1. Lightning Platform API Basics
2. Get to Know the Salesforce Lightning Platform APIs

# Get to Know the Salesforce Lightning Platform APIs

**Note**

**Attention, Trailblazer!**

Salesforce has two different desktop user interfaces: Lightning Experience and Salesforce Classic. This module is designed for **Salesforce Classic**.

You can learn about switching between interfaces, enabling Lightning Experience, and more in the Lightning Experience Basics module here on Trailhead.

## Learning Objectives

After completing this unit, you'll be able to:

- Describe the benefits of the API-first approach to development.
- State use cases for REST API, SOAP API, Bulk API, and Streaming API.
- Name the two types of API limits and describe how they're calculated.

## API First at Salesforce

Ahoy, matey! Are you ready to set sail on the high seas in search of the perfect Salesforce API for the integration you're building? Well, captain, grab your eye patch and find a parrot. You're about to navigate the Lightning Platform waters and learn all about our APIs.

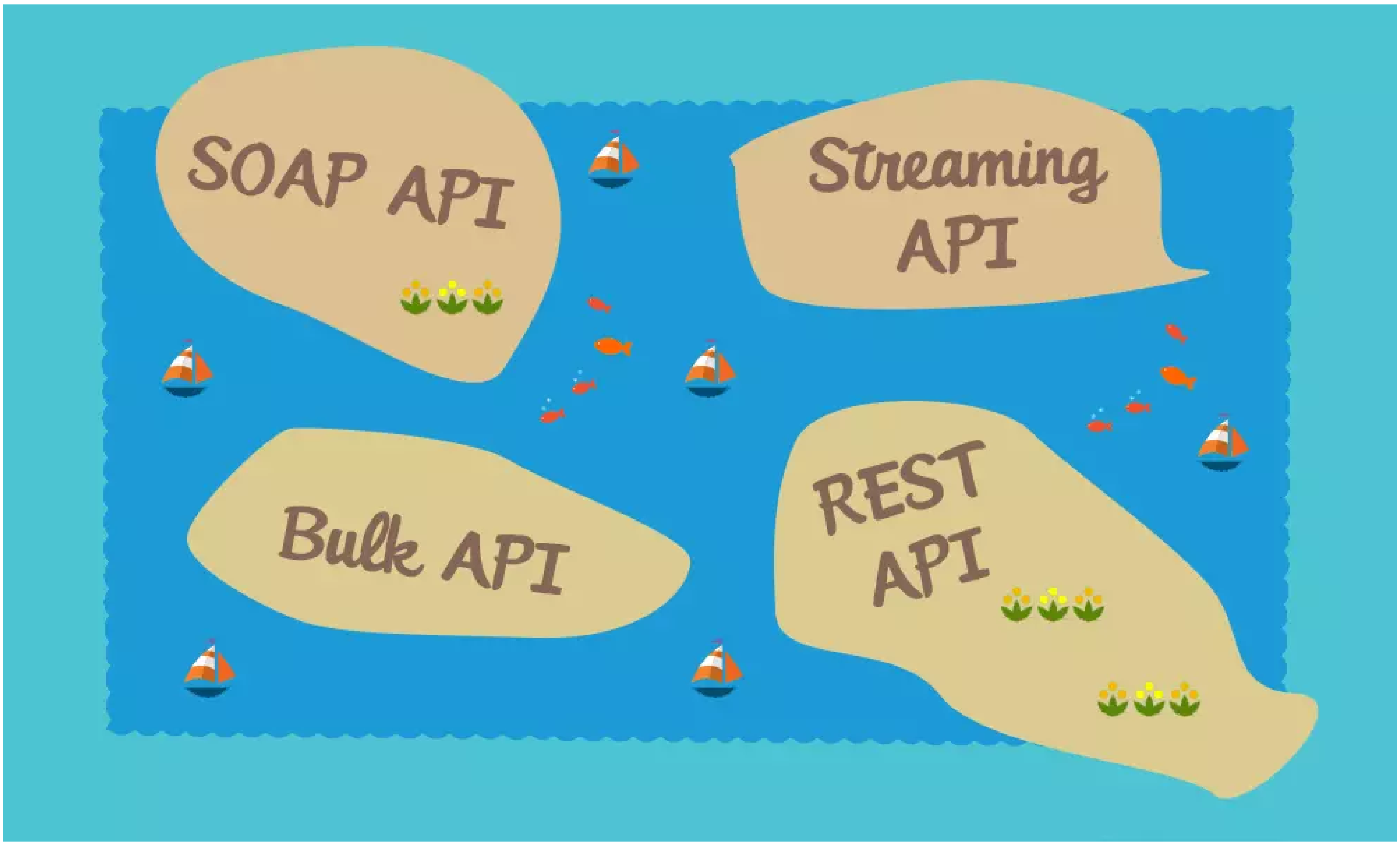
The Salesforce API landscape is as vast as the ocean blue. That's because Salesforce takes an API-first approach to building features on the Salesforce Platform. API first means building a robust API for a feature before focusing on designing its UI. This approach gives you, the Salesforce developer, flexibility to manipulate your data however you want.

Salesforce knows that its customers and partners are always thinking of new ways to extend Salesforce functionality and exciting apps to build for the AppExchange. Providing a comprehensive toolbox for developing on the platform is a top priority. This approach also lets Salesforce build UIs on top of the APIs, ensuring that the behavior is the same between them.

Think of this module as your API first mate. Together we'll walk through some general API info, perform a survey of Salesforce's API suite, and dive into using a few common APIs. All this information will equip you with the knowledge you need to choose the right API for your project.

## Salesforce Data APIs

In the sea of Salesforce APIs, there's a key archipelago of commonly used APIs that we focus on in this module. They are REST API, SOAP API, Bulk API, and Streaming API. Together they make up the Salesforce data APIs. Their purpose is to let you manipulate your Salesforce data, whereas other APIs let you do things like customize page layouts or build custom development tools. You can use other Salesforce APIs to manipulate subsets of your Salesforce data, too. For example, Analytics REST API focuses on Analytics. But these four APIs apply broadly across the spectrum of core Salesforce data.



### REST API

REST API is a simple and powerful web service based on RESTful principles. It exposes all sorts of Salesforce functionality via REST resources and HTTP methods. For example, you can create, read, update, and delete (CRUD) records, search or query your data, retrieve object metadata, and access information about limits in your org. REST API supports both XML and JSON.

Because REST API has a lightweight request and response framework and is easy to use, it's great for writing mobile and web apps.

### SOAP API

SOAP API is a robust and powerful web service based on the industry-standard protocol of the same name. It uses a Web Services Description Language (WSDL) file to rigorously define the parameters for accessing data through the API. SOAP API supports XML only. Most of the SOAP API functionality is also available through REST API. It just depends on which standard better meets your needs.

Because SOAP API uses the WSDL file as a formal contract between the API and consumer, it's great for writing server-to-server integrations.

### Bulk API

Bulk API is a specialized RESTful API for loading and querying lots of data at once. By lots, we mean 50,000 records or more. Bulk API is asynchronous, meaning that you can submit a request and come back later for the results. This approach is the preferred one when dealing with large amounts of data. There are two versions of Bulk API (1.0 and 2.0). Both versions handle large amounts of data, but we use Bulk API 2.0 in this module because it's a bit easier to use.

Bulk API is great for performing tasks that involve lots of records, such as loading data into your org for the first time.

### Streaming API

Streaming API is a specialized API for setting up notifications that trigger when changes are made to your data. It uses a publish-subscribe, or pub/sub, model in which users can subscribe to channels that broadcast certain types of data changes.

The pub/sub model reduces the number of API requests by eliminating the need for polling. Streaming API is great for writing apps that would otherwise need to frequently poll for changes.

## API Access and Authentication

You don't need a treasure map to access Salesforce APIs. All you need is a Trailhead Playground or an org on one of the following editions: Enterprise Edition, Unlimited Edition, Developer Edition, Performance Edition, or Professional Edition (with an add-on). Make sure that you have the "API Enabled" permission, and you're ready to start integrating.

All API calls, except for the SOAP API `login()` call, require authentication. You can either use one of the supported OAuth flows or authenticate with a session ID retrieved from the SOAP API `login()` call. Check the developer guide for your API of choice to get started.

## API Limits

Any captain worth his or her salt knows when to put limits on the crew for the good of the ship. If the captain lets the sailors drink grog all day long, nothing gets done. Similarly, Salesforce limits the number of API calls per org to ensure the health of the instance. These limits exist to prevent rogue scripts from smashing our servers into driftwood. They don't get in the way of your everyday work. Nonetheless, it's a good idea to get familiar with them.

There are two types of API limits. Concurrent limits cap the number of long-running calls (20 seconds or longer) that are running at one time. Total limits cap the number of calls made within a rolling 24-hour period.

Concurrent limits vary by org type. For a Trailhead Playground, the limit is five long-running calls at once. For a sandbox org, it's 25 long-running calls.

Total limits vary by org edition, license type, and expansion packs that you purchase. For example, an Enterprise Edition org gets 1,000 calls per Salesforce license and 200 calls per Partner Community license. With an "Additional API Calls" pack, that same Enterprise Edition org gets an extra 10,000 calls. Total limits are also subject to minimums and maximums based on the org edition, but we won't get into that here. If you want to know more, check out the API Request Limits link in the Resources section.

You have several ways to check your remaining API calls. You can view them in the API Usage box on the System Overview page. From Setup, enter `System Overview` in the Quick Find box, then select **System Overview**. You can also set up notifications for when your org exceeds a number of API calls that you designate. To do so, from Setup, enter `API Usage Notifications` in the Quick Find box, then select **API Usage Notifications**.

When using REST or SOAP API, the LimitInfoHeader response header gives you information on your remaining calls. You can also access the REST API Limits resource for information about all sorts of limits in your org.

## Which API Do I Use?

Choosing the right API for your integration needs is an important decision. Here's some information on our most commonly used APIs, including supported protocols, data formats, communication paradigms, and use cases. Treat this section as a reference you can return to when you're considering which API to use.

Note the four data APIs that we talked about already. We'll be diving into each of them next.

| API Name | Protocol | Data Format | Communication |
|---|---|---|---|
| REST API | REST | JSON, XML | Synchronous |
| SOAP API | SOAP (WSDL) | XML | Synchronous |
| Chatter REST API | REST | JSON, XML | Synchronous (photos are processed asynchronously) |
| User Interface API | REST | JSON | Synchronous |
| Analytics REST API | REST | JSON, XML | Synchronous |
| Bulk API | REST | CSV, JSON, XML | Asynchronous |
| Metadata API | SOAP (WSDL) | XML | Asynchronous |
| Streaming API | Bayeux | JSON | Asynchronous (stream of data) |
| Apex REST API | REST | JSON, XML, Custom | Synchronous |
| Apex SOAP API | SOAP (WSDL) | XML | Synchronous |
| Tooling API | REST or SOAP (WSDL) | JSON, XML, Custom | Synchronous |

### When to Use REST API

REST API provides a powerful, convenient, and simple REST-based web services interface for interacting with Salesforce. Its advantages include ease of integration and development, and it's an excellent choice of technology for use with mobile applications and web projects. For certain projects, you may want to use REST API with other Salesforce REST APIs. To build UI for creating, reading, updating, and deleting records, including building UI for list views, actions, and dependent picklists, use User Interface API. To build UI for Chatter, communities, or recommendations, use Chatter REST API. If you have many records to process, consider using Bulk API, which is based on REST principles and optimized for large sets of data.

## When to Use SOAP API

SOAP API provides a powerful, convenient, and simple SOAP-based web services interface for interacting with Salesforce. You can use SOAP API to create, retrieve, update, or delete records. You can also use SOAP API to perform searches and much more. Use SOAP API in any language that supports web services.

For example, you can use SOAP API to integrate Salesforce with your org's ERP and finance systems. You can also deliver real-time sales and support information to company portals and populate critical business systems with customer information.

## When to Use Chatter REST API

Use Chatter REST API to display Chatter feeds, users, groups, and followers, especially in mobile applications. Chatter REST API also provides programmatic access to files, recommendations, topics, notifications, Data.com purchasing, and more. Chatter REST API is similar to APIs offered by other companies with feeds, such as Facebook and Twitter, but it also exposes Salesforce features beyond Chatter.

## When to Use User Interface API

Build Salesforce UI for native mobile apps and custom web apps using the same API that Salesforce uses to build Lightning Experience and Salesforce for Android, iOS, and mobile web. Build user interfaces that let users work with records, list views, actions, favorites, and more. Not only do you get data and metadata in a single response, but the response matches metadata changes made to the org by Salesforce admins. You don't have to worry about layouts, picklists, field-level security, or sharing—all you have to do is build an app that users love.

## When to Use the Analytics REST API

You can access Analytics assets—such as datasets, lenses, and dashboards—programmatically using the Analytics REST API. Send queries directly to the Analytics Platform. Access datasets that have been imported into the Analytics Platform. Create and retrieve lenses. Access XMD information. Retrieve a list of dataset versions. Create and retrieve Analytics applications. Create, update, and retrieve Analytics dashboards. Retrieve a list of dependencies for an application. Determine what features are available to the user. Work with snapshots. Manipulate replicated datasets.

## When to Use Bulk API

Bulk API is based on REST principles and is optimized for loading or deleting large sets of data. You can use it to query, queryAll, insert, update, upsert, or delete many records asynchronously by submitting batches. Salesforce processes batches in the background.

SOAP API, in contrast, is optimized for real-time client applications that update a few records at a time. You can use SOAP API for processing many records, but when the data sets contain hundreds of thousands of records, SOAP API is less practical. Bulk API is designed to make it simple to process data from a few thousand to millions of records.

The easiest way to use Bulk API is to enable it for processing records in Data Loader using CSV files. Using Data Loader avoids the need to write your own client application.

## When to Use Metadata API

Use Metadata API to retrieve, deploy, create, update, or delete customizations for your org. The most common use is to migrate changes from a sandbox or testing org to your production environment. Metadata API is intended for managing customizations and for building tools that can manage the metadata model, not the data itself.

The easiest way to access the functionality in Metadata API is to use the Salesforce Extensions for Visual Studio Code or the Ant Migration Tool. Both tools are built on top of Metadata API and use the standard tools to simplify working with Metadata API.

- The Salesforce Extensions for Visual Studio Code includes tools for developing on the Salesforce platform in the lightweight, extensible VS Code editor. These tools provide features for working with development orgs (scratch orgs, sandboxes, and DE orgs), Apex, Aura components, and Visualforce.
- The Ant Migration Tool is ideal if you use a script or the command line for moving metadata between a local directory and a Salesforce org.

## When to Use Streaming API

Use Streaming API to receive near-real-time streams of data that are based on changes in Salesforce records or custom payloads. For Salesforce record changes, Salesforce publishes notifications when the changes occur. For custom notifications, you can publish event messages. Subscribers can receive notifications using CometD—an implementation of the Bayeux protocol that simulates push technology. Clients can also subscribe to some types of events with Apex triggers or declaratively with Process Builder and Flow Builder.

Use the type of streaming event that suits your needs.

**PushTopic Event**

Receive changes to Salesforce records based on a SOQL query that you define. The notifications include only the fields that you specify in the SOQL query.

**Change Data Capture Event**
Receive changes to Salesforce records with all changed fields. Change Data Capture supports more standard objects than PushTopic events and provides more features, such as header fields that contain information about the change.

**Platform Event**
Publish and receive custom payloads with a predefined schema. The data can be anything you define, including business data, such as order information. Specify the data to send by defining a platform event. Subscribe to a platform event channel to receive notifications.

**Generic Event**
Publish and receive arbitrary payloads without a defined schema.

### When to Use Apex REST API

Use Apex REST API when you want to expose your Apex classes and methods so that external applications can access your code through REST architecture. Apex REST API supports both OAuth 2.0 and Session ID for authorization.

### When to Use Apex SOAP API

Use Apex SOAP API when you want to expose Apex methods as SOAP web service APIs so that external applications can access your code through SOAP.

Apex SOAP API supports both OAuth 2.0 and Session ID for authorization.

### When to Use Tooling API

Use Tooling API to integrate Salesforce metadata with other systems. Metadata types are exposed as sObjects, so you can access one component of a complex type. This field-level access speeds up operations on complex metadata types. You can also build custom development tools for Force.com applications. For example, use Tooling API to manage and deploy working copies of Apex classes and triggers and Visualforce pages and components. You can also set checkpoints or heap dump markers, execute anonymous Apex, and access logging and code coverage information.

REST and SOAP are both supported.

## Resources

*Integration Patterns and Practices*

*REST API Developer Guide*

*SOAP API Developer Guide*

*Bulk API 2.0 Developer Guide*

*Lightning Platform Streaming API Developer Guide*

API Request Limits

## Quiz Complete!

**+100 points**

Lightning Platform API Basics
100%
Progress: 100%
View more modules