log-level-data

Batch Log Level Data

Batch Log-level Data (LLD) allows you to retrieve and track feeds of log-level event data that include dimensions not available in the Xandr UI or via the API Report Service in a batch-processed manner. Feeds are generated hourly and are split into one or more files (see File Formats section below). The format of the file you receive will depend on what you specified when you subscribed (e.g., Avro, Protobuf, Protobuf-delimited).

For general information about Log-Level Data, see Log-level Data feeds.

File Formats & Schemas

You may specify one or more of following formats when subscribing to the service.
Use the downloads provided below for packaged example files and code for consuming Log-Level Data files.

NOTE:
Example files are created to assist you when testing the implementation you will use to consume Log-Level Data files. To ease testing, the example files are somewhat simpler than the generated files you will retrieve in production:

- Example files for the protobuf format are not compressed (in production, they are Snappy compressed)
- Example data does not contain values that are typical for a given column. Instead, columns are populated with the column's index number converted to the column's type.

Version	Date Released	Schemas Zip	Example Files Zip	Example Code Zip (includes schemas + example files)	Notes
0.5.36	May 2, 2023	Download	Download	Download	Deprecated `data_costs` from the

Version	Date Released	Schemas Zip	Example Files Zip	Example Code Zip (includes schemas + example files)	Notes
					Standard Feed.
0.5.35	March 8, 2023	Download	Download	Download	Added `fallback_ad_index` to the Standard Feed.
0.5.31	February 1, 2023	Download	Download	Download	Added `segment_data_costs` and `feature_costs` to the Standard Feed.
0.5.27	November 1, 2021	Download	Download	Download	A new field, `extended_ids`, has been added to the Standard Feed and the Curator Feed.
0.5.26	October 14, 2021	Download	Download	Download	The Buyer Transparency Feed (`brand_transparency_feed`) is now a fully supported log-level feed (it was previously an

Version	Date Released	Schemas Zip	Example Files Zip	Example Code Zip (includes schemas + example files)	Notes
					alpha release).
0.5.25	September 8, 2021	Download	Download	Download	Two new fields, `device_unique_id` and `ip_address` have been added to the Incrementality Feed.
0.5.24	July 22, 2021	Download	Download	Download	A new field, `postal_code_ext`, has been added to the Standard Feed.
0.5.22	July 21, 2021	Download	Download	Download	Added `device_id` field to the Curator Feed. These IDs values can be looked up by using the Xandr API Device Model Service.
0.5.21	June 18, 2021	Download	Download	Download	Added 3 new fields, `opera`

Version	Date Released	Schemas Zip	Example Files Zip	Example Code Zip (includes schemas + example files)	Notes
					`ting_syste m`, `browser`, and `languag e` to the Curator Feed. These values can be looked up by using the Xandr API Operating System Servic Service, and Language Service respe
0.5.20	May 20, 2021	Download	Download	Download	Added a new field, `device _make_id`, to the Standard Feed. The field contains the ID of the device make, which is generally the manufacturer of the device (e.g., Samsung). To map device make IDs to names, use the Device Make Service.

Version	Date Released	Schemas Zip	Example Files Zip	Example Code Zip (includes schemas + example files)	Notes
0.5.18	April 19, 2021	Download	Download	Download	Added enhancements to the Curator Feed (curator_feed).
0.5.15	March 31, 2021	Download	Download	Download	• Added new field, `personal_identifiers`, to the Standard Feed. This field of "repeated" type appears to both buyers and sellers for transacted, non-transacted and viewed impressions. • Added the initial version of

Version	Date Released	Schemas Zip	Example Files Zip	Example Code Zip (includes schemas + example files)	Notes
					the Curator Feed (`curator_feed`).

0.5.10	February 4, 2021	Download	Download	Download	• The following changes have been made to the the Buyer Transparency
					  • the following fields were added under the `bid_message` (index `9`):
					  • `external`

Version	Date Released	Schemas Zip	Example Files Zip	Example Code Zip (includes schemas + example files)	Notes
					• all_campaign_id • insertion_order_id

Version	Date Released	Schemas Zip	Example Files Zip	Example Code Zip (includes schemas + example files)	Notes
					• bidder_seat_id • bidder_seat_name • The

Version	Date Released	Schemas Zip	Example Files Zip	Example Code Zip (includes schemas + example files)	Notes
					`sasc_cap_savings` field was added under the `resultmessage` (index `10`). • The `custom_parameters` field (index `17`) in the Universal Pixel Feed been changed from an optional

Version	Date Released	Schemas Zip	Example Files Zip	Example Code Zip (includes schemas + example files)	Notes
					field to a repeated one. See the documentation for the individual feeds for more information on the fields that have been added.
0.5.7	April 6, 2020	Download	Download	Download	Added 4 new fields to the Universal Pixel Feed. The new fields are: • `traffic_type` - The source of the traffic being tracked by the

Version	Date Released	Schemas Zip	Example Files Zip	Example Code Zip (includes schemas + example files)	Notes
					pixel. Possible values are `WEB` or `APP` • `application_id` - The ID of the application (in the app store) that the pixel has been placed on. This value can be numeric or alphanumeric (e..g, `com.xandr.application_name`) • `device_unique_id` - The

Version	Date Released	Schemas Zip	Example Files Zip	Example Code Zip (includes schemas + example files)	Notes
					unique identifier representing the mobile device (The value of this field will be `null` except for specific integrations). The numeric prefix indicates the type of unique device identifier: • `0` = IDFA (Apple ID for Advertising) • `1` = SHA1 • `2` = MD5

Version	Date Released	Schemas Zip	Example Files Zip	Example Code Zip (includes schemas + example files)	Notes
					• `3` = ODIN • `4` = OPENUDID • `5` = AAID (Android Advertising ID) • `6` = WINDOWSADID (Microsoft Advertising ID). • `custom_parameters` - Contains all custom parameters that were sent with the pixel fire.
0.5.4	January 27, 2020	Download	Download	Download	Correction to the schema used to release the new

Version	Date Released	Schemas Zip	Example Files Zip	Example Code Zip (includes schemas + example files)	Notes
					Universal Pixel Feed.
0.5.2	November 7, 2019	Download	Download	Download	Added new `external_campaign_id` field to the Standard Feed in LLD. This new optional field should only appear to sellers on resold impression rows. The value of this field is passed in via the `cid` field on a DSP's bid. Since the `cid` field is optional, the new `external_campaign_id` field will only have data when the external DSPs populate it

Version	Date Released	Schemas Zip	Example Files Zip	Example Code Zip (includes schemas + example files)	Notes
					on their bid(s). See the Open RTB specification for more info on the `cid` field.
0.5.1	September 10, 2019	Download	Download	Download	• Added `partner_fees` to Standard Feed. • Added `partition_time_millis` field to all feeds to simplify the loading and partitioning of data into databases. • Added `hashed_user_`

Version	Date Released	Schemas Zip	Example Files Zip	Example Code Zip (includes schemas + example files)	Notes
					`id_64` field to Conversion Pixel and Segment feeds for clients who only want anonymized personal data.
0.4.4	May 29, 2019	Download	Download	Download	Added `tc_string` to standard feed.
0.3.4	April 10, 2019	Download	Download	Download	Added `split_id` to standard feed.
0.3.3	April 5, 2019	Download	Download	Download	• Added `hashed_user_id_64` to segment feed.

Version	Date Released	Schemas Zip	Example Files Zip	Example Code Zip (includes schemas + example files)	Notes
					• Added `hashed_user_id_64`, `latitude_trunc`, `longitude_trunc` to standard untransacted feed.
0.3.0	October 4, 2018	Download	Download	Download	• Added Avro schemas and example files • Added `partition_time_millis` column to all feeds for partition filtering

Version	Date Released	Schemas Zip	Example Files Zip	Example Code Zip (includes schemas + example files)	Notes
					per record
0.2.4	August 22, 2018	Download	Download	Download	• Removed `hashed_device_unique_id` from standard feed schema (no longer used) • Added schema for standard untransacted feed
0.1.9	April 12, 2018	Download	Download	Download	• Allow specifying protobuf version, e.g. `-Dprotobuf.version="2.5.0"`

Version	Date Released	Schemas Zip	Example Files Zip	Example Code Zip (includes schemas + example files)	Notes
					• Patched schema class finder to work with proto3 generated code
0.1.8	March 16, 2018	Download	Download	Download	Added proto2 syntax hint to make proto files compilable with proto3
0.1.7	March 12, 2018	Download	Download	Download	Protobuf-delimited is now GZIP-compressed, updated sample code accordingly
0.1.6	March 7, 2018	Download	Download	Download	Added anonymized personal data fields to various relevant LLD schemas

Version	Date Released	Schemas Zip	Example Files Zip	Example Code Zip (includes schemas + example files)	Notes
0.1.4	December 8, 2017	Download	Download	Download	Added imps_for_budget_caps_pacir column to standard_feed
0.1.3	October 16, 2017	Download	Download	Download	Initial release

Protobuf (Sequence file wrapped Protocol Buffers)

> **NOTE:**
> Only version 2.5.0 of protobuf is currently supported.

Files are Snappy compressed Hadoop Sequence files where the value for each record is a BytesWriteable, the payload of which is an encoded Protocol buffer message.

All schemas specify that fields are optional and `null` values are unset fields in the protobuf message. See the individual feeds under Log-level data feeds for the conditions that cause a field's value to be `null` and for more details on column availability.

See Protobuf Install and Configuration for instructions on how to install and configure the protobuf compiler and to download a project that includes the schemas and sample code.

Protobuf-delimited (Protocol Buffers)

> **NOTE:**
> Only version 2.5.0 of protobuf is currently supported.

Files are GZIP compressed files that contain length-delimited Protocol buffer messages. Each record is a varint specifying the length of the message, followed by the protobuf message itself. One reason to use our protobuf-delimited format instead of our protobuf format is that reading protobuf-delimited files does not require Hadoop or Hadoop with native Snappy support.

All schemas specify that fields are optional and `null` values are unset fields in the protobuf message. See the individual feed service pages for the conditions that cause a field's value to be `null` and for more details on column availability.

See Protobuf Install and Configuration for instructions on how to install and configure the protobuf compiler, and to download a project that includes the schemas and sample code.

Avro

Avro is a data serialization framework that bundles schemas with data. For compression, the DEFLATE codec (level = 1) is used. For more details, see https://avro.apache.org/docs/current/.

> **NOTE:**
> Unlike in our protobuf formats, `null` values are never used. Missing or unset fields are encoded with their default values, as specified in the feed schema.
> Avro is offered for simpler integration with existing third-party Cloud systems. Due to incompatibilities found while testing integrations, a field that is "`enum`" in protobuf is sent as Avro "`int`".