SGCWEBSOCKETS · TECHNICAL DOCUMENT

Google Cloud Pub/Sub

Google Cloud Pub/Sub REST client — publish to topics, pull from subscriptions and manage IAM bindings from Delphi.

eSeGeCe

WWW.ESEGECE.COM

Overview

Pub/Sub brings the flexibility and reliability of enterprise message-oriented middleware to the cloud. At the same time, Pub/Sub is a scalable, durable event ingestion and delivery system that serves as a foundation for modern stream analytics pipelines. By providing many-to-many, asynchronous messaging that decouples senders and receivers, it allows for secure and highly available communication among independently written applications. Pub/Sub delivers low-latency, durable messaging that helps developers quickly integrate systems hosted on the Google Cloud Platform and externally.

At a glance

COMPONENT CLASS `TsgcHTTPGoogleCloud_PubSub_Client`	STANDARDS / SPEC **Google Cloud Pub/Sub overview**
TRANSPORTS **TCP, TLS**	PLATFORMS **Windows, macOS, Linux, iOS, Android**
FRAMEWORKS **VCL, FireMonkey, Lazarus / FPC**	EDITION **Standard / Professional / Enterprise**

Features

- Balancing workloads in network clusters. For example, a large queue of tasks can be efficiently distributed among multiple workers, such as Google Compute Engine instances.
- Implementing asynchronous workflows. For example, an order processing application can place an order on a topic, from which it can be processed by one or more workers.
- Distributing event notifications. For example, a service that accepts user signups can send notifications whenever a new user registers, and downstream services can subscribe to receive notifications of the event.
- Refreshing distributed caches. For example, an application can publish invalidation events to update the IDs of objects that have changed.

- Logging to multiple systems. For example, a Google Compute Engine instance can write logs to the monitoring system, to a database for later querying, and so on.
- Data streaming from various processes or devices. For example, a residential sensor can stream data to backend servers hosted in the cloud.
- Reliability improvement. For example, a single-zone Compute Engine service can operate in additional zones by subscribing to a common topic, to recover from failures in a zone or region.

Technical specification

Standards & specs	Google Cloud Pub/Sub overview · Pub/Sub REST API reference
Component class	`TsgcHTTPGoogleCloud_PubSub_Client` (unit `sgcHTTP_GoogleCloud_PubSub_Client`)
Frameworks	VCL, FireMonkey, Lazarus / FPC
Platforms	Windows, macOS, Linux, iOS, Android

Main properties

The principal published / public properties used to configure and drive the component. Consult the online help for the full list.

`TLSOptions`	Published or public property used to configure or query the component.
`OnAuthToken`	Published or public property used to configure or query the component.
`OnAuthTokenError`	Published or public property used to configure or query the component.
`GoogleCloudOptions`	Published or public property used to configure or query the component.
`LogFile`	Published or public property used to configure or query the component.
`Version`	Published or public property used to configure or query the component.

Main methods

The principal public methods exposed by the component.

`DeleteSnapshot()`	Public function exposed by the component.
`DeleteSubscripton()`	Public function exposed by the component.
`PatchSubscription()`	Public function exposed by the component.
`DeleteTopic()`	Public function exposed by the component.
`Publish()`	Public function exposed by the component.

`PatchTopic()`	Public function exposed by the component.
`ListSnapshots()`	Public function exposed by the component.
`ListSubscriptions()`	Public function exposed by the component.
`ModifyAckDeadlineSubscription()`	Public function exposed by the component.
`ModifyPushConfigSubscription()`	Public function exposed by the component.

Quick Start

Drop the component on a form, configure the properties below and activate it. The snippet that follows shows the typical **Publish a message** configuration sourced from the online help.

> **About this scenario.** Publish a new message in the newly created topic.

Delphi (VCL / FireMonkey)

```
oPubSub := TsgcHTTPGoogleCloud_PubSub_Client.Create(nil);
oPubSub.GoogleCloudOptions.OAuth2.ClientId := '... your google client id...';
oPubSub.GoogleCloudOptions.OAuth2.ClientSecret := '... your google client secret...';
oPubSub.Publish('pubsub-270909', 'topic-1', 'My First Message from sgcWebSockets.'));
```

C++ Builder

```
oPubSub = new TsgcHTTPGoogleCloud_PubSub_Client();
oPubSub→GoogleCloudOptions→OAuth2→ClientId = "... your google client id...";
oPubSub→GoogleCloudOptions→OAuth2→ClientSecret = "... your google client secret...";
oPubSub→Publish("pubsub-270909", "topic-1", "My First Message from sgcWebSockets."));
```

.NET (C#)

```
oPubSub = new TsgcHTTPGoogleCloud_PubSub_Client();
oPubSub.GoogleCloudOptions.OAuth2.ClientId = "... your google client id...";
oPubSub.GoogleCloudOptions.OAuth2.ClientSecret = "... your google client secret...";
oPubSub.Publish("pubsub-270909", "topic-1", "My First Message from sgcWebSockets."));
```

Common scenarios

The following scenarios are lifted verbatim from the online help. Each shows the configuration and method calls needed to drive the component through a specific real-world flow.

1 · Authorization

Google Pub/Sub component client can login to Google Servers using the following methods:

Delphi (VCL / FireMonkey)

```
oPubSub := TsgcHTTPGoogleCloud_PubSub_Client.Create(nil);
oPubSub.TLSOptions.IOHandler := iohOpenSSL;
oPubSub.TLSOptions.Version := tls1_3;
oPubSub.TLSOptions.VerifyCertificate := True;
oPubSub.TLSOptions.OpenSSL_Options.LibPath := oslpDefaultFolder;
```

C++ Builder

```
TsgcHTTPGoogleCloud_PubSub_Client *oPubSub = new TsgcHTTPGoogleCloud_PubSub_Client(NULL);
oPubSub->TLSOptions->IOHandler = iohOpenSSL;
oPubSub->TLSOptions->Version = tls1_3;
oPubSub->TLSOptions->VerifyCertificate = true;
oPubSub->TLSOptions->OpenSSL_Options->LibPath = oslpDefaultFolder;
```

.NET (C#)

```
oPubSub = new TsgcHTTPGoogleCloud_PubSub_Client();
oPubSub.TLSOptions.IOHandler = TwsTLSIOHandler.iohOpenSSL;
oPubSub.TLSOptions.Version = TwsTLSVersions.tls1_3;
oPubSub.TLSOptions.VerifyCertificate = true;
oPubSub.TLSOptions.OpenSSL_Options.LibPath = oslpDefaultFolder;
```

2 · How to Create a new Subscription

Create a new subscription for project with id: pubsub-270909, with subscription name subscription-1 and topic-1

Delphi (VCL / FireMonkey)

```
oPubSub := TsgcHTTPGoogleCloud_PubSub_Client.Create(nil);
oPubSub.GoogleCloudOptions.OAuth2.ClientId := '... your google client id...';
oPubSub.GoogleCloudOptions.OAuth2.ClientSecret := '... your google client secret...';
oPubSub.CreateSubscription('pubsub-270909', 'subscription-1', 'topic-1');
```

C++ Builder

```
oPubSub = new TsgcHTTPGoogleCloud_PubSub_Client();
oPubSub->GoogleCloudOptions->OAuth2->ClientId = "... your google client id...";
oPubSub->GoogleCloudOptions->OAuth2->ClientSecret = "... your google client secret...";
oPubSub->CreateSubscription("pubsub-270909", "subscription-1", "topic-1");
```

.NET (C#)

```
oPubSub = new TsgcHTTPGoogleCloud_PubSub_Client();
oPubSub.GoogleCloudOptions.OAuth2.ClientId = "... your google client id...";
oPubSub.GoogleCloudOptions.OAuth2.ClientSecret = "... your google client secret...";
oPubSub.CreateSubscription("pubsub-270909", "subscription-1", "topic-1");
```

3 · How to Read messages from a Subscription

Read messages from previous subscription created.

Delphi (VCL / FireMonkey)

```
oPubSub := TsgcHTTPGoogleCloud_PubSub_Client.Create(nil);
oPubSub.GoogleCloudOptions.OAuth2.ClientId := '... your google client id...';
oPubSub.GoogleCloudOptions.OAuth2.ClientSecret := '... your google client secret...';
oPubSub.pubsub.Pull('pubsub-270909', 'subscription-1');
```

C++ Builder

```
oPubSub = new TsgcHTTPGoogleCloud_PubSub_Client();
oPubSub->GoogleCloudOptions->OAuth2->ClientId = "... your google client id...";
oPubSub->GoogleCloudOptions->OAuth2->ClientSecret = "... your google client secret...";
oPubSub->pubsub->Pull("pubsub-270909", "subscription-1");
```

.NET (C#)

```
oPubSub = new TsgcHTTPGoogleCloud_PubSub_Client();
oPubSub.GoogleCloudOptions.OAuth2.ClientId = "... your google client id...";
oPubSub.GoogleCloudOptions.OAuth2.ClientSecret = "... your google client secret...";
oPubSub.pubsub.Pull("pubsub-270909", "subscription-1");
```

4 · How to create a new Topic

Create a new topic for project with id: pubsub-270909 and topic name topic-1.

```
Delphi (VCL / FireMonkey)
```

```
oPubSub := TsgcHTTPGoogleCloud_PubSub_Client.Create(nil);
oPubSub.GoogleCloudOptions.OAuth2.ClientId := '... your google client id...';
oPubSub.GoogleCloudOptions.OAuth2.ClientSecret := '... your google client secret...';
oPubSub.CreateTopic('pubsub-270909', 'topic-1');
```

```
C++ Builder
```

```
oPubSub = new TsgcHTTPGoogleCloud_PubSub_Client();
oPubSub→GoogleCloudOptions→OAuth2→ClientId = "... your google client id...";
oPubSub→GoogleCloudOptions→OAuth2→ClientSecret = "... your google client secret...";
oPubSub→CreateTopic("pubsub-270909", "topic-1");
```

```
.NET (C#)
```

```
oPubSub = new TsgcHTTPGoogleCloud_PubSub_Client();
oPubSub.GoogleCloudOptions.OAuth2.ClientId = "... your google client id...";
oPubSub.GoogleCloudOptions.OAuth2.ClientSecret = "... your google client secret...";
oPubSub.CreateTopic("pubsub-270909", "topic-1");
```

5 · Publish a Message with Attributes

Pub/Sub brings the flexibility and reliability of enterprise message-oriented middleware to the cloud. At the same time, Pub/Sub is a scalable, durable event ingestion and delivery system that serves as a foundation for modern stream analytics pipelines. By providing many-to-many, asynchronous messaging that decouples senders and receivers, it allows for secure and highly available communication among independently written applications. Pub/Sub delivers low-latency, durable messaging that helps developers quickly integrate systems hosted on the Google Cloud Platform and externally.

```
Delphi (VCL / FireMonkey)
```

```
oPubSub := TsgcHTTPGoogleCloud_PubSub_Client.Create(nil);
oPubSub.GoogleCloudOptions.OAuth2.ClientId := '... your google client id...';
oPubSub.GoogleCloudOptions.OAuth2.ClientSecret := '... your google client secret...';
oAttributes := TStringList.Create;
Try
  oAttributes.CommaText := 'origin=gcloud-sample,username=gcp';
  oPubSub.Publish('pubsub-270909', 'topic-1', 'My First Message from sgcWebSockets.', oAttribute
Finally
  oAttributes.Free;
end;
```

C++ Builder

```
TsgcHTTPGoogleCloud_PubSub_Client *oPubSub = new TsgcHTTPGoogleCloud_PubSub_Client();
oPubSub->GoogleCloudOptions->OAuth2->ClientId = "... your google client id...";
oPubSub->GoogleCloudOptions->OAuth2->ClientSecret = "... your google client secret...";
oAttributes = new TStringList();
try
{
  oAttributes->CommaText = "origin=gcloud-sample,username=gcp";
  oPubSub->Publish("pubsub-270909", "topic-1", "My First Message from sgcWebSockets.", oAttribut
}
__finally
{
  oAttributes->Free();
}
```

.NET (C#)

```
oPubSub = new TsgcHTTPGoogleCloud_PubSub_Client();
oPubSub.GoogleCloudOptions.OAuth2.ClientId = "... your google client id...";
oPubSub.GoogleCloudOptions.OAuth2.ClientSecret = "... your google client secret...";
oPubSub.Publish("pubsub-270909", "topic-1", "My First Message from sgcWebSockets.", "origin=gclo
```

6 · Google Pub/Sub Client

Pub/Sub brings the flexibility and reliability of enterprise message-oriented middleware to the cloud. At the same time, Pub/Sub is a scalable, durable event ingestion and delivery system that serves as a foundation for modern stream analytics pipelines. By providing many-to-many, asynchronous messaging that decouples senders and receivers, it allows for secure and highly available communication among independently written applications. Pub/Sub delivers low-latency, durable messaging that helps developers quickly integrate systems hosted on the Google Cloud Platform and externally.

Delphi (VCL / FireMonkey)

```
oPubSub := TsgcHTTPGoogleCloud_PubSub_Client.Create(nil);
oPubSub.GoogleCloudOptions.Authorization := gcaOAuth2;
oPubSub.GoogleCloudOptions.OAuth2.ClientId := '... your google client id...';
oPubSub.GoogleCloudOptions.OAuth2.ClientSecret := '... your google client secret...';
```

C++ Builder

```
oPubSub = new TsgcHTTPGoogleCloud_PubSub_Client();
oPubSub->GoogleCloudOptions->Authorization = gcaOAuth2;
oPubSub->GoogleCloudOptions->OAuth2->ClientId = "... your google client id...";
oPubSub->GoogleCloudOptions->OAuth2->ClientSecret = "... your google client secret...";
```

.NET (C#)

```
oPubSub = new TsgcHTTPGoogleCloud_PubSub_Client();
oPubSub.GoogleCloudOptions.Authorization = gcaOAuth2;
oPubSub.GoogleCloudOptions.OAuth2.ClientId = "... your google client id...";
oPubSub.GoogleCloudOptions.OAuth2.ClientSecret = "... your google client secret...";
```

Sources used to build this document

Every external claim links back to a primary source. The online-help references decode the canonical deep-link the company maintains for this component.

Primary standard / spec — Google Cloud Pub/Sub overview	cloud.google.com/pubsub/docs/overview
Primary standard / spec — Pub/Sub REST API reference	cloud.google.com/pubsub/docs/reference/rest
Online help — component page	www.esegece.com/help/sgcWebSockets/Components/HTTP/Google/PubSub/Google_Cloud_Pub_Sub.htm
Delphi demo project (in the sgcWebSockets package)	`Demos\20.HTTP_Protocol\03.Google\01.Google_PubSub`
Component page	www.esegece.com/products/websockets/http/google-cloud-pubsub/
Product page	www.esegece.com/products/websockets/

> **Document scope.** This document covers the publicly-documented surface of the Google Cloud Pub/Sub component shipped with sgcWebSockets. For full property, method and event reference consult the online help linked above.