Integrate.io and Salesforce

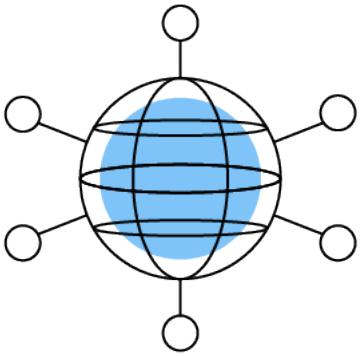
GENERATE A CUSTOMER 360° VIEW WITH INTEGRATE.IO
BI-DIRECTIONAL DATA INTEGRATION

Integrate.io

Table Of Contents

01

HOW THE WORLD GOT CONNECTED



Back in Spring 2000, a new company called Salesforce announced an exciting new feature that they called a Web API[1].

An API is an interface that allows software to share data and processes with other software. The Salesforce Web API took that concept online, creating a way for people to build systems that pulled information directly with the Salesforce web service. It was a game-changer that helped kickstart a wave of digital transformation.

Fast forward 20-plus years. We live in an age of infinite integration, of Everything-as-a-Service, of entire enterprise stacks that only exist in the Cloud.

Companies now face the challenge of keeping all of their systems connected. Proper integration ensures timely access to quality data. Data analytics depends on high-quality information. Well-organized data leads to useful operational insights.

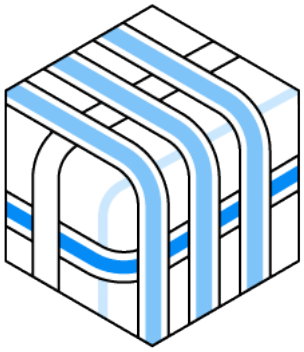
Salesforce has grown to become the world's most popular customer relationship management (CRM) platform. Almost one in every five businesses that use a CRM will rely on Salesforce to find leads, convert sales, and delight customers[2].

Every Salesforce user will tell you that the platform works better when it's connected to your other systems. It's what Salesforce is designed for. And Salesforce itself offers some excellent native support for system integration.

But for a real-life enterprise setting, you'll need something that offers more control over the flow of data between systems. You'll need an automated data pipeline.

02

THE BENEFITS OF AN AUTOMATED DATA PIPELINE



Automated data pipelines use a cloud-based ETL (Extract, Transform, Load) process to take data from one place, reorganize it according to your needs, and move it somewhere else.

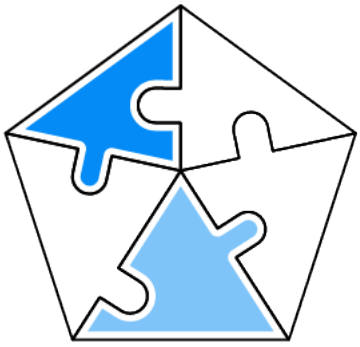
A cloud ETL like Integrate.io offers a number of advantages over direct integration. For instance, with Integrate.io you can:

- Automatically integrate with a wide range of enterprise software
- Create no-code data pipelines with a drag-and-drop interface, even if you don't have any programming skills
- Manage each data source in isolation without impacting the rest of your stack
- Secure all data transactions with Integrate.io's industry-leading cybersecurity measures, including field-level encryption
- Build a transparent data integration process that protects data quality

In this paper, we'll look at how a data pipeline might sit within your organization's data infrastructure, and how it can unlock the full potential of Salesforce.

03

FIVE SALESFORCE INTEGRATION USE CASES



Salesforce is a central hub for data related to customer and sales activity. The Salesforce database is a rich source of powerful analytics insights.

The platform's extensibility means that you can easily build connections between Salesforce and other databases within your stack. These work on a one-to-one basis.

For instance, you might have a connection between Salesforce and your ERP, and a different connection between Salesforce and your billing system.

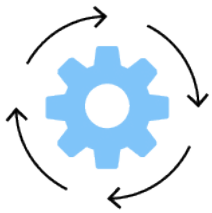
Over time, these one-to-one integrations can become cumbersome. For example, say you have six systems that share information with Salesforce. When you want to make a structural change in Salesforce, you will have to manually rewrite each one of these connections in turn. At best, it's a massive time investment. At worst, it introduces a risk of data corruption or loss.

An automated data pipeline allows you to create a more flexible one-to-many connection for your entire stack. In this model, Salesforce only integrates with a single other application, such as Integrate.

io's ETL. When you need to change something on Salesforce, you only need to reconfigure the Integrate.io integration.

The ETL will then transform all data according to your new configuration. Your existing connections to other systems will still work.

With this kind of infrastructure in place, you can start to create some incredibly ambitious integration projects. Here are five examples of how people use Salesforce and data pipelines to deliver incredible results.



AUTOMATING SALESFORCE ACTIVITY

An activity within Salesforce can often trigger an activity elsewhere if you've got an active integration between Salesforce and your other systems.

Sometimes, circumstances may prevent you from linking Salesforce to another system. You may not have the in-house skills to integrate the two systems, or Salesforce may not support that particular integration. You might also need a data integration process that offers greater visibility so you can identify and rectify any errors.

In this case, you would use an automated data pipeline to create the integration. Typical integrations include things like:

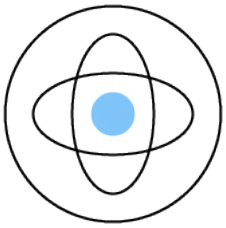
- **Marketing**: When a lead converts, you'll need to update their status on your marketing automation platform. A data pipeline can help you sync Salesforce with tools like Marketo and Hubspot.
- **Ordering**: Sales usually involve some form of delivery process. A data pipeline can trigger automatic order creation on an ERP such as Netsuite or Shipstation.
- **Invoicing**: Sales activity also generates an invoicing process, which means issuing the invoice and reconciling payment. Data pipelines can take data from Salesforce or create new events on payment systems such as Stripe, bill.com, or Invoiced.
- **Finance**: New revenues need to be reflected in your accounting systems, so Salesforce can send this data to Quickbooks or Xero.

When you're using an ETL to automate these processes, the data goes through three stages:

1. **Extract:** First, the ETL requests data from the system on which the data originates. This could be Salesforce or another system in your stack, such as a marketing automation platform.
2. **Transform:** The ETL now validates and amends the data so that it's compatible with the destination database. On a platform like Integrate.io, you can control this transformation without needing to write complex code.
3. **Load:** When the data is ready to go, it's transmitted to the destination system.

For example, let's say you're automating the invoice process on Stripe. The ETL will monitor the Salesforce database for changes, such as a new sales record. When it detects an event, it will export all information about that sale.

After that, it checks that the data is valid, then transforms into a format that Stripe understands. Finally, it loads the data into Stripe, which triggers a new invoicing process.



BUILDING A 360° CUSTOMER OVERVIEW

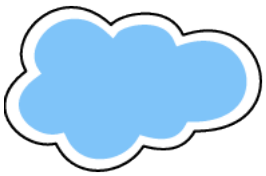
A sale is just one part of the sales cycle. You interact with customers at so many other stages along the way, from marketing at the lead generation stage to post-sales customer service.

If you want to understand your customers, you need to consolidate all of their information into a single customer profile. That requires you to bring data together from systems like:

- **Lead generation**: Your website and social media will offer insight through tools like Google Analytics that you can help to shape your marketing efforts. These channels might also help generate leads that pass to your marketing team.
- **Marketing:** The marketing team uses automated platforms like Hubspot and Marketo to qualify leads and assign them personas. These platforms then guide people along the buyer journey, eventually converting them to sales.
- **Sales**: Salesforce manages most of the sales-related data, but there will also be some useful analytics from e-commerce platforms such as Shopify.
- **Fulfillment:** Whether it's a physical process or a digital service, sales will generally result in some kind of fulfillment process. Data from this process (including information about returns) will be stored in a fulfillment system like ShipStation or an ERP like Netsuite.
- **Invoicing:** After the order is fulfilled, you'll create an invoice for the customer and for your records. Systems such as Invoiced and bill.com might handle this data.
- **Payment:** Most customer payments will pass through a digital payments processor. This processor will hold detailed payment information that you'll need to reconcile on an accounting system, such as Stripe.
- **Customer service:** You might use systems like Zendesk, Intercom, or Freshdesk to deliver after-sales support to customers.

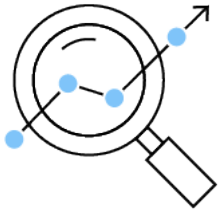
Salesforce is a powerful CRM for tracking customer service actions, although some businesses might prefer an analytics-oriented solution such as Base CRM.

- **Renewals**: If you offer a subscription service, you may use something like Recurly or Zuora to manage ongoing payments. Your own internal analytics reports might flag customers who are about to enter a new buying cycle. As you see, customer data can possibly exist across up to a dozen disparate systems. Each one of these systems holds a single piece of the puzzle, but you can only understand your customers when you see the whole picture. Data pipelines can provide you with that overview in one of two ways:
- **Salesforce consolidation:** Configure your data pipeline to automatically update your CRM with any new customer activity. In this model, Salesforce becomes a Single Source of Truth for your enterprise. All other systems are verified against the contents of the CRM database.



- **Data warehousing:** Bring data from all of your systems together in a single repository, such as a data warehouse. Here, the warehouse is the Single Source of Truth, which means that all systems including Salesforce can be verified against the warehouse records.

In either case, the goal is to create a single customer record that offers a 360° view. Every team can work off this record, including sales, marketing, and operations.



ENABLING ANALYTICS AND TESTING WITH SALESFORCE DATA

Moving away from sales and operations, your analytics team plays a vital role in strategy and high-level decisions. The analytics team has an unquenchable appetite for data, and Salesforce is a banquet of important information.

The easiest way to make Salesforce data available to the analytics team is to use an automated data pipeline. With the right ETL/ELT platform, you can move data into repositories such as:

- **Amazon Redshift**
- **Google BigQuery**
- **Snowflake**
- **MySQL relational database**

The automated approach offers a workaround for one common analytics roadblock: Salesforce's governor limits. These limits put a cap on daily database activity, with the limit depending on your Salesforce membership level.

Real-time database queries of the kind used by the analytics team can quickly eat up your transaction allowance, especially if your users are active on the system (which, ideally, they should be.) Automated pipelines offer a more efficient approach. The pipeline runs a single batch query on a regular schedule, allowing the ETL to pull the information it needs without consuming all of your available transactions.

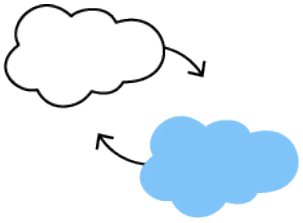
Another advantage of an ETL is that you can use it to mask or otherwise obfuscate personally identifiable information (PII). Moving PII between databases can raise regulatory issues, especially if you're moving across state or national borders. However, an ETL process can remove or hide any PII without impacting the end quality of your analytics.

The same applies to data for software testing. Your IT team may sometimes require a copy of your customer database to help develop and test new applications. For security reasons, you should always remove PII from any copies of data that you provide to them, even if you're working with an in-house team.

USING SALESFORCE AS A SINGLE SOURCE OF TRUTH

Salesforce is such a powerful and diverse platform that you choose to use it as a central data repository. Instead of sending Salesforce data to a warehouse, you might choose to integrate other applications with Salesforce. Salesforce becomes the home of your master data, or your Single Source of Truth.

This is especially useful if you're using Salesforce as your main CRM. The majority of customer information will already exist in the Salesforce database, so other systems can simply access that database for the latest values. If customer information is updated on another system, your integration process can push that through to Salesforce.



This strategy can be a quick win, especially for small businesses experiencing rapid growth. It ensures data availability and reduces inconsistencies, and does so without requiring any major new software expenditure.

But there are some technical issues to consider with this approach, such as:

- **Compatibility**: Salesforce may not be directly compatible with all other systems in your stack.
- **Ease of use:** You may need to write some code to allow other systems to read directly from the Salesforce database.
- **Security**: Systems behind your firewall may need to communicate over the open internet with Salesforce's cloud database.
- **Data management:** As mentioned previously, you'll need to batch and schedule data tasks so that you don't run into Salesforce's transaction limits.

An automated pipeline can solve most of these issues. Integrate.io also offers an easy no-code interface, so you can connect to Salesforce without needing to write any software.

SALESFORCE TO SALESFORCE INTEGRATION

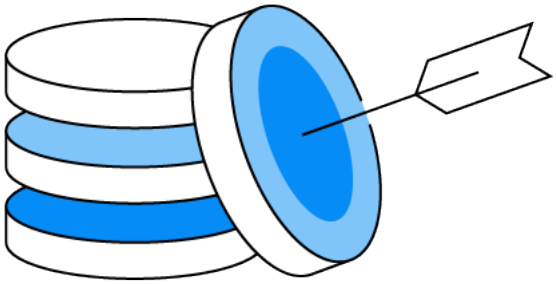
Orgs are a fundamental Salesforce concept. Your org is the digital representation of your organization within Salesforce's database. Theoretically, each company is a single org, all users are members of that org, and all customers exist within the org's database. In practice, it doesn't always work out like this. Large enterprises might own multiple Salesforce orgs, or a merger could leave you with two parallel orgs. Or you may have a situation where two orgs from different businesses want to collaborate on a large project.

Whatever the reason, there are situations where you may need to move data from one org to another. There are three ways of doing this:

- **Direct Salesforce-to-Salesforce (S2S) integration:** Salesforce supports the creation of S2S integrations — you'll find the option on the contacts tab in the platform. You can choose which objects and fields to share, although objects are limited to: Accounts, Attachments, Cases, Case Comments, Contacts, Leads, Opportunities, Opportunity, Products, Tasks, and Custom Objects.
- **Integration via data warehouse:** Data warehouses can provide an intermediate stage for any integration. It works the same way as any other integration: You extract data from the target system and then load it to the destination. This approach is best if you want to store Salesforce data externally for analytics purposes.
- **S2S integration via data pipeline:** A data pipeline provides a best-of-both-worlds approach. Your pipeline can offer a seamless transition between databases, with no need for complex configurations. Plus, you have full control over the data you extract and how it's processed at its destination.

04

SALESFORCE INTEGRATION BEST PRACTICES



A Salesforce integration project is not easy. You'll be working with a huge database that's subject to live read and write transactions. There are a lot of potential pitfalls to watch out for, such as:

- Bad data getting into your Salesforce database.
- Incomplete Salesforce data reaching your data warehouse
- Conflicts between customer records held on Salesforce and other platforms
- PII and other sensitive data being exposed during transit
- Connectivity issues causing data loss or corruption.

All integrations should happen within the framework of a good data governance policy. You should also follow integration best practices at every step of the journey. Here are a few of the most important rules to bear in mind.

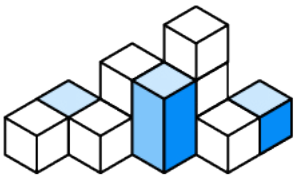
1. Establish metrics for success before you begin.
2. Make your architecture scalable.
3. Always know your data.
4. Have the right support before you begin.
5. Have robust error-handling protocols.
6. Start small and expand.

ESTABLISH METRICS FOR SUCCESS BEFORE YOU BEGIN

What's the underlying business need for integration? There are lots of potential benefits from an integration project, such as:

- Improving analytics accuracy
- Providing a detailed customer view to departments like sales and marketing
- Eliminating work arising from manual data processes
- Meeting security or compliance requirements for data transfers
- Creating data backups or a Single Source of Truth

Your project has a better chance of success if you know in advance what you're aiming for. Talk to all major stakeholders and ask them what they would like to see from a Salesforce integration.



MAKE YOUR ARCHITECTURE SCALABLE

When you know your goals, you'll be better equipped to start thinking about your architecture. The main things you'll need to consider here are:

- What data will flow in and out of Salesforce?
- What other systems are involved?
- What other major data repositories are involved (such as a data warehouse or data lake)?
- How frequently should data transfers occur?
- When does data cross barriers, such as transitioning from behind your firewall to the internet?
- What process will drive all data transfers between systems?

When you're mapping all of this out, you need to think about your future needs. You may want to add new systems to give you new capabilities, or you may choose to swap out one system for a better alternative.

You'll also need to think about how your needs will change as you grow. Salesforce scales up well for most growing businesses, but the rest of your infrastructure may not. Rather than having complex dependencies between systems, try to have a modular structure that makes it easy to swap out individual data sources. This will allow you to make changes without needing to rethink your entire architecture.

ALWAYS KNOW YOUR DATA

Data quality is the one thing you can't compromise on, especially when you're dealing with customer information. This responsibility implies two things. First, you need to design your integrations to ensure consistent quality. Next, you need an oversight process to detect and rectify failures.

The key to quality is knowing your data. You need to have a deep understanding of the information structure within Salesforce. Speak to your Salesforce admin if you're in any doubt.

You'll also need to know everything about your Salesforce org. Different teams may have different approaches to data usage, and this can create inconsistency on a deep level. You'll need to know exactly how data moves through Salesforce before you can start planning your project.

It's also essential to know the data structures within all of your other systems. This will have an enormous impact on your overall architecture. For instance, if you use an ETL to integrate data, you'll map fields from your target database onto your destination database. To get this right, you'll need to understand the precise relationship between both sources.

HAVE THE RIGHT SUPPORT BEFORE YOU BEGIN

A Salesforce integration doesn't have to be a highly technical project. A person with fair-to-middling IT skills can, in some circumstances, set up a data flow between Salesforce and another system.

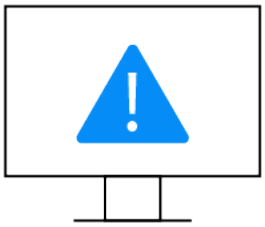
But projects can quickly get more complicated than anticipated. If you're doing anything other than an out-of-the-box integration, you'll need people with the following skills:

- **Database management:** An integration is fundamentally about taking information from one database and moving it to another. You'll need someone who understands SQL and the Salesforce database language, SOQL.
- **APIs and cloud computing:** The API makes integration possible by creating a secure layer between each system. You, or someone on your team, may need to know how to call the Salesforce API, as well as the interface for your target system.
- **Coding:** Automating an integration requires some knowledge of how to write software. This may be a short batch script that you set to run automatically. In some instances, you may need to build a more complex application.
- **Security:** Any movement of data poses a degree of risk. If you're working on a complex integration, you'll need to collaborate with someone who can spot any security implications before they arise.
- **Compliance:** Security's counterpart, compliance, means that you'll have to watch out for potential breaches of the organization's data governance framework. This helps you stay in line with any possible legal issues arising from moving data.

Depending on the complexity of the project, you may only need one IT person to help out, or you could require a whole team. Scope out the project and make sure you have the right resources. It's better to have a team in place before you hit a snag, rather than after.

An automated data pipeline can remove the need for much of this additional expertise. For instance, Integrate.io's no-code interface

means that you won't have to write software or worry about API calls. Plus, Integrate.io's ETL offers robust security with field-level encryption, and it can help you stay compliant.



HAVE ROBUST ERROR-HANDLING PROTOCOLS

Errors happen in data integration. There are some problems that you simply can't avoid, like:

Incorrect mapping of a target database to a destination database

Illegal values or corrupt data cause an exception

- Data gets corrupted in transit
- Destination database is offline
- Connection error interrupts data transfer
- Scheduled job fails to trigger

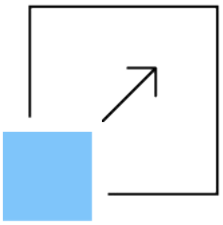
These are just some of the many things that can go wrong when you're moving data. But the goal is not to build an error-proof system. The goal is to create error-handling protocols that detect and resolve problems as they arise.

Salesforce has excellent tools for recovering from errors. Most other systems also help troubleshoot problems. But none of them can detect data errors that occur in transit. For example: Imagine that you send a client record from Salesforce to Redshift, but the record is corrupted in transit. Salesforce doesn't know there's any issue after data export, and Redshift works with the data you provide. Neither system will flag any kind of error.

This is why it's useful to have visibility in your data transfer. An automated pipeline allows you to monitor all extractions and loads, plus it can log details of transformations. If your integration passes through a data warehouse, then you'll have a backup copy of the affected data.

Whatever approach you take, it's essential that you have:

- Active monitoring for data integration issues
- Regular quality audits on all integrated databases
- A person who has ownership of any flagged data quality issues
- Adequate resources to rectify any errors
- Protocols for dealing with data loss or conflicting records
- Find errors as soon as possible, and then fix them.



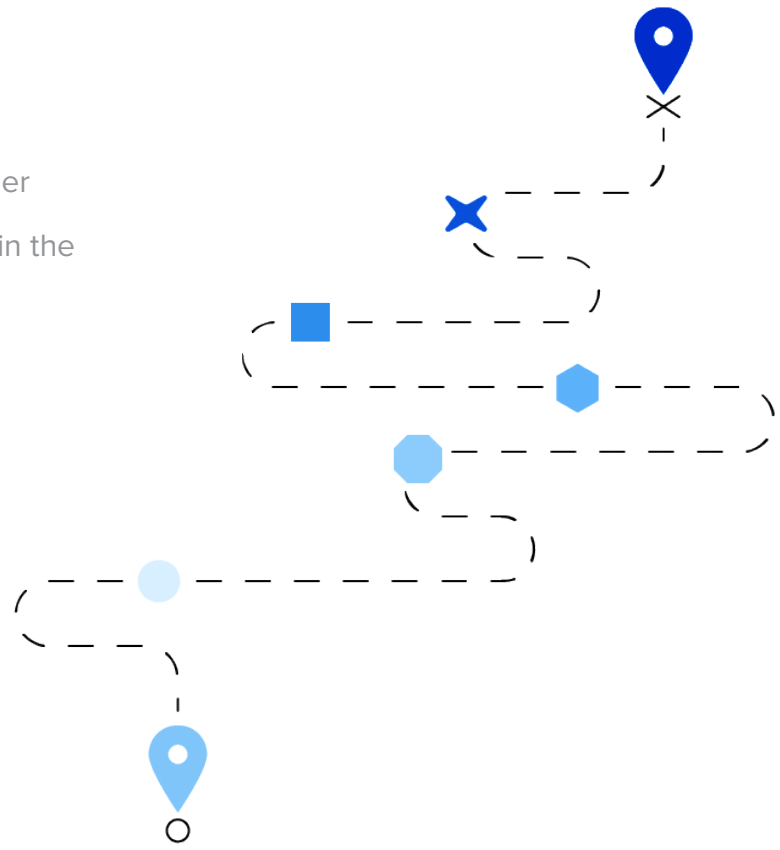
START SMALL AND EXPAND

Due to the complexity of integration projects, it's best to follow a path of gentle evolution. Build on each of your successes while learning from your mistakes.

Talk to your project stakeholders and identify your roadmap for the integration project. It's a good idea to start off with some easy wins so that you can start to build a degree of project momentum. You may end up with a plan that looks like this:

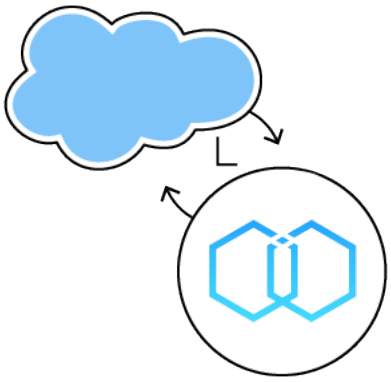
- **Phase 1:** Automate Salesforce data backup to the warehouse.
- **Phase 2:** Integrate Salesforce data with customer information from other systems and store integrated data in the warehouse.
- **Phase 3:** Trigger automated events on the fulfillment and billing systems.
- **Phase 4:** Automatically update marketing automation based on events in Salesforce.

This is one example. Whatever your schedule, choose one that allows you to prove the fundamental concepts early on and then build from there.



05

CREATING YOUR FIRST SALESFORCE INTEGRATION WITH INTEGRATE.IO IN UNDER FIVE MINUTES



Planning your data infrastructure can be time-consuming. As per the best practices above, you'll need to work with others to devise a scalable strategy that ensures future success.

But the actual process for creating an integration couldn't be easier. Let's look at a typical integration: copying Salesforce data to an Amazon Redshift data warehouse for analytics purposes.

There are three elements in this structure:

- Target database: Salesforce
- Destination database: Amazon Redshift
- Data pipeline: Integrate.io

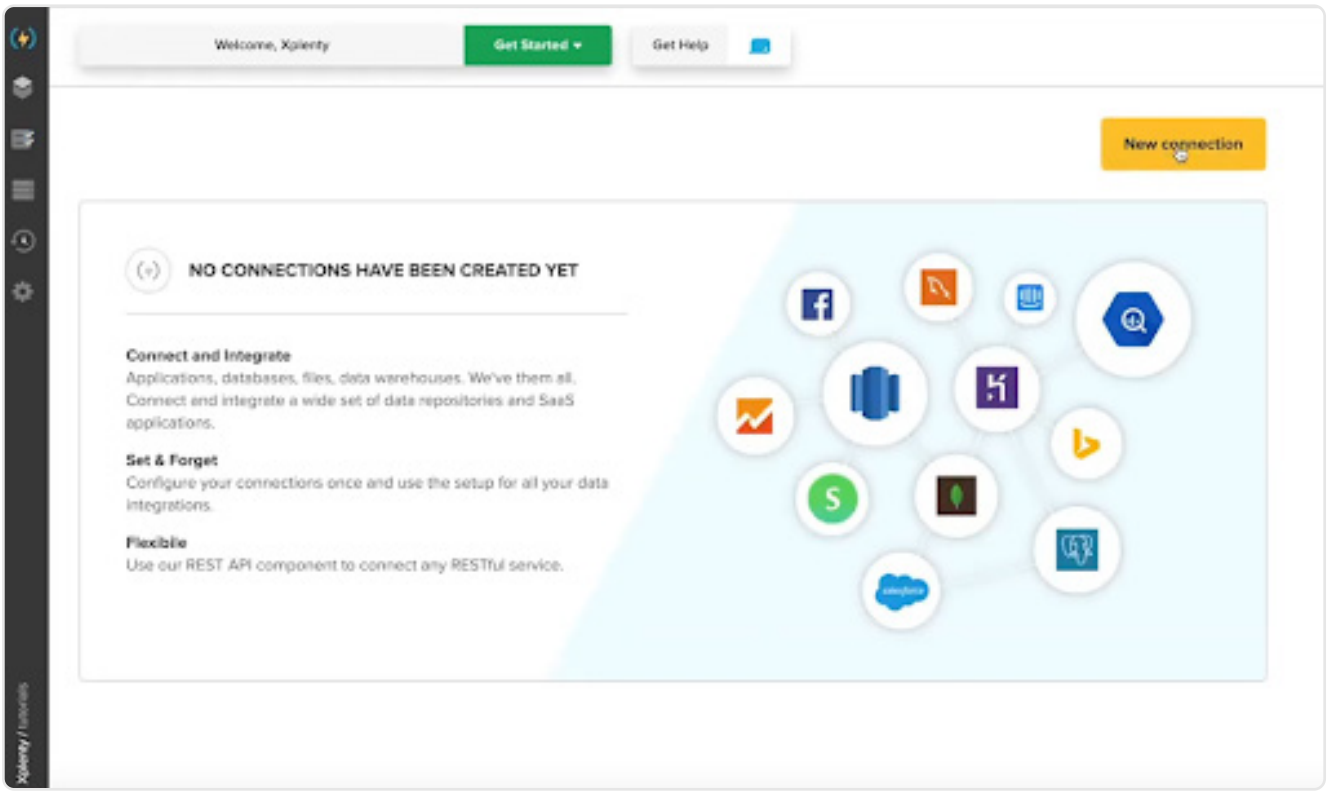
Here's how to set up this integration from scratch. It should take no more than five minutes, even with zero coding experience.

CONNECT INTEGRATE.IO TO BOTH DATABASES

First, you'll need to log into the Integrate.io admin console and give Integrate.io access to both databases (image ↓).

Click the New Connection button. Select Salesforce from the list and click Authenticate. Salesforce will ask if you want to grant access to Integrate.io. Click Allow, and you've completed the first part.

Now you'll need to create a New Connection for Redshift. You'll need the hostname, port, and region for your cluster, as well as your username and password. Once you've entered these details, click Create Connection.

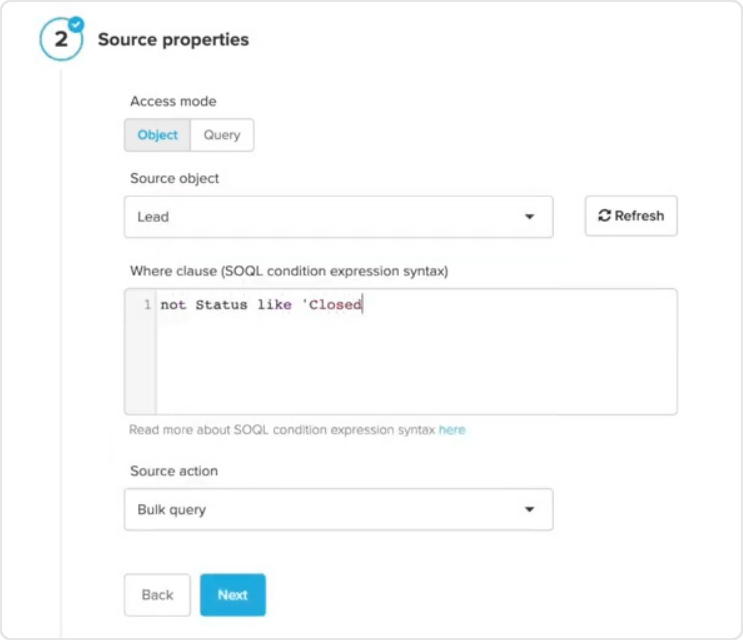


CREATE AN EXTRACTION PACKAGE

Next, go to Packages and click New Package. Give your pipeline a name and click Create Package.

Next you'll need to add both sources. Click on Add Component and choose Salesforce from the menu. In Source Properties, you can select which objects you want to extract, and select individual fields within the object. If you're familiar with SOQL, you can write more detailed queries to extract data (image ↓).

Click Refresh to see the results of your query and to confirm that you're pulling through the data you need. When you're finished, click Save to create your new package.



APPLY TRANSFORMATIONS

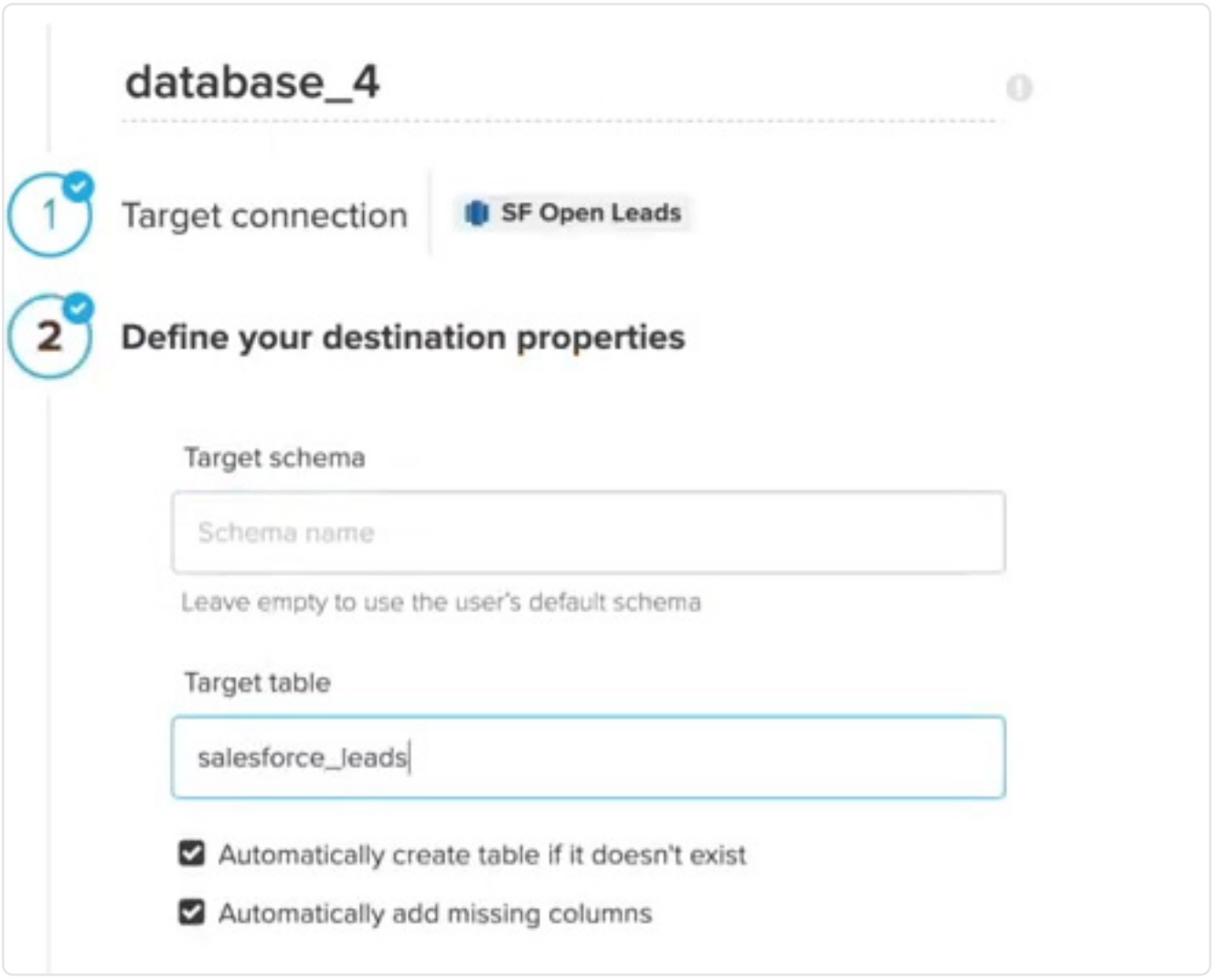
Click on the package in Integrate.io's drag-and-drop editor. You'll find a menu of transformations that you can apply to the Salesforce data while it's in transit.

For instance, if you want to sort all of the data, you can apply a SORT transformation. Simply choose Sort from the menu, apply your criteria, and click Save.

You can deploy multiple transformations in your ETL workflow. Add as many as you need in the editor until your data is in the right format.

ADD YOUR DESTINATION

In the drag-and-drop editor, click on the final transformation in your workflow. This time, go to the Destinations section of the menu and choose Redshift.



Choose the destination table within your Redshift database. You'll also be able to choose what happens to the incoming data, with choices like:

- Append (add to the end of the current table).
- Overwrite (either truncate or delete the current table and insert new data).
- Merge new data with the current table, either through deletion or updating.

You'll also find some advanced options, including an SQL editor for advanced queries.

Once you're finished, click Save.

VERIFY AND RUN THE PACKAGE

When you've completed your data pipeline, verify that everything is working correctly. If you don't get any errors, you're good to go. Now click Run to see your data pipeline in action. You'll need to choose a cluster on which to run. If a cluster isn't available, you can create a new one (image ↓).

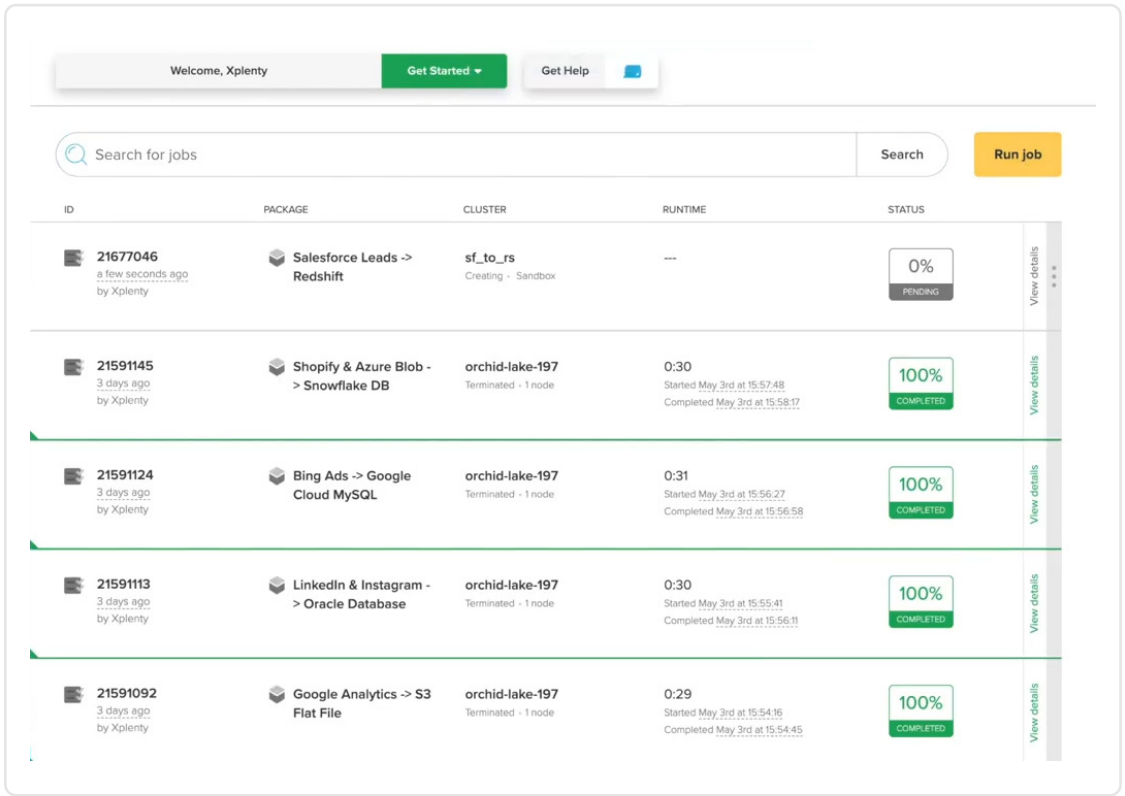
Your job will now run automatically. Integrate.io will extract data from Salesforce, transform it according to your specifications, and load it into Redshift.

CHECK THE OUTCOMES

You can confirm that everything ran smoothly by accessing Redshift and taking a look at the destination table. You should see that all of your data from Salesforce is now available.

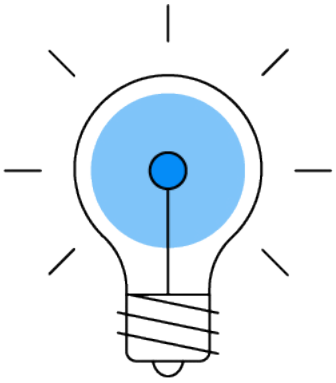
If you applied any transformations, make sure that the new data is organized in the way you specified. If you need to change anything, you can reopen the package in Integrate.io and make amendments.

After only a few minutes of work, you have a functioning Salesforce integration that can run on any schedule you choose.



06

CONCLUSION



Salesforce is an extraordinary platform for sales enablement, relationship management, and other vital customer functions. It's also a goldmine of useful data that your analytics team can turn into insights. These insights will help shape your strategy and ensure future success.

In addition, Salesforce works best when it's integrated with other enterprise systems. They set the ball rolling on large-scale integration back when they launched that first API at the start of the century. Today, it's one of the most versatile and intraoperative platforms in the enterprise space.

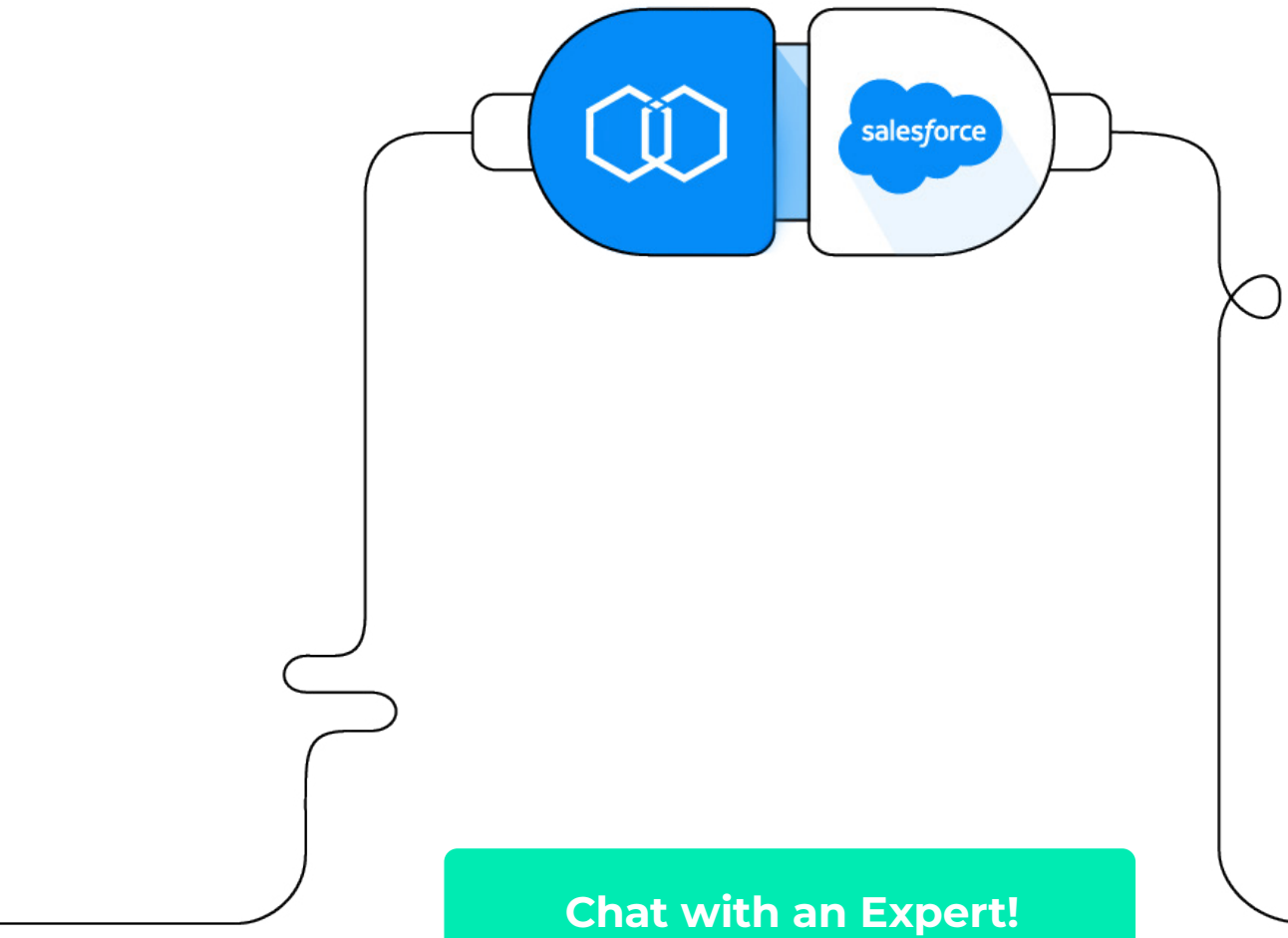
With the right infrastructure, your Salesforce org can become the engine of an unstoppable growth machine. Integrate.io can not only provide you with that infrastructure, but it will allow you to grow and evolve with ease.

We hope that you have enjoyed reading this whitepaper.

We hope that you have enjoyed reading this whitepaper. If so, be sure to check out our other Whitepapers related to "Leveraging No-code Opportunities in Data Pipelines".

Learn more about how Integrate.io can help

Make your data work for you!



Chat with an Expert!