

HarbourBridge UI User Guide

Index

[Introduction](#)

[Terms/Terminology](#)

[Currently Unsupported](#)

[User Journeys](#)

[Setting up HarbourBridge](#)

[Connecting to a Spanner Instance](#)

[Source database connection mechanism](#)

[Configure Schema](#)

[Conversion Assessment](#)

[Spanner Draft](#)

[Tables](#)

[Column](#)

[Primary Key](#)

[Foreign Key](#)

[Interleave](#)

[SQL](#)

[Drop & Restore Table](#)

[Indexes](#)

[Drop & Restore Index](#)

[Add Secondary Index](#)

[Issues & Suggestions](#)

[Warnings](#)

[Suggestions](#)

[Change Global Data type](#)

[Review Changes](#)

[View Assessment](#)

[Session Management](#)

[Save Session](#)

[Session History](#)

[Prepare Migration](#)

[Migration Modes](#)

[Migration types](#)

[Setting up source database](#)

[Setting up target database](#)

[Setting up connection profiles](#)

[Monitoring](#)

[Generated Resources](#)

[End Migration](#)

[FAQ](#)

Introduction

HarbourBridge is an [open source](#) schema and data migration tool for migrations to Cloud Spanner. It provides a unified interface for the migration wherein it gives users the flexibility to modify the generated spanner schema and run end to end migration from a single interface. It provides the capabilities of editing table details like columns, primary key, foreign key, indexes, etc and provides insights on the schema conversion performance along with highlighting important issues and suggestions.

Terms/Terminology

- **Connection Profile** - A representation of a source or destination in terms of the connectivity information required to connect to it (e.g. hostname, user, etc).
- **Migration job** - A migration job represents the process of migrating schema and/or data from source to destination, including managing the dump of historical data, replicating data continuously, monitoring and error handling
- **Session file** - A session file is a snapshot of an ongoing HarbourBridge conversion session. It contains metadata for migration and is structured in JSON format.
- **Interleave** - Spanner's table interleaving is a good choice for many parent-child relationships. With interleaving, Spanner physically co-locates child rows with parent rows in storage. Co-location can significantly improve performance. For more information on interleaving check [here](#).
- **Migration Request Id** - A unique identifier generated for each migration request.
- **Synthetic Primary Key** - It is mandatory for a table in spanner to have a primary key. In cases where the primary key is missing in the source database, HarbourBridge generates a new column **synth_id** and populates it with UUID.
- **Metadata database** - A spanner database with the name **harbourbridge_metadata** which is responsible for storing saved sessions from HarbourBridge.

Currently Unsupported

- **Source database** - We provide support for migrations from following source databases from HarbourBridge UI - MySQL, PostgreSQL, Oracle and SQL Server. All other databases are currently unsupported.
- **Transformations** - Complex transformation like row operations, table splitting, table merging, etc are currently not supported.

User Journeys

HarbourBridge provides support for both schema and data migration. For schema migration, HarbourBridge automatically builds a default Spanner schema from the schema of the source database. This schema can be customized using the HarbourBridge schema assistant. After schema customizations the user can then go ahead with the migration wherein they select the [mode of migration](#) - schema, data or schema and data and [type of migration](#) - bulk migration or minimal downtime migration and then execute the migration. After all the details have been specified, a database gets created in Spanner with the customized schema and data is copied from the existing database to Spanner.

Running an end to end migration comprises of five basic steps, each of these will be talked about in detail in the following sections:

1. [Setting up HarbourBridge](#)
2. [Connecting to a Spanner Instance](#)
3. [Connecting to source database](#)
4. [Setting up spanner schema](#)
5. [Creating spanner database and migrating data to Spanner](#)

Setting up HarbourBridge

Before starting an end to end migration the user first needs to perform a few steps to set up the environment. Follow these [steps](#) to ensure you have a working Cloud Spanner setup, and the necessary permissions in place. Once the user is authenticated and they have created a spanner instance, they can start using HarbourBridge using either of the two mechanisms:

Build from source

This mechanism provides access to the latest version of HarbourBridge. In order to connect to build HarbourBridge from source, the user can make a copy of the HarbourBridge codebase from the github repository by running the following commands:

```
git clone https://github.com/cloudspannerecosystem/harbourbridge
cd harbourbridge
make build
```

To start the HarbourBridge UI, run the following command:

```
./harbourbridge web
```

Using Gcloud

Note: Please note that this mechanism will work only on linux machines.

This mechanism provides access to the last released version of HarbourBridge. In order to connect to HarbourBridge via [Gcloud](#), the user can install the harbourbridge component of gcloud by executing the below command:



```
gcloud components install harbourbridge
```

To start the HarbourBridge UI, run the following command:

```
gcloud alpha spanner migration web
```

Connecting to a Spanner Instance

To perform a migration the user needs to connect to a spanner instance where they specify a valid GCP(Google Cloud Platform) project ID and a valid spanner instance. After the successful connection to the spanner instance HarbourBridge validates if the [metadata database](#) exists. If not, then a new metadata database is created.

 **Project Id:** span-cloud-testing **Spanner Instance Id:** harbourbridge-instance 

Connect to Spanner

Project ID *
span-cloud-testing

Instance ID *
harbourbridge-instance
Min. 4 characters and Max. 30 characters 22/30

CANCEL **CONNECT**

Source database connection mechanism

HarbourBridge allows users to connect to the source database via three connection mechanisms:

- **Connect to Database** - In this connection mechanism the user needs to first select the type of source database engine and then specify the source database connection details like host, port, username, password and database name. Currently, HarbourBridge UI supports the following source databases - MySQL, PostgreSQL, Oracle and SQLServer. User also needs to specify the **spanner dialect**, it specifies the query language used. Spanner provides support for two dialects - Google Standard SQL and PostgreSQL. Note: Users can connect to both local and cloud instances.

Connect to Source Database

Database Engine *
MySQL

Connection Detail

Hostname *
localhost

Port *
3306

User name *
root

Password
.....

Database Name *
sakila

Spanner Dialect

Select a spanner dialect *
Google Standard SQL Dialect

Connect

Cancel

- **Load database dump** - The user needs to generate a dump of their source database and then provide a path to the generated dump file in the input along with the type of database engine. HarbourBridge supports dump file migration for MySQL and PostgreSQL. User also needs to specify the **spanner dialect**, it specifies the query language used. Spanner provides support for two dialects - Google Standard SQL and PostgreSQL. Dump files for MySQL and PostgreSQL can be generated by using [mysqldump](#) and [pg_dump](#) commands respectively. Please note that the dump file should be present in the same machine on which HarbourBridge is running.

Load from database dump

Database Engine *

MySQL

Dump File

File path *

Cloud_SQL_Export_2023-01-05 (18_39_18).sql

✓

Upload File

Spanner Dialect

Select a spanner dialect *

PostgreSQL Dialect

Convert

Cancel

- **Load session file** - In cases where the user would want to save the schema modifications and come back to them later , for instance in case of complex databases or if multiple users are working on the same database they can do so with the help of [session files](#). Users just need to select the database engine and specify the file path for the saved session file to resume the migration process. Please note that the session file should be present in the same machine on which HarbourBridge is running.

Load from session

Database Engine *
MySQL

Session File

File path *
sakila.json

✓

Upload File

Convert

Cancel

Configure Schema

In the configure schema page, the user can view details of the source database and how the spanner database would look like after migration. This includes following information:

1. Conversion Assessment
2. Spanner Draft
 - a. Table details
 - i. Column
 - ii. Primary Key
 - iii. Foreign Key
 - iv. Interleave
 - v. SQL
 - b. Index details
3. Issues & Suggestions

Select Source > **Configure Schema(Google Standard SQL Dialect)** PREPARE MIGRATION VIEW ASSESSMENT SAVE SESSION DOWNLOAD SESSION

Estimation for MySQL to Spanner conversion: 0% of tables can be converted automatically, 12.5% requires minimal conversion changes and 87.5% requires high complexity conversion changes.
To make schema changes go to [Spanner Draft](#) pane. 1

MYSQL **SPANNER DRAFT** 2

Filter Filter by name

Status	Object Name
▼	sakila
▼	Tables (16)
🔍	actor
🔍	address
🔍	category
🔍	city
🔍	country
🔍	customer
🔍	film
🔍	film_actor
🔍	film_category

actor **DROP TABLE**

COLUMNS					PRIMARY KEY		FOREIGN KEY		SQL	
S No.	Name	Type	Pk	Nullable	Name	Type	Pk	Nullable		
1	actor_id	smallint	🔑	Not Null	actor_id	INT64	🔑	Not Null		
2	first_name	varchar		Not Null	first_name	STRING		Not Null		
3	last_name	varchar		Not Null	last_name	STRING		Not Null		
4	last_update	timestamp		Not Null	last_update	TIMESTAMP		Not Null		

ISSUES AND SUGGESTIONS (4) **RULES (0)**

Filter unread warning note suggestion

Status	Summary
⚠️	Some columns will consume more storage in Spanner e.g. for column 'actor_id', source DB type smallint is mapped to Spanner data type int64
⚠️	Column 'actor_id' is an autoincrement column. Spanner does not support auto_increment attribute
⚠️	Autoincrement Hotspot Occured for Table actor and Column actor_id
⚠️	Some columns have default values which Spanner does not support e.g. column 'last_update'

Conversion Assessment

Conversion assessment helps the user to understand the complexity of the schema conversion. It is broken down into 3 categories - automatic conversion, minimal complexity conversion and high complexity conversion. The conversion complexity is calculated based on the warnings and issues per table.

Spanner Draft

When migrating a schema to Spanner, using the default conversion may be sub-optimal owing to functional differences in the features supported and different performance characteristics of Spanner. Following the [best practices](#), it may be better to change keys, drop or add some indexes, or add/remove columns to existing tables in the Spanner schema.

Tables

Users can view detailed information for a table by selecting it from the **Spanner Draft** section. This includes details on columns, primary keys, foreign keys, interleave property, spanner DDL and the option to edit these along with deleting or restoring the table.

Column

Column tab provides information on the columns that are a part of the selected table. It also provides the option to edit the column wherein a user can modify a column name, delete a column, change the data type of the column or modify the null property of the column. Once the user is done with required modifications, they can click on **SAVE & CONVERT** and the update would reflect in the [session file](#) and across all the components in the database.

COLUMNS					PRIMARY KEY	FOREIGN KEY	INTERLEAVE	SQL
MySQL					Spanner			
Order	Name	Type	Pk	Nullable	Name	Type	Pk	Not Null
1	user_id	varchar		Not Null	user_id	STRING		<input checked="" type="checkbox"/>
2	product_id	varchar		Not Null	product_id	STRING		<input checked="" type="checkbox"/>
3	quantity	bigint			quantity	INT64		<input type="checkbox"/>
					<input type="text" value="Column Name *"/> + ADD COLUMN			
<div> CANCEL SAVE & CONVERT </div>								

Primary Key

Users can view and edit the primary key of a table from the primary key tab. They can remove/add a column from the primary key or change the order of columns in the primary key. Once these changes are made, the [session file](#) is updated and they can also be verified from the [SQL tab](#).

COLUMNS					PRIMARY KEY	FOREIGN KEY	INTERLEAVE	SQL	
MySQL					Spanner				
Order	Name	Type	PK	Nullable	Order	Name	Type	Pk	Nullable
1	user_id	varchar		Not Null	1	user_id	STRING		Not Null
2	product_id	varchar		Not Null	2	product_id	STRING		Not Null
					<input type="text" value="Column Name *"/> + ADD COLUMN				

Foreign Key

Users can view and edit the foreign key of a table from the foreign key tab. They can modify the foreign key constraint name, drop the foreign key or convert the foreign key into [interleave](#), if the table is [interleavable](#). Once these changes are made the [session file](#) is updated.

COLUMNS		PRIMARY KEY		FOREIGN KEY	INTERLEAVE		SQL	
MySQL				Spanner				
Name	Columns	Refer Table	Refer Columns	Name	Columns	Refer Table	Refer Columns	
user_cart	user_id	user	user_id	user_cart	user_id	user	user_id	<div>⋮<div>Drop Foreign Key Convert to interleave</div></div>

Interleave

[Interleaving](#) physically co-locates child rows with parent rows in storage. Co-location can significantly improve performance. For example, if there is a *Customers* table and an *Invoices* table, and the application frequently fetches all the invoices for a customer, users can define Invoices as an interleaved child table of Customers. In doing so, a data locality relationship between two independent tables is declared resulting in significant performance improvement. HarbourBridge provides the option to convert a table into an interleaved table if it fulfills all the [criteria](#). The Interleave tab shows up only for tables which are possible candidates for interleaving, based on the existing foreign keys. Once a table is converted into an interleaved table, the UI shows the information of the parent table. Users can also choose to remove this interleaving property and restore the foreign key by clicking on '**Convert back to foreign key**'.
Note: Interleaving property needs to be set during the migration and a table cannot be interleaved after migration.

cart
 DROP TABLE

COLUMNS	PRIMARY KEY	FOREIGN KEY	INTERLEAVE	SQL
<div>Convert to Interleave</div>				

cart
 DROP TABLE

Interleaved: user

COLUMNS	PRIMARY KEY	FOREIGN KEY	INTERLEAVE	SQL
<div>Convert Back to Foreign Key</div>				

SQL

Once the user is done with all the schema modifications they can then visit the SQL tab which shows the Spanner DDL for the modified schema.

cart
 DROP TABLE

COLUMNS	PRIMARY KEY	FOREIGN KEY	INTERLEAVE	SQL
<pre>-- -- Spanner schema for source table cart -- CREATE TABLE cart (user_id STRING(20) NOT NULL, -- From: user_id varchar(20) product_id STRING(20) NOT NULL, -- From: product_id varchar(20) quantity INT64, -- From: quantity bigint) PRIMARY KEY (user_id, product_id)</pre>				

Drop & Restore Table

HarbourBridge also provides the users with the capability to drop and restore tables from the spanner database. Once the table is dropped, it appears as **deleted** under the Spanner Draft section and can be **restored** from there.

The screenshot shows the HarbourBridge interface with the 'Mysql' database selected. The 'Spanner Draft' section is active, and the 'cart' table is highlighted. A 'DROP TABLE' button is visible. A modal dialog titled 'Delete table (cart)?' is open, asking for confirmation by typing the table name 'cart' into a text field. The dialog also includes 'Cancel' and 'Confirm' buttons.




Filter	Filter by name	COLUMNS	PRIMARY KEY	FOREIGN KEY	INTERLEAVE	SQL
Status	Object Name	MySQL			Spanner EDIT	
shopping	Tables (5)	Order	Name	Type	Pk	Nullable
cart		1	user_id			Not Null
people		2	product_id			Not Null
test1						
user						
x_1						

The screenshot shows the HarbourBridge interface with the 'Mysql' database selected. The 'Spanner Draft' section is active, and the 'cart' table is highlighted. A 'RESTORE TABLE' button is visible. The table details are shown below.

Filter	Filter by name	COLUMNS	PRIMARY KEY	FOREIGN KEY	INTERLEAVE	
Status	Object Name	MySQL			Spanner	
shopping	Tables (5)	Order	Name	Type	Pk	Nullable
people		1	user_id	varchar	PK	Not Null
test1		2	product_id	varchar	PK	Not Null
user		3	quantity	bigint		
x_1						
cart						

Indexes

HarbourBridge provides information on indexes for source and spanner databases. It gives details of columns that are a part of the index along with their sort order. Users can edit the index and modify the order of columns in the index, sort order of the columns, add new columns to the index and remove columns from the index.

 index1 (TABLE: CART)  DROP INDEX					
MySQL			Spanner  EDIT		
Column	Sort By	Column Order	Column	Sort By	Column Order
quantity	Asc	1	quantity	Asc	1
user_id	Asc	2	user_id	Asc	2
product_id	Asc	3	product_id	Asc	3

Drop & Restore Index

HarbourBridge also provides the users with the capability to drop and restore existing indexes from the spanner database. Once the index is deleted, it appears as **deleted** under the Spanner Draft section and can be **restored** from there.

Add Secondary Index

Apart from the existing indexes for the source database, users can also add secondary indexes for any table, if required. In order to add an index for a table, the user needs to select the **Add Index** option and provide some details mandatory to create an index like **index name**, **columns** that are a part of the index and their **sort order**.

!	▼	people
		Indexes (0)
✓	▶	test1
✓	▶	user

Add Index

Add rule
✕

Rule info

Rule name *
 new_index
Max. 60 characters, and starts with a letter. 9/60

Rule definition

Rule type *
 Add Index

For Table *
 people

Index Name *
 index1
Max. 60 characters, and starts with a letter. 6/60

Column Name *
 name

Sort *
 Ascending

[— REMOVE COLUMN](#)

+ ADD COLUMN

ADD RULE

Issues & Suggestions

HarbourBridge scans through the generated spanner schema and notifies the user of any warnings encountered. It also makes intelligent suggestions to the user which would help them utilize the spanner capabilities to the fullest.

Following are some of the warnings and suggestions handled by HarbourBridge:

Warnings

- Detection of an auto-increment key in source database because auto-increments are currently not supported in spanner

- Spanner data type consuming more storage than source data type
- Redundant indexes
- Addition of [synthetic primary key](#) - synth_id
- [Hotspotting](#) due to timestamp or auto-increment keys





Suggestions

- Modifications related to converting a table into an interleaved one
- Converting an index to interleaved index

ISSUES AND SUGGESTIONS (2) RULES(0) I<	ISSUES AND SUGGESTIONS (3) RULES(0) I<
<div>Filter unread warning note suggestion</div>	<div>Filter unread warning note suggestion</div>
Status Summary	Status Summary
<div>⚠ Redundant Index for Table cart and Column user_id</div>	<div>⚠ Some columns will consume more storage in Spanner e.g. for column 'c', source DB type int is mapped to Spanner type int64</div>
<div>💡 Table cart can be converted as Interleaved Table go to Interleave Table Tab</div>	<div>⚠ Column 'c' is an autoincrement column. Spanner does not support auto_increment attribute</div>
	<div>⚠ Autoincrement Hotspot Occured for Table people and Column c</div>

Change Global Data type

HarbourBridge has a default mapping for all the data types of the supported databases. But it also provides the users with the flexibility to modify the data type mapping, if required. For changing the global data type the user needs to go to the **Rules tab** and click on **Add Rule**, an add rule form opens up wherein the user needs to select the rule type as **Change global data type** and specify the source data type for which they need to modify the mapping and the resulting spanner datatype they want it to be mapped to. After specifying all these details once a user clicks on **Add Rule**, the rule takes effect immediately and the spanner data type is modified based on the new data type mapping.

MySQL					Spanner  EDIT				
Order	Name	Type	Pk	Nullable	Order	Name	Type	Pk	Nullable
1	name	varchar			1	name	STRING		
2	birth_date	date			2	birth_date	DATE		
3	c	int		Not Null	3	c	 INT64		Not Null

Add rule

Rule info

Rule name *

rule1

Max. 60 characters, and starts with a letter.5/60

Rule definition

Rule type *

Change global data type

For object type *

Column

When

Table is *

All table

and column is *

All column

Convert from

Source data type *






date

Convert to

Destination data type *

STRING

ADD RULE

MySQL					Spanner  EDIT				
Order	Name	Type	Pk	Nullable	Order	Name	Type	Pk	Nullable
1	name	varchar			1	name	STRING		
2	birth_date	date			2	birth_date	 STRING		
3	c	int		Not Null	3	c	 INT64		Not Null

Review Changes

In cases where the user is modifying any column associated with an interleaved table, HarbourBridge provides a review pane which shows how their changes would impact tables across the database. And once the user has verified these changes they can click on '**Confirm Conversion**' and the changes would be applied to the database.

Review changes to your schema

Changing an interleaved table will have an impact on the following tables : **user, cart.**

user

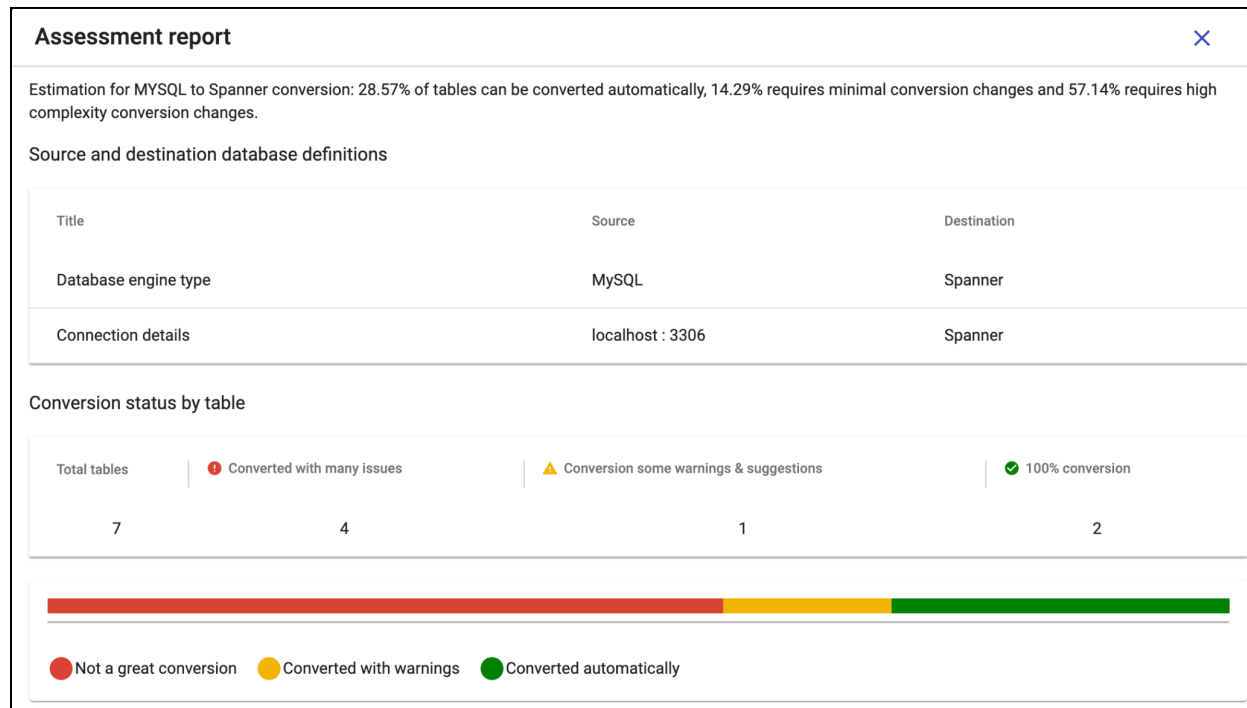
Column	Type	Updated Column	Updated Type
user_id	STRING	user_id1	STRING

cart

Column	Type	Updated Column	Updated Type
user_id	STRING	user_id1	STRING
product_id	STRING	product_id2	STRING

View Assessment

It provides the summary of schema conversion in terms of assessment report. Assessment report helps the user to understand the complexity of the schema conversion. It's broken down into 3 categories - automatic conversion, minimal complexity conversion and high complexity conversion. The conversion complexity is calculated based on the warnings and issues.



Session Management

Save Session

HarbourBridge provides support for collaborative schema modifications so that users can check-point their schema edits and resume later from the point they left off. It also allows multiple users to work simultaneously on the schema assessment for the same database. For this, users can save a [session](#) after any schema modifications by completing a small form wherein they specify the session name, editor name, database name and some notes related to the current session. Once a user clicks on save, an entry is created in the [metadata database](#) with corresponding session details and they can resume the session anytime by going to the [Session History](#) section in the home page.

Save Session

Session Details

Session Name *

session1

Max. 60 characters, and starts with a letter.8/60

Editor Name

xyz

Max. 60 characters, and starts with a letter.3/60

Database Name *

testdb

Max. 60 characters, and starts with a letter.6/60

Notes

edited the primary key

Save Session

Session History

All the saved [sessions](#) show up here with the details about database name, editor name, spanner dialect, etc. Users can resume or download a session from this section. In case a user resumes a session it would be equivalent to the [load session file](#) connection mechanism, the only difference is that metadata is fetched from the [metadata database](#) in the configured spanner instance. In case a user wishes to download a session file, they can do so by clicking on the **Download** button for the required session.

Session history

Choose a session to resume an existing migration session, or download the session file.

Session Name	Editor	Database Type	Database Name	Spanner Dialect	Notes	Created At		
session 2	John	mysql	sakila	Google Standard SQL	minor changes	Mar 01 2023 19:39:43 GMT+0530	Resume	Download
session2	Jacob	mysql	sakila	PostgreSQL	xyz	Mar 01 2023 19:56:45 GMT+0530	Resume	Download
session1	shreya	mysql	sakila	Google Standard SQL		Feb 17 2023 20:29:49 GMT+0530	Resume	Download

Prepare Migration

Once the user is done with schema modifications, they can go ahead with the next step in migration wherein the database gets created in Spanner with the modified schema and data gets migrated to the new spanner database from the existing source database.

Migration Modes

There are three supported modes of migration:

- **Schema** - This mode of migration creates a spanner database with modified schema without writing any data to the new spanner database.
- **Data** - This mode of migration writes data to an existing Spanner database. Please note that for data migration to work the schema of the existing spanner database must match with HarbourBridge's generated spanner schema.
- **Schema and Data** - This mode will create a new spanner database with the modified schema and perform the data migration to the new schema.

Migration Types

HarbourBridge supports two types migration:

- **Bulk migration** - HarbourBridge reads data from the source database and writes it to the database created in Cloud Spanner. Changes which happen to the source database during the bulk migration may or may not be written to Spanner. To achieve a consistent version of data, stop writing on the source while migration is in progress. While there is no technical limit on the size of the database, it is recommended for migrating moderate-size datasets to Spanner(up to about 100GB).
- **Minimal downtime migration** - A minimal downtime migration consists of two components, migration of existing data from the database and the stream of changes (writes and updates) that are made to the source database during migration, referred to as change database capture (CDC). Using HarbourBridge, the entire process where Datastream reads data from the source database and writes to a GCS bucket and data flow reads data from GCS bucket and writes to a spanner database can be orchestrated using a unified interface. It is suggested for databases that require minimal downtime and for larger databases(> 100GB). Currently, HarbourBridge provides minimal downtime migration support for **MySQL, Oracle and PostgreSQL** databases.

The **Prepare Migration** page provides a summary of the source and target databases. It also provides users with the options for selecting migration mode and migration type. Once the users select these values they need to set up source and target database details. Each of these is described in detail below.

Select Source > Configure Schema(Google Standard SQL Dialect) > **Prepare Migration**

Source and Target Database definitions

Title	Source	Destination
Database Type	mysql	Spanner
Number of tables	16	16
Number of indexes	26	26

Migration Mode:
Schema And Data

Migration Type:
Minimal downtime Migr...

Prerequisites
Before we begin, please ensure you have done the following:

Source database details:

1

Setup Source database details

Configure

Target database details:

1

Setup Target details

Configure

2

Setup source connection profile

Configure

3

Setup target connection profile

Configure

Migrate

Setting up source database details

This is required only in case of [session file connection mechanism](#). In this case HarbourBridge needs the source database details to perform data migration. While specifying the source database details, the user can either use **direct connection** or **dump file** connection mechanism and specify the required parameters to connect to the source database. Once the user clicks on save, HarbourBridge shows error or success depending on the status of their connection.

☐ Connect via dump file

☒ Connect via direct connection

Connect to Source Database

Connection Detail

Hostname *

Port *

User name *

Password

Database Name *

CANCEL

SAVE

Setting up target database details

In order to create a spanner database and/or migrate data to it, the user needs to specify the target **database name**, it serves as the name of the spanner database that gets created.

Target Details

Spanner Database *

testdb

Region: us-central1

Spanner Instance: shreya-test-1

Dialect: Google Standard SQL

Cancel

Save

Setting up connection profiles

In case of minimal downtime migration, HarbourBridge needs information about connection profiles which are required by the datastream. There are two connection profiles that need to be set up - **source connection profile** and **target connection profile**.

- **Source connection profile** is used by datastream to connect to the source database and read the data from the source database. Users can either use an existing source connection profile or create a new connection profile from HarbourBridge by specifying a new name for the connection profile and allowlisting the IPs.

Source Connection Profile



Create a new connection profile



Choose an existing connection profile

Select connection profile *

ui_demo_source

Cancel

Save

Source Connection Profile

☒ Create a new connection profile

☐ Choose an existing connection profile

Connection profile name *

test-conn-profile

Copy the public IPs below, and use them to configure the network firewall to accept connections from them.

34.71.242.81

34.72.28.29

34.67.6.157

34.67.234.134

34.72.239.218

Test Connection

Cancel

Save

- **Target connection profile** is used to connect to the GCS bucket where the datastream writes data written to. Users can either use an existing target connection profile or create a new one from HarbourBridge by specifying a new name for the connection profile. Please ensure that the GCS bucket is empty in case you choose an existing connection profile to ensure consistency between source and spanner database. In case the user opts for a new target connection profile, HarbourBridge creates a new GCS bucket with bucket name as the [Migration Request ID](#).

Target Connection Profile

☒ Create a new connection profile

☐ Choose an existing connection profile

Connection profile name *

test-profile

Cancel

Save

Target Connection Profile

☐ Create a new connection profile
 ☒ Choose an existing connection profile

Select connection profile *

dest-conn
 ▼

Cancel
Save

Monitoring

HarbourBridge tracks the updates for schema migration, data migration and foreign key update. This information is polled from the backend at **5 sec** intervals and presented in the UI with separate progress bars. In case of an error during migration, the state of these progress bars is marked as canceled. After the successful completion of migration the generated resources are presented to the users for utilization.

Schema migration completed successfully!

Data migration completed successfully!

Foreign key updated successfully!

Generated Resources:

- Spanner database: [ul_demo54](#)
- GCS bucket: [hb-0bb600bc-3bab-4307-b3b7-3c4a7408a42c/](#)

Generated Resources

For the different types of migration following are the list of resources generated:

- **Bulk migration** - Link to spanner database created and the GCS bucket which has the [session file](#) written to it.
- **Minimal downtime migration** - Link to spanner database created, GCS bucket with [session file](#), datastream and dataflow job launched.

End migration

This is applicable only in case of minimal downtime migration. After a successful schema migration and initialization of dataflow and datastream jobs HarbourBridge presents the user with the steps to be followed to complete the migration. After completion of the migration process users can choose to clean up the migration jobs by clicking on the **Clean Up** button present on the UI.

End Migration

Source database: test_db(mysql)

Spanner database: [ui_demo55](#)

Please follow these steps to complete the migration:

1. Validate that the schema has been created on Spanner as per the configuration
2. Validate the data has been copied from the Source to Spanner.
You can use the [Data Validation Tool](#) to help with this process.
3. Stop the writes to the source database. **This will initiate a period of downtime.**
4. Wait for any incremental writes on source since the validation started on Spanner to catch up with the source. This can be done by periodically checking the Spanner Database for the most recent updates on source.
5. Once the Source and Spanner are in sync, start the application with Spanner as the Database.
6. Perform smoke tests on your application to ensure it is working properly on Spanner
7. Cutover the traffic to the application with Spanner as the Database. **This marks the end of the period of downtime**
8. Cleanup the migration jobs by clicking the button below.

[Cancel](#)

[Clean Up](#)

FAQ

1. How to start the HarbourBridge UI ?
 - To bring up the HarbourBridge UI, please follow the steps mentioned [here](#).
2. Can HarbourBridge be used without connecting to the spanner instance?
 - Yes, HarbourBridge can be used for schema assessment and modifications without connecting to the spanner instance
3. When is a table interleavable?
 - A table is interleavable into a parent table if it has a foreign key referencing the parent table and the primary key is a superset of the primary key of the parent table.
4. What to do in case you are unable to connect to a spanner instance?

- If you are connecting from a GCE VM please verify the access scope of your GCE VM. It should be set to **Allow full access to all Cloud APIs** to allow connections to Cloud API. In case it is set to **default** access, please modify the access level by following the steps below and try again -
 - Stop the VM
 - Edit the VM configuration and change the access scope to **Allow full access to all Cloud APIs**.
 - Restart the VM
 - Otherwise, execute the following command: **gcloud auth application-default login**
5. What happens behind the scenes in minimal downtime migration?
- HarbourBridge orchestrates the entire process using a unified interface, which comprises the following steps:..
 1. Setting up a GCS bucket to store incoming change events on the source database while the snapshot migration progresses.
 2. Setting up the bulk load of the snapshot data and stream of incoming change events using Datastream. **Within the HarbourBridge UI, source and target connection profile will need to be setup**
 3. Setting up the Dataflow job to migrate the change events into Spanner, which drains the GCS bucket over time.

Once the GCS bucket is almost empty, users need to stop writing to the source database so that the remaining change events can be applied. This results in a short downtime while Spanner catches up to the source database. Afterwards, the application can be cut over to Spanner. Currently, HarbourBridge provides minimal downtime migration support for **MySQL, Oracle and PostgreSQL** databases.

