

Choosing the Right Integration Approach

October 2020

Choosing the Right Integration Approach



Introduction

Every business has integration between their business applications. In a very small business, that integration may be entirely manual, and in a very large business, there may be dozens of different integration technologies deployed for different needs. Choosing the right integration approach is as important as choosing the right applications to run your business - it's not optional, and it can be a confusing choice. In this document, we'll look at the factors that you should consider when choosing an integration approach and provide some indicative guidelines for integrations in the ServiceNow environment.

Considerations

The right integration approach depends on the job at hand - there isn't a single integration approach that will suit all requirements. As such, there are a list of considerations that you must think about before deciding on the right approach for a specific need. This list includes:

Volume - how much data do you have - and how much do you need to move? What needs to move along with the data? Are there attachments or related records that need to be moved at the same time as the data to give it context or make it meaningful? For example, change requests often need to be accompanied by information about the configuration item being impacted - data which will be in a separate location.

Timing - when does the data need to move? Does it need to move in real-time because it's being used to drive dashboards or processes in which timing is important? Or can the integration execute as a batch job outside of regular business hours? For example, a monthly report might pull data on a weekend morning to minimize performance impact.

Performance - will your data extraction or data load have a performance impact on the systems that you are using? This may be a consideration that drives timing - and may even suggest a multi-stage approach that uses a combination of real-time and batch transfers. For example, ServiceNow can be impacted by large web service data exports since they use the same communication channels as the users.

Purpose - what type of integration do you need? Is this pure data movement? Or are you executing a process across multiple systems/organizations? Or are you just firing a trigger into another tool to launch a remote action? For example, launching a remote-control application or creating an incident record from a chat transcript.

Resilience - how critical is the integration to your business? Is it something that should be monitored on a 24/7 basis to ensure that it's always available? What are the business implications of the interaction not happening? Is there data loss if a destination is unavailable, or does your approach use a "store and forward" model to ensure that the message gets through once connectivity is restored? For example - an MSP losing connectivity with their customers would have a significant impact on their business.

Distribution - How does this data need to be distributed? Is it a one-way transfer, or a bidirectional synchronization? Is it a point-to-point integration, or are there multiple destinations for the data? For example, data may be replicated to a data warehouse and used to train a machine learning tool at the same time.

Data Ownership - Are there concerns around data ownership subsequent to the integration? Once the data has been replicated, how should updates be addressed? Can the destination make updates to the data or is it read-only once replicated? What happens if the source makes an update? Is the record re-replicated?

Manipulation - does the data need to be manipulated before it is replicated? Often this is required when data is moving from one application to another. For example, does it need to be translated to a different language, put into a different format or have certain data stripped from the record for privacy?

Privacy - does the data contain PII information that is subject to privacy regulations? Are you able to keep tabs on how the data is moved around, so that you are able to identify specific data and remove it should that be necessary. For example, GDPR may require that a customer's or an employee's information be removed at their request.

Security - Does the data need to be encrypted once it leaves the source? Should it just be encrypted in transit, or does it also need to be encrypted at rest? What encryption protocols need to be supported for this security? Organizations in regulated industries may not have any choice in this requirement. The specific ServiceNow modules/processes being integrated may impact this too. For example, the HR module has ratcheted down security rules around a lot of it, and maybe integrating it with a non encrypted methodology is not the right answer. The modules used in ServiceNow may determine the privacy and security requirements for their integration.

Auditability - How important is it that you can prove a data transfer took place? Again, companies in regulated industries may need to do this, but so also may service providers who need to be able to demonstrate that they sent a request to one of their customers at a specific date/time to satisfy service level agreements.

Time to Value - How quickly is your integration needed? What is the daily cost of not having the integration in place? Packaged solutions are going to have the edge here, since there will be less development and testing involved - but if your integration requirements are particularly unique, this may not be an option.

Manageability - how will the integration be maintained? Our customers tell us that the annual maintenance of a home-built integration can be up to 60% of the initial effort to develop it. Every time there is a requirement change, or either "end" of the integration is upgraded to a new version, integrations have to be re-built and re-tested to ensure that the integration still works as expected.

Overall Cost - an obvious consideration, but one that is often calculated inaccurately. Ensure that you are looking at the Total Cost of Ownership - as opposed to just the license cost. Factor in development time, testing time, regular maintenance for upgrades and process changes, additional hardware and software requirements, and so on. A TCO calculator [like this one](#) from Perspectium can help here.

So there's a lot to consider when you're looking at an integration. But even THAT list isn't exhaustive. Other questions that you should be asking include:

1. Are you consolidating data, distributing data, archiving data or federating data?
2. Who has the final say on what the data looks like at its destination?
3. How quickly does the transfer of data need to happen? Is it the same in both directions?
4. How critical is the data being moved? What happens if it doesn't get there?
5. Who needs the data, how often, and why?
6. Is the data in the cloud, on-prem, or both?
7. How will access to data save costs or improve efficiencies?

Potential Integration Approaches

To make it easier to consider the many potential integration approaches, we have grouped them into categories - defined below:

Perspectium – integration as a service technology delivered by Perspectium's DataSync and ServiceBond products. These products are described in detail on our [website](#).

iPaaS - integration Platform as a Service tools deliver a platform on which you can build your own integrations, and often come with a vast array of connectors. Vendors in this space include Informatica, Mulesoft, Boomi and Jitterbit. It's important to note that these solutions just deliver a platform - the creation and management of the actual integration is left to the customer. These tools are often used by large organizations where the data warehouse team makes the integration buying decisions - since they are looking for a single solution that will work with most data from most applications.

DIY - Do-It-Yourself approaches typically rely on web service interfaces written in-house by a "spare" developer to perform an integration task. Since they are created for a specific task, any changes to that task usually require a re-write and re-test of the (typically undocumented) integration. This approach is often used by organizations without a lot of experience in integration, due to the escalating ongoing costs and the solution inflexibility that soon become apparent.

Integration Hub - This is ServiceNow's own integration technology, and is really just a re-packaging of the web-services approach that they have been delivering

for years. This product is delivered in the same “style” as iPaaS solutions - in that a platform is provided, and it is up to the customer to create the actual integration on top of that with the tools provided. This approach is good for trigger-based integrations like chat and remote control, and for when you want to keep iPaaS inside ServiceNow.

ETL Tools - Extract, Transform, Load tools are one of the older forms of integration, delivering data to data store transfers of data. They work well in situations where application business rules or security requirements don't matter. They are good for staying within (behind) a firewall, when data is not time sensitive. Sample vendors in this space include Pentaho, SAS, Talend and Xplenty.

Swivel Chair - is exactly what it sounds like. These are manual integrations - where a human copies information from one tool into another. There are obvious shortcomings with this approach - but it is relatively cheap, and appropriate for very small-scale integrations.

In an attempt to make your integration decisions easier, we compiled the summary chart on the next page to provide guidance:

	Perspectium	iPaaS	DIY	Integration Hub	ETL tools	Swivel Chair
Volume	Huge	Large	Medium	Medium	Huge	Small
Technology	Native SN app	Web Services/API	Web Services	Web Services	Database	Human
Purpose	Data/Process	Data	Data	Process/Trigger	Data	Manual
Timing	Realtime/Batch	Batch	Batch	Realtime/Batch	Batch	Manual
Performance	No impact	Potential Impact	Potential Impact	Potential Impact	No impact	No impact
Resilience	24/7 monitoring	Platform only	Not provided	Not provided	Not provided	Not provided
Distribution	One-one One-many Multi-step Bidirectional	One-one Bidirectional	One-one	One-one Bidirectional	One-one	Manual
Manipulation	Pre-send	Post-send	Post-send (if any)	Post-send	Post-send	Manual
Privacy	Obfuscation	Varies by vendor	Not provided	Not provided	Not provided	None
Security	Encryption	Encryption	Not provided	Encryption	Not provided	None
Auditability	Available	Varies by vendor	Not provided	Not provided	Varies by vendor	None
Time to Value	Rapid	Extended	Extended	Medium	Extended	Rapid
Manageability	Vendor-managed	Unmanaged	Unmanaged	Unmanaged	Unmanaged	Unmanaged
Cost	\$\$	\$\$\$\$	\$\$	\$\$	\$\$\$	\$

We're here to help with all your ServiceNow integration needs. If you have any questions about how to connect ServiceNow to anything at all, please reach out to us [via email](#) or on our [website](#), and we will be happy to help!