exalate

Build vs. Buy: The Pitfalls of Building your in-house Software Integration Solution

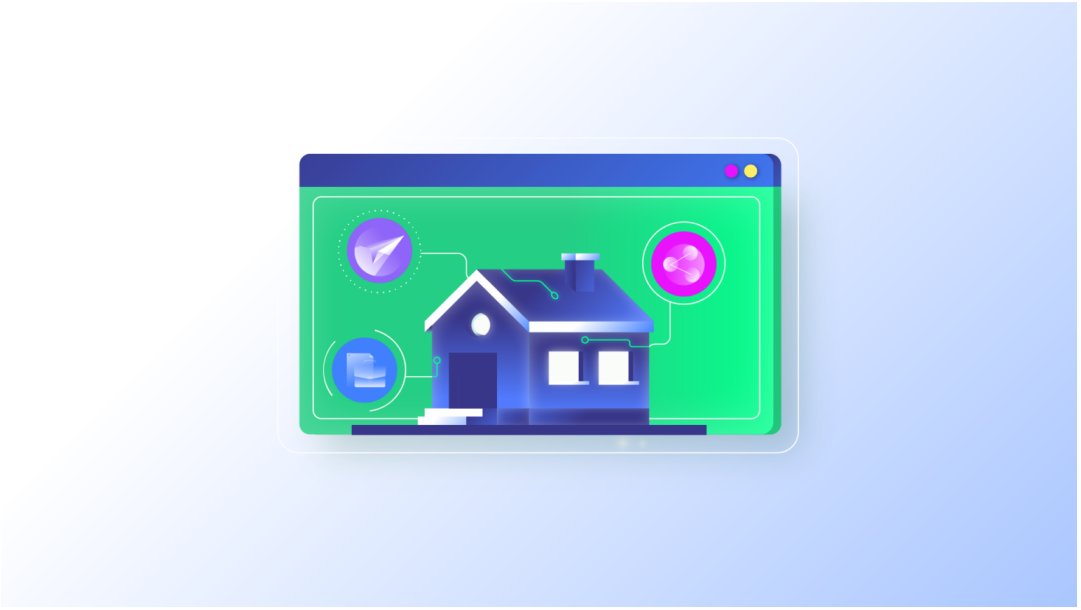


Table of contents

In the global knowledge economy, companies need to be agile enough to stay competitive. To achieve this agility, corporations are constantly trying to improve their business workflows for their products and services. This has given rise to work management systems that depend heavily on integrating data and information within the company to optimize these internal processes.

Increasingly, today's corporations also live in an interconnected world. From large corporations to small and medium enterprises, companies have a need to exchange data with one another at many stages of their workflows.

Companies have made great strides in improving integration processes between their own teams. However, that integration stops at company borders because they often use different work management systems.

Even those firms that use the same systems But it's this data in these different systems that must be integrated if companies are to reduce the friction of exchanging information and improve efficiencies.

So the dilemma remains. How do companies integrate data between themselves in the most efficient means possible? In an earlier article The Journey of Software Integration, after briefly looking at manual and shared information transfers, we concluded that to optimize work management systems without sacrificing the internal processes of companies, automated integration was the only viable answer.

But what's the best method of cross-company integration? Should companies build their own integration, or look to buy a 'best of breed' software package? In this article, we are going to expand on the advantages and disadvantages of each approach by examining the phases of an integration project.

Phases of the Integration Project

The Environment

Integrations are quite complex. Not only are there initial technical challenges to overcome, but there are many other factors to take into account. Companies come to the table with totally different organizational structures, people with different skill sets and agendas, different processes, and even different work management systems. These factors all add to the complexity of the integration process.

We're going to focus on one portion of the process, examining whether to develop a custom integration application in-house using existing company resources or to purchase one of the commercially available software applications that provide this service.

To read more about the entire integration process, check out The Journey of Software Integration.

The Initial Set-Up

Before a company can accurately assess the advantages or disadvantages of a make or buy decision, they must clearly spell out the requirements for data integration. This includes:

- Determining the communication path between the two environments: What deployment model will the companies use? What are the specific authentication, authorization, identification, information, data, and network security issues to be used during the integration?

- Determining a common data model: This means the companies must ensure that the data exchanges are properly mapped and any needed data transformations between systems occur properly.

This set is vital for any successful data integration project and becomes especially critical in a cross-company integration scenario. A three-space text field will not map to a two-digit text field no matter how many attempts are made to force the issue. While this may sound trivial, there have been specific instances where technicians have spent days troubleshooting such a 'trivial' issue.

- Identifying any redundancies: This includes identifying failure paths and proper notification protocols to take place in case of probable failure. It's also important to take into account unforeseen delays and data synchronization between companies and systems. Plus, there needs to be a formal rollback system in place in case of unexpected system crashes.

- Accounting for data tracking: To ensure our data integration is valid and complete, any cross-company integration application needs a tracking system to log all data transfers. This is vital to conduct audits of data transfers to ensure the reliability of the system.

- Developing application functionality: Assuming that all of the requirements have been accurately gathered, the developers can start to code in the required functionality. While seemingly straightforward, in reality, it can be an arduous task. Programming the functionality might often seem straightforward enough, but the real work always comes in determining how to handle errors and exceptions.

More sophisticated applications are developed to capture and deal with all data errors and exceptions without causing the application to terminate. It's important that the fitting functionality of the application not only includes proper execution of code but also alerts the identified integration issues back to the appropriate technical personnel.

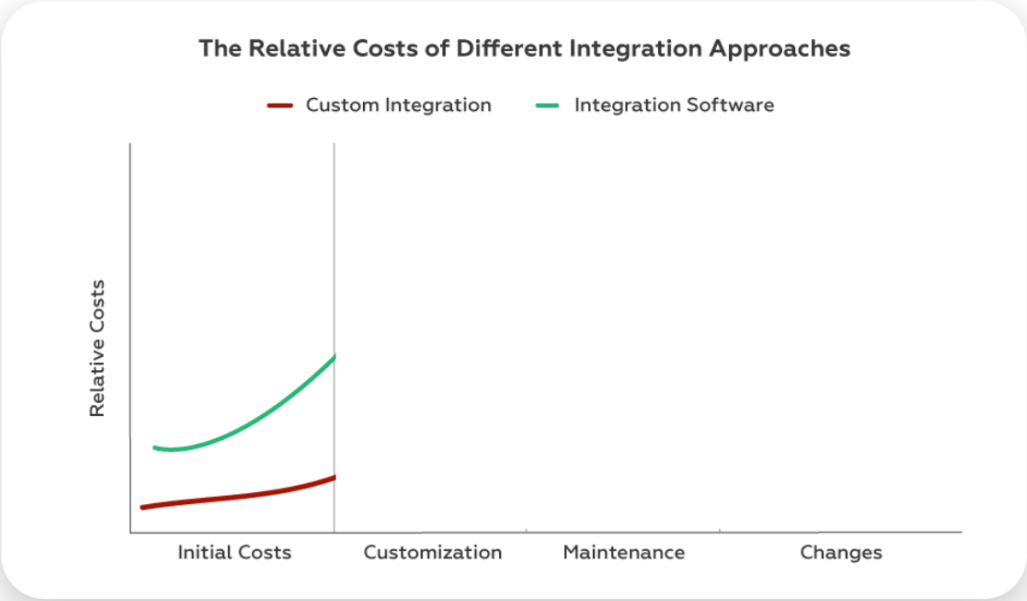
- Validating the solution: Once the application has been developed, one of the most challenging aspects will be the validation of the integration functionality. To ensure the proper functioning of the integration application, it needs to be 'wrung out' by being subjected to tests for functionality and performance.

These tests must also validate the stability and availability of the integration to ensure that duplication or other unintentional errors in the integration process don't cause the system to crash.

Additionally, tests must also be developed to discover unexpected errors that lead to system crashes, causing unintentional loss of data. For instance, trying to put a three-letter country code into a two letter text field - KOR (for Korea) will not cross into a KR text field, no matter how much we wish it would.

- Managing the production release: Now that the application has been developed and properly validated, it needs to be released to the production systems for the various companies involved in the cross-company integration process.

It also needs to be rolled out into two different systems at the same time. Implementation into the systems of all companies involved must be completely transparent to the end-users. Because any unplanned disruptions to production work management systems can cost the companies involved thousands (if not millions) of dollars in lost revenue.



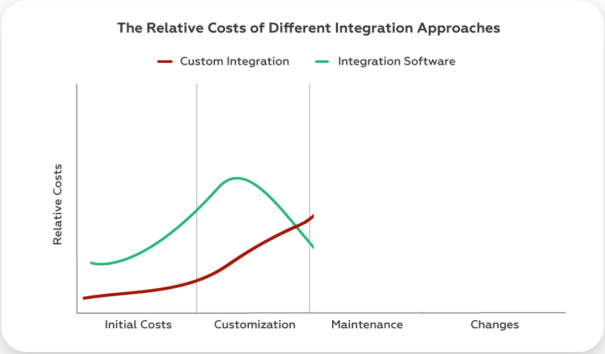
Any rollout of an application for production must also include rollback plans in case of unintended failures or consequences caused by the application.

- Planning ongoing monitoring: Monitoring must be put in place to ensure the application is continuously monitored for accuracy and reliability. Trained personnel can spot and report unexpected data anomalies and service interruptions.

Customization

Once an integration solution is rolled out, additional customization requests will inevitably come up. Business processes tend to be fairly dynamic. This Is when reality kicks in. There will almost always be constant requests to make changes to the initial integrations, to make things “just a little better.”

Unless the in-house development team takes into account functionality for allowing end-users to make changes to workflows, customizations will have to be made by the development team. As most companies are very particular about their sensitive internal data, any customization for the in-house developed applications will have to be coordinated between the DevOps teams of both companies. This can be a long-drawn-out process.



Additionally, most development teams are engaged in several in-house projects at the same time, and customization of the data integration might be just one of them. So, these in-house customization requests often have to compete with other projects. As a result, they'll often languish unnecessarily for months unless they're properly prioritized.

Maintenance

In software engineering and development circles, it's pretty common knowledge that maintenance accounts for most of the costs associated with software projects. It's safe to say these costs are even higher for cross-company integrations.

The costs involved are significant. But they only reflect the difficulties in maintaining the custom in-house applications. When the underlying systems are upgraded or completely replaced, applications developed for the original systems will need to be modified, or in some cases completely rewritten.

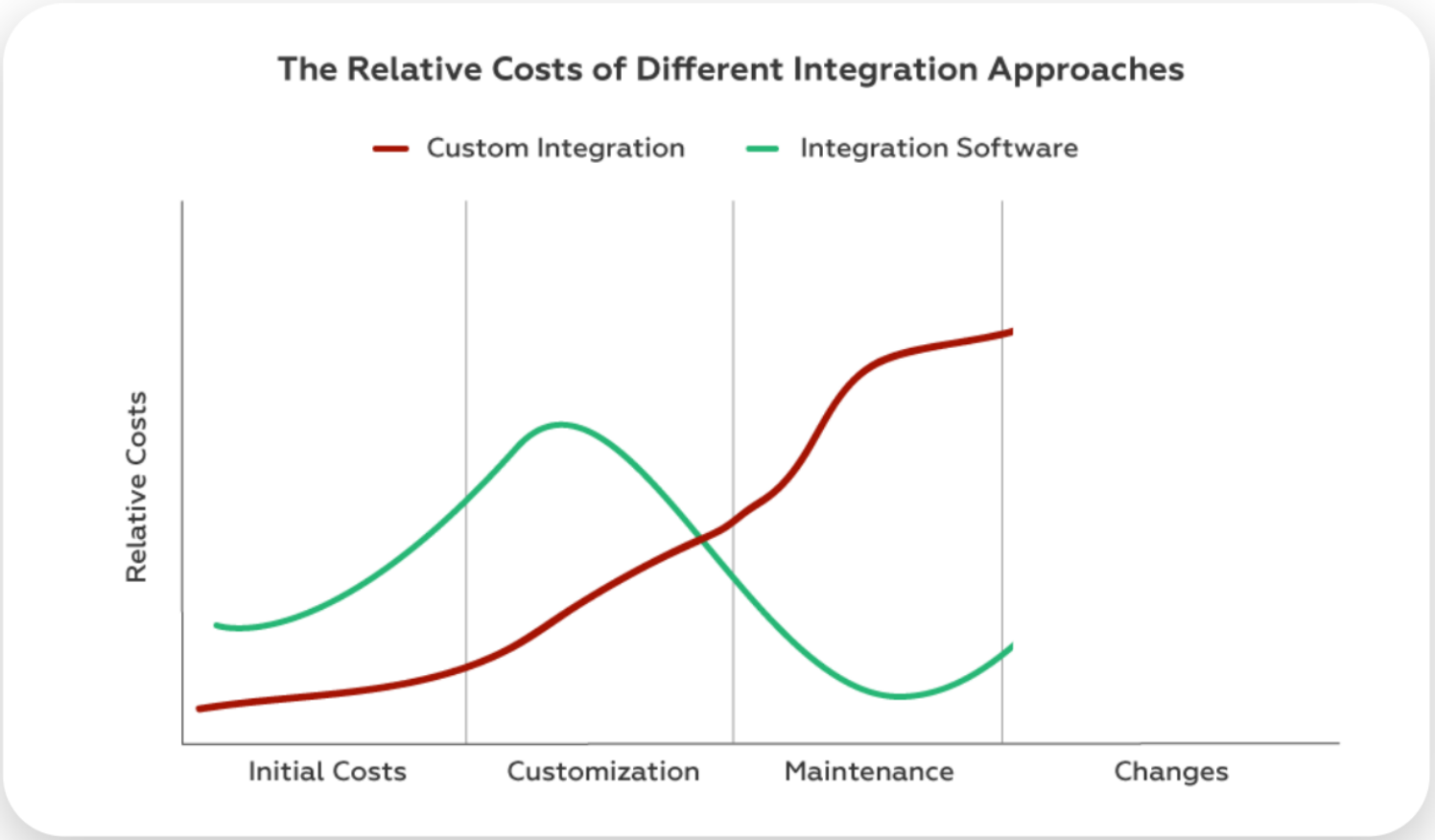
Moreover, the documentation on the original application will have to be maintained and updated whenever changes are implemented. And whenever the original developers or the system maintenance personnel leave, knowledge transfer to new IT personnel becomes a sort of problem itself.

This often creates 'the perfect storm' of issues that dramatically drive up costs and make customization difficult to maintain.

A recent case study of a large corporation that maintains in-house custom integration software revealed the depths of these problems. It runs into maintenance difficulties every year with one of its major partners. Their partner relies on short term contractors to maintain their workflow systems.

It becomes very obvious that there is no turnover between contractors from year to year. Every time this corporation tries to customize or upgrade the integrations, it has to re-educate the new contract development team about the specific features of the integration.

On one occasion, it was even required to deploy its own development team to the partner's site to assist them with the upgrades needed on their systems to continue the functionality of the system. While the integration is essential for maintaining the client's mission, these difficulties with cross-company integration have caused costs to skyrocket.



These costs can be significantly reduced by using the right commercial integration software. These products all come with support agreements that allow the companies to integrate their data and information to leverage the significant expertise of the developers and engineers within the integration software firm.

These experts also have teams responsible for maintaining the software documentation providing needed training. They even work with their clients on troubleshooting specific issues and provide the required support along the way.

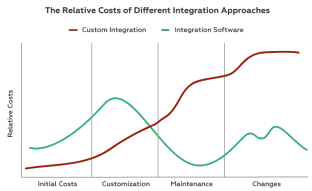
Changes

Business environments are dynamic and change quickly. Companies must fluidly accommodate changes in order to stay competitive. For integration applications developed in-house, this adds an extra layer of complexity to the customization issues.

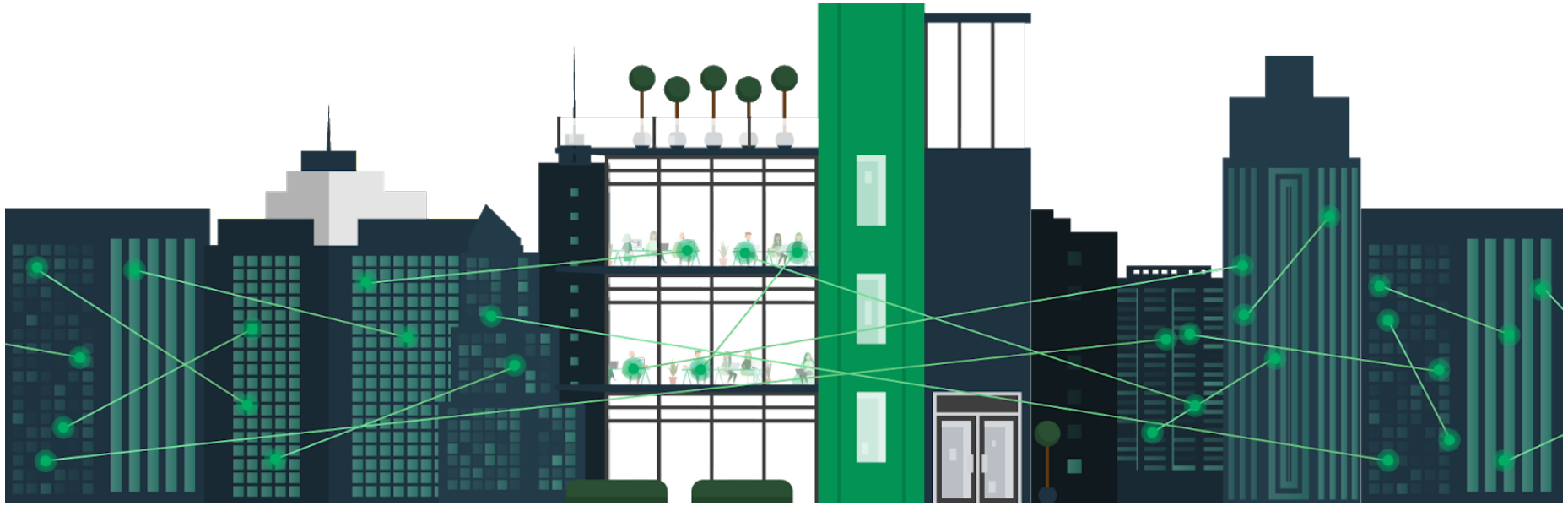
These constant changes have a pretty dramatic effect on any cross-company integration. It's normal to expect configuration changes on either side of the integration. These can include new workflows, permissions, data fields, and new insights from using the existing workflows.

"You know, if we took our existing process and added these two new data points, we could eliminate the need to perform this additional check" is but one example.

Unfortunately, custom integrations are not set up to handle changes so dynamically. Unless a solid change management process has been finely tuned, changes between both end both business and technical coordination.



The use of commercial integration software can significantly reduce difficulties and delays caused by system and process changes. Most of these companies work with their clients to develop seamless processes for handling change requests.



This integration software allows the process owners to define what information gets sent out externally and how incoming messages are processed internally. Changes in the internal company processes will not have an effect on cross-company integration.

This autonomy is an important feature in any such integration setting, and currently, Exalate is the only commercial integration software providing this functionality. Instead of pinpointing specific processes, Exalate allows more flexibility in the integration of systems and allows the systems to be loosely coupled and independent of each other.

Conclusion

When cross-company integration becomes an essential part of optimizing workflows and work management systems, companies are faced with a decision to develop the integration applications in-house or purchase commercial integration software. It's a classical make-or-buy scenario.

Confronted with the up-front costs of enterprise integration software, many corporations are tempted to develop a solution through the use of their own internal development staff. The initial costs seem cheaper, and the development staff is already on-board. It's just another simple development project for them to tackle. Right?

Not so fast. Building your own solution is a huge undertaking, not to be underestimated. While initial costs to start an in-house integration project may seem the cheaper option, many difficulties arise during customization and maintenance that will exponentially rise costs.

Customizations must be done on both ends of the systems. And how will this work across development teams on both sides? When one or more of the underlying systems are upgraded, how will the application be retrofit (if it can be done at all)? Who will be responsible for system documentation and change management? How do you transfer knowledge when one or more of the key developers moves on to another company? The solutions to all of these issues dramatically drive up costs and reduce efficiencies.

Most of these difficulties can be averted by incorporating the right commercial integration software into your project. While the initial upfront acquisition costs are higher, they drop off dramatically in the customization and maintenance phases of the integration.

Commercial integration firms have pricing plans that offer support at all phases of the integration project. They provide documentation support and can provide the best practices for customization.

Additionally, commercial integration software is built with changes to underlying systems in mind and are built to operate independently of changes to these underlying systems

One of the most critical advantages of acquiring commercial integration software, however, is the advantage of integration transparency. Your focus is on optimizing your business, not getting distracted by potentially messy in-house integration projects. Acquiring one of the best-in-breed integration software packages allows you to do just that.

When it comes to the right cross-company integration solution, the following features become the most prominent ones to take into consideration when buying a solution:

- **Autonomy**
  Configurations evolve. How to deal with the situation that a workflow changes on one end. How do you accommodate this in the configuration of the solution and at what stage is such change introduced?

- **Flexibility**
  With cross-company integrations, you never know what future requirements you will encounter.

  For instance, if you have an incident to issue mapping where you have 150 projects on the Jira end (this is an actual case), and new projects are added on the fly on the Jira side. How do you implement an efficient dispatching capability that is easy to maintain (without coding)
- **Reliability**
  Systems go down - that's a fact. You can imagine that your partner is upgrading their Jira. How do you ensure that all the changes are synced in the same order as they happened?

Exalate was built to meet these criteria and to solve the thorniest cross-company integration issues. It allows you to focus on optimizing processes within your company while seamlessly integrating with your external partners exactly where and when you need to integrate.

It gives your company the flexibility, reliability, and autonomy you need to successfully execute cross-company integrations. By solving these cross-company integration issues, it lets you focus on your number one priority; running your business.