Ably Reactor Integrations

Reactor Integrations link your other systems to Ably, bringing your compute closer to the edge, This provides the flexibility to use the services and systems you're already working with while Ably handles the complexity and scale of doing so with minimum latency.

ably

In a hybrid and multi-cloud world, systems are increasingly spread across different environments. To connect these disparate parts to access compute, process messages, and run code anywhere often requires bespoke engineering. In addition, handling and transforming high volumes of realtime messages is tricky. Queuing is often required to process and consume messages but adds complexity to system design. Reactor Integrations solve both these problems.

Effortlessly integrate and bring your compute to the edge

Reactor Integrations link your other systems - be they legacy, hybrid, or cloud environments - to the Ably network. This provides the flexibility to use the services and compute you're already working with, eliminating the need for you to build the integrations, operate and scale the required infrastructure for those integrations, and ensure minimum latency.

Queue and consume high volumes of realtime messages in bulk, transform or filter messages by executing functions on them, or pass messages on to other systems for onward processing. In short, Reactor Integrations bring your compute and business logic to the edge of Ably's network so you can build rich realtime features and functionality without adding further complexity to your system architecture.

Your worker servers consume from our queues.
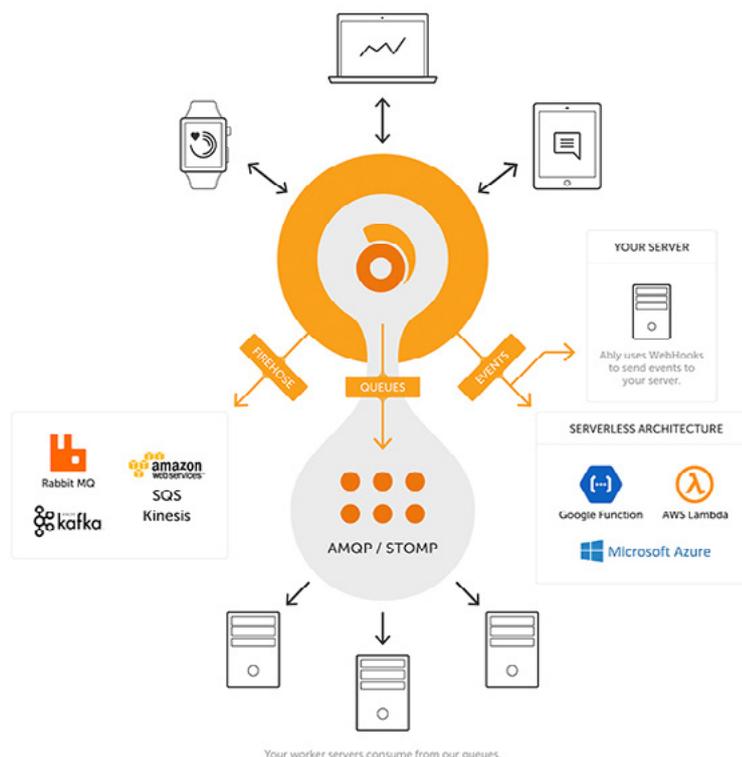
Benefits

✓ **Ably-hosted queues using AMQP or STOMP**
Seamlessly scale your realtime messaging with Ably-hosted queues.

✓ **Ready-made processing pipelines seamlessly link your systems to Ably**
Easily trigger events, queue into servers, or stream into third-party systems so you can focus your engineering efforts elsewhere.

✓ **Bring your compute to the edge of the Ably network**
React to data as it happens, trigger events in milliseconds, reliably and rapidly execute business logic across your systems. The Reactor is designed to give you the engineering flexibility to execute your compute at the edge, how you see fit.

✓ **Reduce engineering complexity and ship faster**
Building with Reactor eliminates the need for you to build integrations, operate and scale the required infrastructure for those integrations, and ensures minimum latency between systems.

✓ **Scale and adapt seamlessly**
Ably can accommodate any amount of engineering and business growth.

✓ **Extensive cloud and protocol support**
Queue data using AMQP, STOMP, Rabbit MQ, Apache Kafka, Amazon SQS, and Amazon Kinesis. Trigger functions and events with Google Function, AWS Lambda, Microsoft Azure, and WebHooks. Ably is always adding more integrations.

How are Reactor Integrations used?

Reactor Integrations consist of three components:

- **Reactor Queues** are Ably-hosted queues for queuing realtime messages into servers

- **Reactor Functions and Webhooks** execute on serverless functions or run code

- **Reactor Firehose** passes data onto internal or external services for processing

Reactor Queues: guaranteed, scalable message delivery and storage

Reactor Queues provide a straightforward and robust way to consume realtime messages from your worker servers without having to worry about queueing infrastructure. Realtime messages can be published to internal queues (hosted by Ably using the STOMP or AMQP protocols) or external external streams or queues (such as Kinesis, Kafka, RabbitMQ).

When delivered with [our connection state recovery](#), queues provide a decoupled (publishers can publish without waiting for consumers), resilient (messages are stored until a consumer has acknowledged the message has been processed successfully), and scalable (adding more consumers increases throughput capacity) means to publish data streams to any number of devices. Queues gives you the assurance that every message published on a channel is received by all devices subscribed to that data.

Reactor Functions and Webhooks: trigger functions and events

Easily monitor channels for events and trigger functions or run code when they occur. Following channel lifecycle events (such as channel creation, presence events (members entering or leaving channels), or messages being published you can invoke serverless functions with Reactor Functions or trigger events via Webhooks.

This allows you to run code to transform and enrich realtime messages on the fly, adding rich functionality. For example, a developer might want to send a welcome message to someone when they first enter a chat channel. A Reactor Function can be used to achieve this.

Reactor Firehose: pass data to internal or external services

Pass realtime messages from the Ably network directly to another internal or external service, allowing you to process data in realtime.

This makes it easier to trigger events, persist messages to your own database, and publish updates once a channel becomes active. For example, using workers consuming from your stream or queue, you could persist each message of a live chat to your own database, start publishing updates once a channel becomes active, or trigger an event if a device has submitted a location that indicates that it has reached its destination.

Key features

- ✓ Message queues to process, transform, and react instantly to data in motion

- ✓ Firehose to stream directly to internal or external services at low latencies

- ✓ WebHooks to your servers when channel or presence events occur

- ✓ Invoke serverless functions across third-party clouds

- ✓ Battle-tested, globally-distributed [Data Stream Network](#)

- ✓ Simple yet powerful APIs

Key numbers

< 1/10
second latency worldwide (<100ms)

Billions
of messages sent each day

100s of millions
of messages per second network capacity

30,000,000+
monthly end users

7,500+
developers building on Ably

If you're interested in how the Ably Reactor can work for you, get in touch hello@ably.io or visit our website www.ably.io.

Simpler architecture and reduced engineering complexity

To achieve this degree of interconnectivity in-house, organizations would need to focus additional engineering resource on the already-difficult task of streaming data between devices. Growth-focused organizations tend to offload this heavy infrastructure burden. Opting to stream data with Ably's Data Stream Network they gain out-of-the-box queuing and data processing pipelines at no extra cost. And they're able to redeploy resources into their core engineering problems.

Scalable, future-proof engineering

As engineering teams grow and businesses mature the number of realtime messages and types of functions needed can change. Constantly building and supporting new integration pipelines constitutes a heavy engineering burden. Ably's realtime messaging infrastructure layer accommodates ever-evolving scale and compute requirements.

Reactor Integrations give engineers the flexibility to develop with Ably-hosted queues and their existing systems. It simultaneously simplifies system architecture while reducing overall infrastructure investment. And, when combined with the Ably Adapter, provides a truly interoperable and future-proof realtime infrastructure layer.

The Ably Adapter eliminates lock-in in a fragmented ecosystem of protocols, future-proofs system architecture, and frees engineers to develop in numerous directions. By providing native interoperability between protocols such as WebSockets, MQTT, SSE, and proprietary realtime protocols developers can do what they do best: build without worrying about infrastructure or integration issues down the line. For our customers it often goes hand-in-hand with the Reactor.

The Ably Adapter supports:		Reactor Integrations support:	
Ably Native SDKs	Websockets	Google Cloud Functions	AWS Lambda
SSE/HTTP Streaming	HTTP	Azure Functions	Cloudflare Workers
MQTT	AMQP	Webhooks	WebSub *Get in touch*
STOMP	GRPC *Get in touch*	AMQP	Kafka *Get in touch*
		Kinesis	Amazon SQS
		Zapier	IFTTT

As cloud and edge computing continue to evolve and become an essential part of modern apps, we'll support the best protocols, systems, and services that emerge, leaving you free to adopt systems best-suited to your organization. Just as organizations choose Ably because we offer a best-in-class realtime messaging APIs, so should organizations be free to choose the best-in-class services and link those to Ably.

We believe Reactor Integrations is the best available mechanism to support common use-cases and futureproof your architecture based on three key factors:

- Ensuring you can maintain a supremely secure, performant, and scalable execution environment. Cloud service providers can focus soley on their core competencies, and so can provide an extremely effective technical solution better than any homegrown one.

- Helping you to maintain an excellent developer and operator experience as cloud providers can offer tooling and ecosystems to support wider needs.

- You own your business logic. At Ably we believe that business logic is best managed in services under your own control. This way you have the right level of control in order to manage operations for that code.