

axway



WHITE PAPER

Navigating the new streaming API landscape

10 ways real-time, event-driven architectures
are redefining the future of business



Navigating the new streaming API landscape

Streaming APIs that deliver data to web, mobile, and device applications have been evolving for many years, but today an important convergence is happening between regular web APIs and streaming technology. The result is the advent of the event-driven architecture, and it's redefining the future of business.

There are a lot of online services, tools, and devices that use the word "streaming," making for a crazy mix of information that will make even the most seasoned technologist's head spin. The objective of this paper is to help anyone – technical or not – navigate the fast-growing and ever-evolving streaming API landscape. We'll walk you through some of the most common use cases for making data available in a real-time stream to help you understand how and when it makes sense to invest in your streaming API infrastructure.

10 ways real-time, event-driven architectures are redefining the future of business

To help paint a picture of why streaming data using APIs isn't just about video, financial data, or news, we'll take a look at 10 areas of significant streaming technology adoption. We'll start with a couple of universally recognizable names and then explore some that are lesser known:

01 Sales and CRMs with Salesforce

02 Everything social using Twitter

03 Market data from Xignite

04 Messaging and automation with Slack

05 Support and chat with Zendesk

06 Digital currency using Blockchain

07 IoT with Nest

08 Sporting events with Triathlon

09 In-person gathering with Meetup

10 Gaming with Twitch

These real-world examples of the streaming API landscape illustrate just a few of the possible use cases for delivering real-time data to end users through browsers, mobile apps, and internet-connected "things."

Streaming: it's much more than video

When you mention streaming to anyone in the context of the web, they immediately think of Netflix and YouTube. This paper is not about the streaming of video content. It's about widening your perspective on what real-time streaming can be, and the many reasons a company, organization, institution, or government agency would want to invest in streaming APIs and event-driven architecture.

01 Sales and CRMs with Salesforce



When you Google streaming APIs, Salesforce tops the list. The dominance of Salesforce on this topic, and in the workplace as a customer relationship management and sales platform, makes it an excellent place to begin any conversation about the streaming API landscape. It shows what is possible in the context of a tool many already use, and a set of digital resources they already find meaningful to day-to-day operations.

The Salesforce Streaming API augments other Salesforce API variations, including but not limited to the SOAP and REST versions. All Salesforce APIs are meant to give users granular access to all the data managed within their CRM implementation. Depending on the scope of data an organization manages via their Salesforce installation, the types of APIs they will need for integration will vary. Most organizations will only need the Salesforce REST API, which uses a basic request and response API model to tackle most any system integration, or delivery of a web or mobile application.

When do users need the Streaming API?

Salesforce users who manage a lot of data, as well as high-volume engagements and activity, will need the Salesforce API that employs streaming technology. This will make Salesforce a more real-time and event-driven experience — which helps users make more sense of the volume of data available within Salesforce.

When it comes to the underlying streaming technology for the Salesforce APIs, the platform employs the CometD protocol to enable the delivery of events they define as PushTopics using a query language they have named SOQL. CometD “leverages WebSocket when it can and makes use of an Ajax Push technology pattern known as Comet when using HTTP.” This allows for anything that happens via Salesforce to be defined by a PushTopic and delivered as an event in real time via the Salesforce Streaming API. It also turns anything that happens via Salesforce into a stream that becomes subscribe-able via any topic defined as a PushTopic.

Salesforce developers can craft a PushTopic using a custom SOQL definition, which defines a specific channel. Clients can then subscribe to these channels, and when any object as defined by the PushTopic is created, updated, deleted, or undeleted, an event is triggered. If the details of the change meet the rules defined in the PushTopic query, a notification is generated and sent out to each subscribing client via a long-running HTTP or WebSocket stream, depending on the scenario.

Overlaying an event-driven architecture across the entire Salesforce platform allows all clients to be able to respond to events in real time, and not just make requests via the regular API.

The reasons you want to use the Salesforce Streaming API are in alignment with why you'd use streaming technology in general – to reduce client polling on APIs and allow applications to respond to common events in real time, keeping the interface up to date and machine learning models using Salesforce Einstein always learning.

A couple of interesting and unique things about the Salesforce Streaming API involve the concept of message reliability, the ability to replay past events and the idea of message durability, which saves events for 24 hours, allowing for the retrieval and replay of messages within recent history.

The Salesforce Streaming API demonstrates the potential of embracing an event-driven approach to operating a platform. Salesforce is one of the original API pioneers, and the company's approach to adopting an event-driven layer to an existing API platform represents the future for all API providers – Salesforce is just ahead of the curve. It shines a light on why companies are increasing their investment in event-driven architecture, revealing a healthy maturity pattern that emerges to help deal with higher volumes of activity.

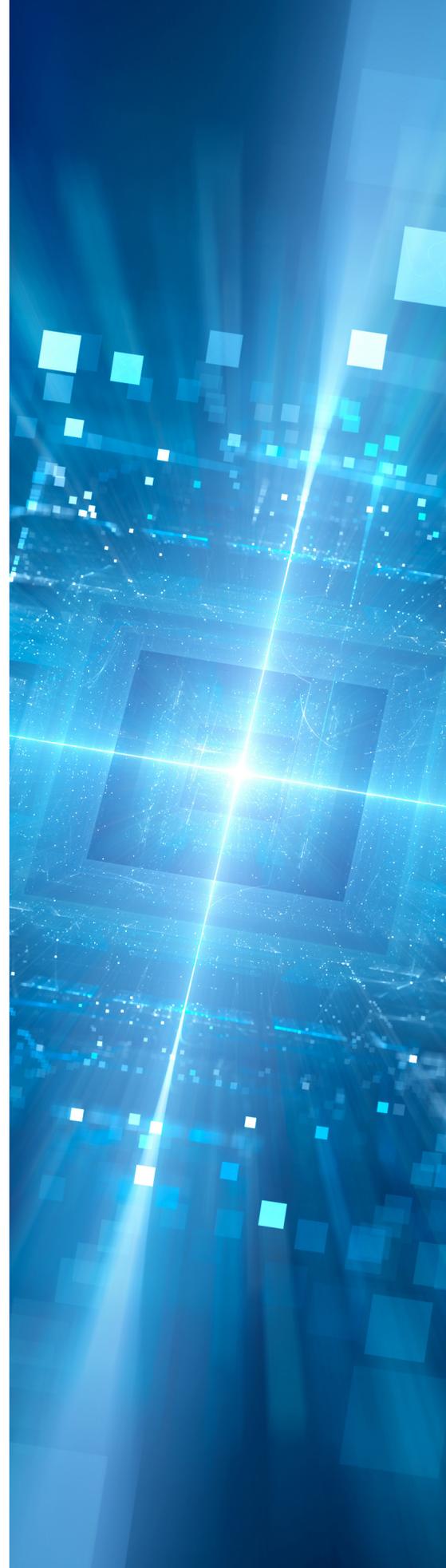
02 The Twitter Streaming API



After Salesforce, the most prominent streaming API people think of is Twitter, which gives developers access to the social signals they desire. The state of the Twitter Streaming API has shifted over the last couple of years because of the Twitter/Gnip relationship/acquisition and is something that seems to be moving towards existing as an enterprise offering. However, as of the writing of this paper, there is still a publicly available streaming endpoint for accessing the valuable social data stored within the Twitter platform.

The Twitter Streaming API returns public Twitter status updates through the filtering of search phrases, user IDs, and by location, as well as public statuses that match one or more filters a consumer has defined. Multiple parameters may be specified, which allows most clients to use a single connection to the Streaming API.

Both GET and POST requests are supported but GET requests with too many parameters may cause the request to be rejected for excessive URL length. Use a POST request to avoid long URLs. They allow for filtering up to 400 keywords, 5,000 follow user IDs and 25 0.1-360 degree location boxes (if you want anything more you have to upgrade to enterprise access).





You can use three parameters to set up filters for any Twitter API stream connection:

- **Follow** – A comma-separated list of user IDs, indicating the users to return statuses for in the stream.
- **Track** – Keywords to track. Phrases of keywords are specified by a comma-separated list.
- **Locations** – Specifies a set of bounding boxes to track. See locations for more information.

The free streaming tier requires you to disconnect if you want to change your filters at all, where if you upgrade to the enterprise tier you can add and change the filters while it runs. You also get added benefits like replay, and you get up to 250K users and keywords at the enterprise level. The free streaming tier gives you a nice drink of the Twitter hose, but if you want a larger, more sustained drink, all roads appear to lead toward the enterprise levels of access.

Twitter provides a lot of information about how to keep a connection stable, and how to avoid violating rate limits to help ensure a stable(r) experience. Twitter is still working on figuring out the business model and enterprise strategy for the free layer of Twitter, and their Gnip acquisition and former partnership. For the near future, the Twitter Streaming API continues to provide an avenue for obtaining garden and fire hose streams of Twitter users, search, and location data – delivering precise streams of an individual user, lists, and other searches from across the Twitter platform.

Streaming of social data ensures that business users can stay in tune with changes across the social landscape in real time. This provides a first mover advantage, with updates in real time and the ability to respond to the most important changes occurring across the slice of the social landscape they are looking to understand and participate in. Moving beyond having to ask for updates, to having the most relevant social updates pushed to you on the web and on the mobile devices we depend on to stay in tune with our personal and professional lives.

03 Market data from Xignite



Beyond Salesforce and Twitter, another prominent API and data format that people think of when talking about streaming data is out of the market data provider Xignite, which provides topical streams of financial data using the latest technology. The Xignite market data streaming APIs put Server-Sent Events (SSE) to work, delivering market data to web and mobile applications in real time.

The streaming API delivers market data to other systems, web, and mobile applications with the following features:

- Global real-time and delayed equity, index, currency, and metal quotes
- Zero-footprint production infrastructure solution
- Native browser support for HTML5 open standard Server-Sent Events (SSE)
- HTTP protocol eliminates special firewall configuration requirements
- Scalable to millions of end-user devices

The Xignite API helps companies realize the potential of market data within their web, mobile, and desktop applications by:

- Streaming market data directly to your apps on end-user devices in minutes
- Eliminating the need for fanout infrastructure hardware and software
- Drastically reducing time to market for large-scale financial mobile and web applications
- Reducing total cost of ownership
- Optimizing network bandwidth by streaming only field values that have changed
- Optimizing mobile user experience by minimizing battery usage

Staying ahead of the conversation when it comes to financial markets is essential. There is plenty of evidence regarding how even a millisecond advantage in receiving signals regarding market changes can give an investor the advantage they need. If you have to request data when it comes to market changes, you are immediately at a disadvantage compared to investors who have data delivered in real time using streaming APIs.

Xignite is a strong example of why streaming technology is relevant --people get the need for real-time market data in today's business world. The robust and diverse suite of APIs, including SOAP, REST, and a streaming API edition for all resources deliver billions of API calls that support financial applications, and consumers who are looking to understand how markets are behaving in real time.

Xignite's streaming APIs deliver many data points that any company is going to need if they want to pay attention to the markets in real time.

Getting business done, shifting conversations, and moving markets

Salesforce, Twitter, and Xignite provide us with three significant examples of streaming APIs out in the wild, and why API providers should consider delivering data and content in real time.

All three have demonstrated the importance of having a streaming API solution available for delivering data when and where it is needed in real time, and when the most meaningful events occur.

04 Messaging and automation with Slack



Another significant area people think of when it comes to streaming online is messaging. One of the top messaging platforms available online today that also employs a streaming messaging API, is Slack. The Slack Real Time Messaging API, also called RTM, is a WebSocket-based API that allows developers to receive events from Slack in real time.

Once you have connected to the Slack message server it provides a stream of events, including both messages and updates to the current state of the channel, allowing the messaging client to easily maintain a synchronized copy of what is happening, with almost everything that happens resulting in an event being sent to any connected clients. Slack supports a wealth of events, like when a new message is created, a file has been uploaded, and almost any action that can be taken via the Slack messaging application(s) interfaces.

There are many other real-time messaging applications available online, but Slack provides a strong example of a streaming API implementation that powers a widely used online messaging platform. It's moving the blueprint for how we deploy streaming technology beyond just CRM, social, and market data.

Slack shows how streaming technology can be used to help provide access to a variety of resources, while also exposing the meaningful events that occur across any platform. By leveraging streaming APIs to ensure the most recent and relevant conversations occurring in your messaging applications are at your fingertips, you stay informed on what is happening with the conversations you have each day.

05 Support and Chat with Zendesk



Another relevant streaming API that also operates in the messaging space is out of Zendesk, a company that focuses on a different market segment from Slack to provide a real-time, event-driven API for seamlessly managing support operations.

The Zendesk Real-Time Chat API provides programmatic access to ongoing activity from any Zendesk Chat account and streaming access to a handful of read-only resources:

- Display chats and agents metrics
- Create and display a real-time dashboard
- Monitor a specific department
- Predict or estimate capacity and other derived metrics

The Zendesk RealTime Chat API balances the integration opportunities around its streaming WebSocket-based API and the REST API, with the Streaming API allowing a developer to receive events from Zendesk Chat after you subscribe to one or several topics, and the REST API allows you to pull the actual data you need for each of those elements.

This is an example of how simple web APIs and event-driven architecture work in concert, to ensure you are supporting your customers in real time and reducing friction in their world, which will translate to a happier, more loyal customer down the road.

06 Digital currency using Blockchain



Augmenting the market data opportunity being served up by Xignite, another significant area of growth in the usage of streaming technology has been around the extremely active cryptocurrency and blockchain space. This is another area where you will find request and response APIs living alongside streaming APIs, providing access to the events that are happening across the disruptive blockchain landscape.

Almost every major blockchain provider has an API as well as a streaming API, but one notable one is available at Blockchain. Similar to Zendesk, the REST and WebSocket APIs work side by side, providing information about new transactions and existing blocks. It leverages a two-way WebSocket stream for pushing up-to-the-second market information, while also allowing consumers to submit transactions back to the server in a real-time, circular flow of information.

Leveraging streaming technology, Blockchain is:

- Staying in tune with markets at a whole new pace and operate at a much more distributed level
- Creating digital worlds where having the most up-to-date information is how you obtain the edge you need to stay ahead of the market
- Echoing everything we know about traditionally regulated markets but adding much more volatility than mainstream markets are willing to bear
- Providing another example of how streaming APIs can be used to help understand the complexities in any market, and get the information regarding changes, trends, and other fluctuations in real time, instead of having to wait and manually ask for it

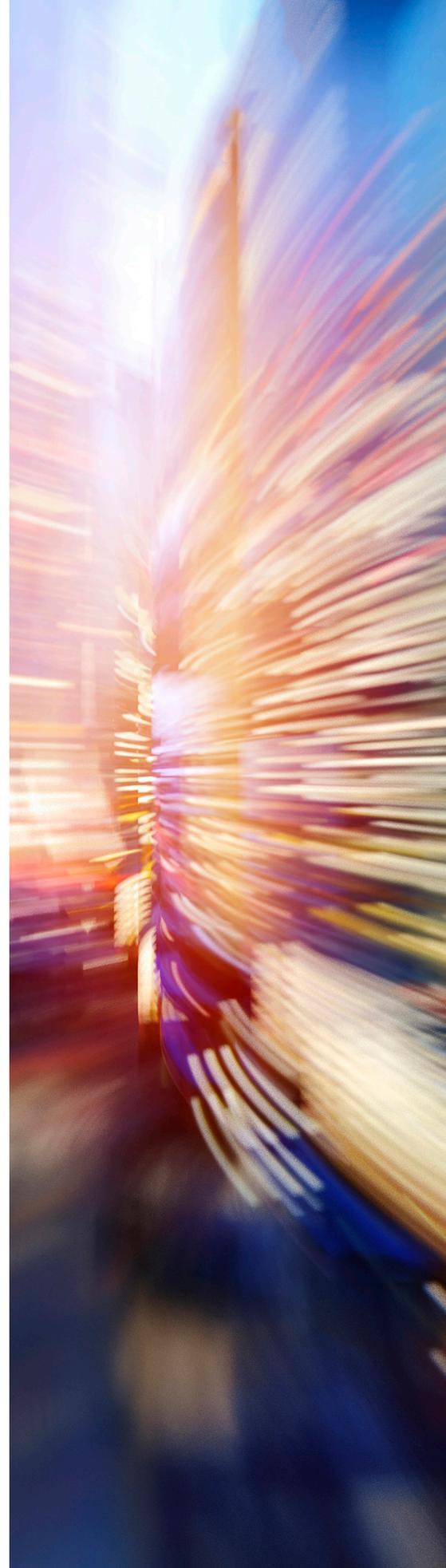
07 IoT with Nest



The Nest streaming API is meant to support clients connecting directly to Nest services, which powers Nest devices through cloud-to-cloud integrations that allow API consumers to listen for changes, instead of polling the Nest REST API for anything new.

The Nest streaming API is available at the same endpoint as the regular API, but instead of sending an application/JSON content type you send a text/event-stream content type request.

Once connected, you will receive updates for the requested Nest device as each event occurs, delivered as a long-running HTTP connection that employs the web standard Server-Sent Events (SSE).





Nest provides a compelling example of delivering SSE-powered streams from events that are occurring via home thermostat, smoke and CO2 alarms, and indoor and outdoor security cameras. This illustrates a sensible use of not just streaming services, but also doing it intelligently using SSE instead of the bulkier WebSockets, or other custom approaches. It is common to come across providers using Websockets for one-way streaming, which is overkill for anything that doesn't need to send data in both directions.

Nest's approach to using streaming APIs provides us with a blueprint for not just home thermostats, but for any Internet of Things device in our homes or businesses. Using streaming APIs to send us the data we need in any moment – in this case it is the temperature of a room in our home or business.

08 Sporting events with Triathlon



The Triathlon API provides programmatic access to data surrounding endurance events around the globe. This gives Websocket access to commentary around Triathlon events that light up real-time streams of information which are used to support events and reach out to the audience.

The Triathlon API further pushes the boundary for streaming technology by building on existing messaging and social streams to boost engagement in both physical and digital worlds. It demonstrates how streaming APIs can be applied to events and leveraged to meet the demands of an audience that is equipped with multiple phones and wants to know what is happening at every moment of the race.

With its streaming APIs, Triathlon is:

- Better connecting sporting event participants with the fans
- Tracking what is going on in the sporting event and pushing relevant signals and other data to fans in real time via the devices they already have
- Bringing athletes, coaches, and organizers closer to the fans, making them truly feel like they are part of the action

09 In-person gathering with Meetup



Meetup has changed how we gather in our communities, providing a new online forum for organizing, managing, and participating in face-to-face gatherings in the real world. In addition to standard access via their regular request and response APIs, Meetup allows you to stream the RSVPs, photos, and comments associated with each event.

Meetup does a good job at augmenting their existing web APIs with streaming editions because they provide multiple approaches to getting data in a real-time way.

- Long polling – A long-polling stream can be easily consumed using JavaScript in any modern browser. Event RSVPs, photos, and comments are received when they happen, or as soon as your script finishes handling its last notification.
- Websockets – For browsers that support it, WebSockets is a more efficient alternative to the long-polling stream. This is a push-only endpoint and will discard any messages received from the client after the socket is open.
- Chunked – Live HTTP stream of event RSVPs, photos, and comments within public Meetup groups. This method uses chunked transfer encoding to maintain a persistent connection with the client. This connection will only be terminated for server maintenance or a connection error.

With its streaming APIs, Meetup is:

- Offering a seamless switch between long-running HTTP connections, obtaining data in ongoing chunks of data, or moving off HTTP and using TCP powered WebSockets to get the job done
- Allowing Meetup API consumers to shift the protocol and message when it comes to getting at Meetup event data in real time, satisfying the demand for real-time event information within applications
- Ensuring that Meetup organizations can stay in tune with what is happening across the events they are organizing within each city, and helping ensure that Meetup attendees are better engaged, having conversations, and participating more within the overall Meetup experience





10 Gaming with Twitch



Twitch bridges the worlds of online video streaming and online data streaming. It streams video, and also uses streaming APIs to make data about the streaming videos available for use in web and mobile applications.

You can stream data about live video streams using these Twitch APIs:

- Get Stream by User
- Get Live Streams
- Get Streams Summary
- Get Featured Streams
- Get Followed Streams

Twitch is drawing a distinct line between streaming video and the streaming of data. Just like streaming APIs can work with request and response APIs, streaming APIs can work alongside the streaming of video.

With its streaming APIs, Twitch is:

- Allowing for digital media to be streamed, but also the metadata around the streaming media
- Encouraging users to stay up to date with the latest information and chatter occurring around each video stream, strengthening engagement across the platform
- Making data around game activity more accessible by platform operations, as well as gamers and fans, making the experience more intimate and real-time

Making real-time real

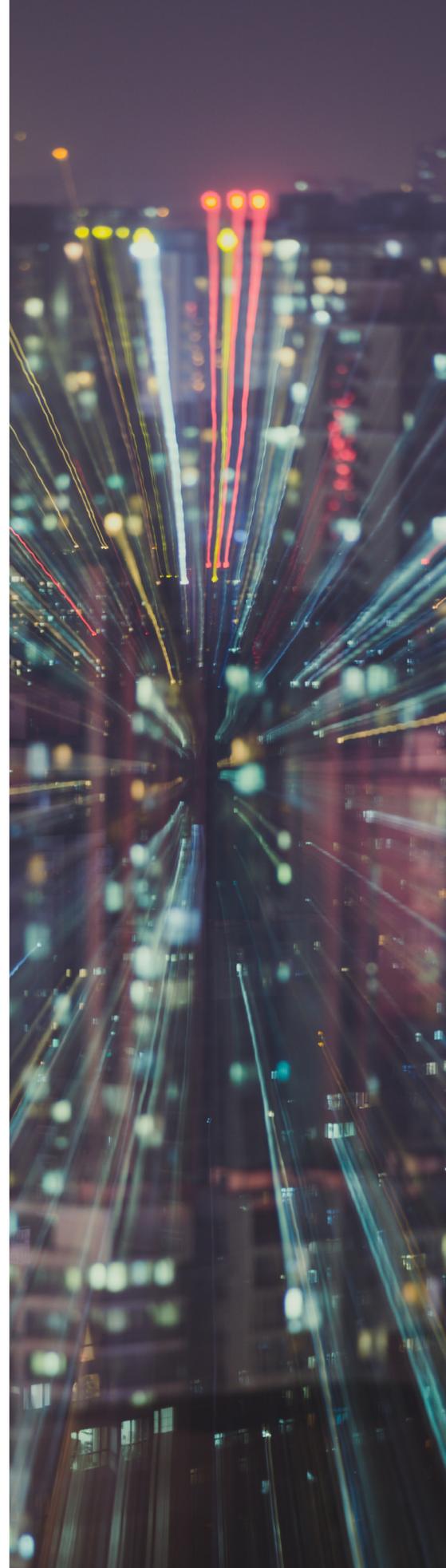
This paper delves into only 10 examples of streaming APIs that exist today. It is meant to provide a simple cross-section of what is possible when you offer streaming APIs in an event-driven architecture that makes your data and resources accessible in real time.

Every provider listed above offers streaming APIs that augment, and often work in concert with, their simpler web APIs. It's not just about getting data faster. It is about satisfying the most demanding consumers on top of your existing platform and serving them up the data and events that matter to them the most – in real time.

There are a handful of streaming protocols driving the majority of streaming platforms today. Websockets has commanded a lion's share of the streaming API attention with its two-way protocol, but Server-Sent Events (SSE) is seeing an uptick in adoption recently because of its single-direction approach that is simpler to implement, and in closer alignment with the HTTP protocol. SSE provides low-overhead transport for data to be delivered as it is updated, delivering data to consumers when it matters most. It can be applied to contacts, sales, messaging, devices, events, and any other digital asset imaginable.

As Salesforce, Twitter, Xignite, Slack, Zendesk, Blockchain, Nest, Triathlon, Meetup, and Twitch demonstrate, streaming APIs have a symbiotic relationship with existing web APIs. Delivering resources in concert with existing API infrastructure, streaming technology complements other event-driven patterns like publish and subscribe (Websub) and Webhooks.

The evolving streaming API landscape continues to expand on the ways in which digital resources are moved around the enterprise and on the web, establishing a robust set of protocols for delivering multi-speed APIs that operate primarily via HTTP, but also using TCP solutions like Webhooks and Kafka when it makes sense.



3 tips for converting your APIs to streaming APIs

Knowing when to invest in streaming versions of your APIs comes from experience. It takes operating your APIs as basic request and response solutions, getting to know the data being made available, and understanding how consumers will be putting that data to work.

Here are three tips for understanding how and when to invest in your streaming API infrastructure:

1. Know your data

How much does it change, and what are the most meaningful events that occur throughout the average day putting this data to work? Having an intimate knowledge of your data will help you understand whether or not streaming APIs will be necessary.

2. Know your technology

Study the different types of streaming technology like Server-Sent Events (SSE) and Websockets, understand the tradeoffs between HTTP and TCP approaches, and compare one-way and multi-directional streams. Having a clear understanding of the technologies in use will help you sensibly apply the right set of tools.

3. Know your consumers

Understand who consumes your data, what they need, and how they are putting data to work before you begin investing in streaming technology. Knowing your consumers will ensure you are investing in the right places and with the right technology, maximizing the benefits of an event-driven architecture across your organization.

AMPLIFY™ Streams
Augment your existing APIs
with an event-driven layer to
deliver data when and where
it's needed

GET STARTED

portal.streamdata.io/#/register

Is your API program
accelerating innovation?
Take the free Digital Maturity
Assessment to find out

TAKE ASSESSMENT

axway.com/digital-maturity