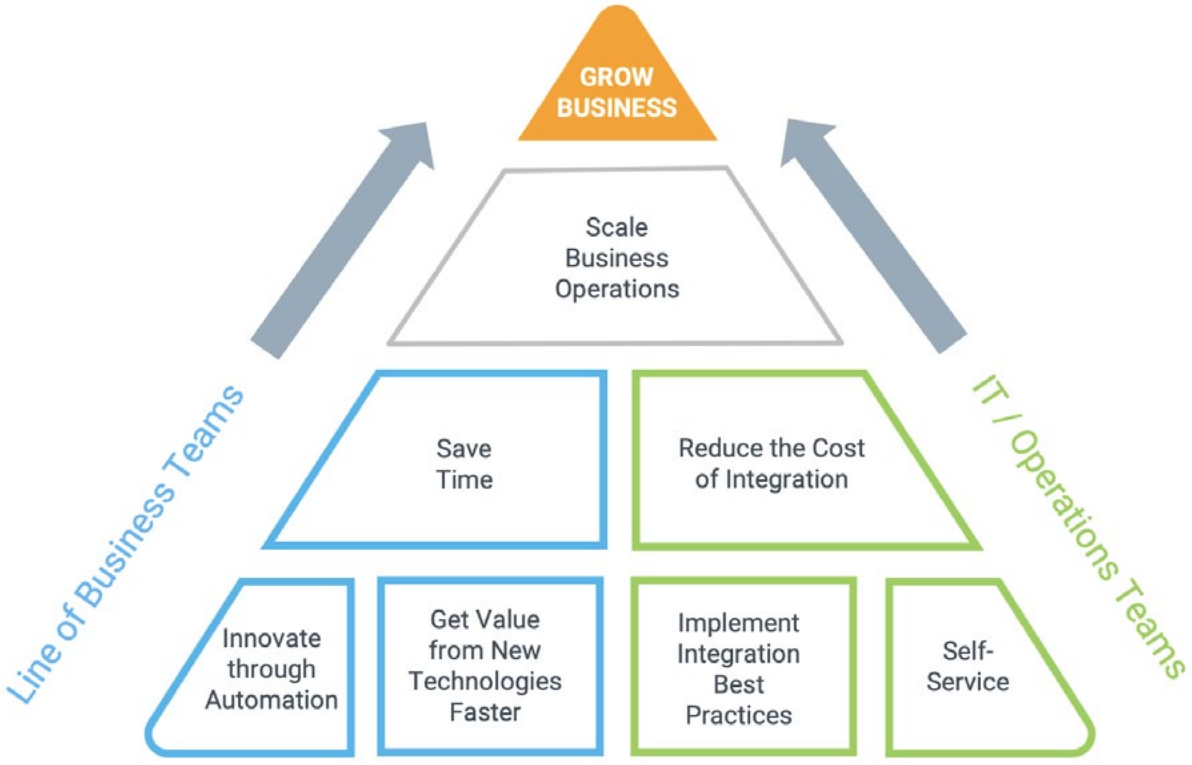
Operational visibility and support:
Visibility into your integrations, how they are performing, errors, and easy failure recovery is often overlooked as selection criteria. If your operations teams spend hours monitoring integrations and digging through logs to manually recover failures, that is time they are not spending on high-value projects. Even if you have just a few errors a week, it can require a dedicated resource to just make sure the integrations are working. Robust operational visibility is a core pillar to keeping the cost of ownership down. This includes the ability for non-technical users to monitor and recover errors so the responsibility of managing the integrations is moved to the respective business teams.

Change management:
Much like death and taxes, changes to the business requirements for integrations are inevitable. As much as possible, you want to enable the business teams with self-service change management. While not every change can be done by the business, most changes are relatively straight-forward, such as mapping changes or adding a new field. These are changes that business teams should be able to manage through a self-service interface. The more IT teams have to handle change requests, the higher the cost of technology as those resources are expensive and are being diverted from working on high-value projects.

Price:
When people think of the cost of technology, they usually think of the price. Obviously, it is a consideration but it is only one data point. For example, coding to the API seems like it would be a low-cost option since you don't have to "buy" anything. However, if you factor in visibility and change management, the cost of that option increases significantly. You will need to staff your IT team appropriately to meet business needs with at least one dedicated developer and an operations headcount to manage integrations.

Summary

To prepare for high growth, you must improve data sharing across teams, innovate business processes, and free up resources. Developing an integration strategy and implementing an integration platform puts the necessary process and technology pieces in place to innovate and scale your operations.



Summary

Your integration strategy should delivery these foundational building blocks:

- Ability to innovate business processes through automation
- Ability to integrate new technologies into existing infrastructure
- Implementation of best practices
- Self-service management and monitoring

When done right, you free up time for both your business and technical teams to focus on higher-value projects. Through automation, your business teams are no longer under pressure to fill operational data gaps with people. Instead, everyone has the data they need in the system they already use. Your sales, marketing, customer service, and support teams are all focused on providing consistent customer experiences that drive growth.

For your IT team, with automation, implementation of best practices and empowering the business with self-service, you no longer need technical resources focused on chasing down errors or building new integrations from scratch. The relatively low-value nature of integration work can be shifted to business teams, who want the control anyway, reducing the cost of integration and increasing agility. Your development team is now free to work on making your product better so you can sell more, at a higher price and make your competition irrelevant.

SKU: WP-DBG-20190805