

KPIs/Behavioural Data Integration

- [Supported cloud object storages](#)
- [Supported data formats](#)
- [Integration steps](#)
 - [Configure access](#)
 - [AWS S3](#)
 - [Google Storage](#)
 - [Microsoft Azure Storage](#)
 - [Automating the integration](#)
- [User Actions](#)
- [Logins](#)
- [Purchases](#)
- [Virtual Purchases](#)
- [Level Ups](#)
- [Items](#)
- [Social Dynamics](#)
- [KPIs](#)

In order to support large data volumes, Solsten provides support for cloud-based object storage technologies. These solutions are optimized, cost-effective, and secure ways of storing and sharing arbitrarily large datasets (Big Data).

Supported cloud object storages

- [Google Cloud Storage](#)
- [Amazon Web Services Simple Storage Service](#)
- [Microsoft Azure Storage](#)

Supported data formats

- [Newline delimited JSON](#) (Preferred by Solsten): The individual lines in the file (separated by `\n`) are valid JSON objects, but the complete file may not be
- [Comma-separated values \(CSV\)](#)

Integration steps

Solsten will attempt to connect to your cloud storage, test connectivity, and validate file contents. We will send a notification in case of error:

- Unable to connect to the storage
- JSON/CSV format is not valid

If everything is okay, your data will be regularly processed by Solsten.

Configure access

Store your files separated by their category in the cloud object storage of your choice.

Bucket structure:

- Please include separate root directories for the various data types: `purchases`, `logins`, `actions`, `kpis`, `virtual_purchases`, `level_ups`, `items`, and `social_dynamics`.

- Directories can have any number of subfolders, for example `purchases/date=16000000000/load_id=fgd4b4g4v34t/purchases.csv` (Hive-partitioned directory structure).
- If using a flat directory structure (files right in the root directories), please include a timestamp or some other time-based UUID in the filenames. For example, `purchases/purchases_1600089751.csv`, `purchases/purchases_1600089777.csv`, etc.
- Please indicate the file format for each filename:
 - Files with the prefix `.json` are assumed to be newline-delimited JSON files
 - Files with the prefix `.csv` are assumed to be CSV files



- Ensure that Solsten can access the storage.
- Ensure that we can link behavioural/KPIs data to assessment data by using unique user identifier `user_id`.

AWS S3

- Set IAM policy for Solsten' AWS user (will be provided to you) enabling LIST and GET operations on the contents of the S3 bucket, or
- Share new credentials to the AWS S3 bucket storing the data files (access key and secret key)

Google Storage

- Set IAM policy for Solsten' GCP service account (will be provided to you) enabling LIST and GET operations on the contents of the GCS bucket, or
- Share service account key in JSON format

Microsoft Azure Storage

- Make sure that your storage account's firewall settings [allow connectivity](#)
- Share a [Shared Access Token \(SAS\)](#) with LIST and GET ACL on your files with Solsten

Automating the integration

You can write any amount of data into your cloud object storage area of choice. New data files will be automatically identified and processed by Solsten. A frequent pattern of automation could be daily scheduled upload of data to the cloud storage. Please upload new files instead of overwriting existing ones.

We expect only new changes to the data to be shared (incremental load pattern) instead of pushing full dataset repeatedly.

User Actions

Expected folder: `actions/`

Purpose: track arbitrary user actions. This data type provides a flexible structure to identify a wide variety of actions or activities that the user may take in the game. The differentiation between different actions (on top of setting their specific ID, name, and associated parameters) is completely up to you, but we advise you to keep the actions small, granular, and easy to associate with a specific user.

Please provide a real-time, event-based collection of the data instead of aggregates if possible.

```

1  {
2      "timestamp": "2019-01-01 15:04:05",
3      "user_id": "1001",
4      "action_id": "2002",
5      "action_name": "fight_win",
6      "action_params": [
7          {
8              "key": "enemy_id",
9              "value": "5005"
10         }

```

```
11 ]
12 }
```

Logins

Expected folder: `logins/`

Purpose: track user log ins and information about users.

```
1 {
2   "timestamp": "2019-01-01 15:04:05",
3   "user_id": "1001",
4   "clan_id": "2002",
5   "level": 10,
6   "device": "iPhone X",
7   "country": "US",
8   "language": "en-US",
9   "platform": "Apple",
10  "register_date": "2018-11-14",
11  "last_login_timestamp": "2019-01-01 11:10:06"
12 }
```

Purchases

Expected folder: `purchases/`

Track real-money purchases. Please provide a real-time, event-based collection of the data instead of aggregates if possible.

```
1 {
2   "timestamp": "2019-01-01 15:04:05",
3   "user_id": "1001",
4   "item_id": "2002",
5   "count": 1,
6   "price": 25.5,
7   "platform": "Steam"
8 }
```

Virtual Purchases

Expected folder: `virtual_purchases/`

Track virtual purchases. Please provide a real-time, event-based collection of the data instead of aggregates if possible.

```
1 {
2   "timestamp": "2019-01-01 15:04:05",
3   "user_id": "1001",
4   "item_id": "2002",
5   "count": 1,
6   "virtual_price": 25.5,
7   "currency": "gold"
8 }
```

Level Ups

Expected folder: `level_ups/`

Purpose: track level-ups for each user. Please provide a real-time, event-based collection of the data instead of aggregates if possible.

```
1 {
2   "timestamp": "2019-01-01 15:04:05",
3   "user_id": "1001",
4   "level": 10
5 }
```

Items

Expected folder: `items/`

Purpose: track detailed information about each item.

```
1 {
2   "item_id": "1001",
3   "price": 25.5,
4   "currency": "USD",
5   "name": "Axe",
6   "type": "weapon"
7 }
```

Social Dynamics

Expected folder: `social_dynamics/`

Purpose: track information on the social dynamics of the user, i.e. interactions between users. Please provide a real-time, event-based collection of the data instead of aggregates if possible.

```
1 {
2   "timestamp": "2019-01-01 15:04:05",
3   "user_id": "1001",
4   "target_user_id": "1002",
5   "interaction": "added_as_friend",
6   "interaction_params": []
7 }
```

KPIs

Expected folder: `kpis/`

Purpose: track detailed KPI data associated with a particular user. Any time granularity is supported, but we encourage at least daily KPIs.

```
1 {
2   "timestamp": "2020-03-12 00:00:00",
3   "user_id": "1001",
4   "register_date": "2019-11-18",
5   "churn_date": "",
6   "kpi_id": "transaction_count",
7   "value": 3.0
8 }
```