New Features in
ibaHD-Server v2.7.0

Oktober 2021
iba AG

iba AG • Königswarterstr. 44 • D-90762 Fürth • Germany • www.iba-ag.com

Table of Contents

1 Important Information

This document describes the new features of ibaHD-Server v2.7.0.
The main new feature for this version of ibaHD-Server is an extension of the API gRPC interface. Some minor other improvements are also part of this version as well as critical bug fixes.

2 API Interface

2.1 GetLastRecordedChannelValue()

Returns the last recorded raw data value for all requested channels within the specified time range. Especially for digital or text channels that change infrequently, the returned samples may point to timestamps in the past. This means that the value of the channel has not changed within the specified time range and is still valid.

Involved Messages

GetLastRecordedChannelValueRequest

Allows specifying the requested channels. The time range can be used to narrow down the search, e.g., to query the last recorded sample within a week. To retrieve the last value that was ever recorded, “time_range_to” can be set to a date in the future or int64 max value.

GetLastRecordedChannelValueResponse

Returns a single raw data point for each requested channel. If no sample was found for a given channel, the response will not contain any data this channel.

2.2 GetLastEventOccurence()

Returns the last occurrence for all requested events within the specified time range.

Involved Messages

GetLastEventOccurrenceRequest

Allows specifying the requested event channels. The time range can be used to narrow down the search, e.g., to query the last event occurrence within a week. To retrieve the last event occurrence that was ever recorded, “time_range_to” can be set to a date in the future or int64 max value.

GetLastEventOccurrenceResponse

Returns a single event for each requested event channel. If no event was found for a given event channel, the response will not contain any data for this event channel.

2.3 Diagnostic stores

The GetHdStoreRequest is extended to optionally include diagnostic stores. Diagnostic stores can be recognised by the following format: "<DIAGNOSTIC>StoreName”.

2.4 Linear interpolation downsampling

GetAggregatedChannelData() provides a way to query aggregated channel data for time-based channels. It allows specifying the number of samples that should be sent for a given time range, as well as the selection of min, max and average aggregations.
The aggregation algorithm can be changed from the default “Min/Max/Avg downsampling” algorithm

to the alternative “Linear interpolation downsampling” algorithm. This algorithm tries to select raw data points in an equidistant manner (requested time range / requested samples) and applies linear interpolation between adjacent samples if the stored samples do not exactly match the target sample timestamps.

2.5 is_backup property GetHdStores()

GetHdStores() allows to retrieve a list of HD stores from the server. Now the response contains an is_backup property to indicate if the returned HD store is a mounted backup or a normal HD store.

2.6 Return an extra sample before and after the request

To properly draw charts, it is required to have an additional sample outside of the requested time range. An extra sample before the requested time range is supported for all channel types while an extra sample after the requested range is only supported for float and double channels. This can be enabled optionally by setting the “add_extra_sample_out_of_time_range” option in both GetRawChannelData() and GetAggregatedChannelData().

2.7 Identification of gaps in the requested data

Sometimes gaps exist in the data due to e.g., a stop of the data acquisition. When the ibaHD-API requested a time range containing a gap, there was no clear indication data was missing. The response data would just omit timestamps with no data.
To improve this gap identifiers are now returned. When a gap starts, a single NaN value is included in the response message on the first timestamp without data. This indicates no data is available until the next returned valid value. If the NaN marker is the last returned data point, this means the gap runs until the end of the requested time range.

3 Optimized the communication between ibaHD clients and server

When requesting the data for signal trees, the ibaHD-Server would send the data of all configured stores into a single message to the clients. This could lead to timeouts when a large number of stores and signals are configured. For ibaHD-Server 2.7.0, this communication is optimized by sending the data of each store in separate messages with a maximum of 5000 signals per message. If more signals are configured for this store, additional messages containing the remaining signals are sent.
The maximum number of signals default set to a value of 5000 but can be changed through the "ReaderMaxSignals" registry value with key:
“HKEY_CURRENT_USER\SOFTWARE\iba\ibaHD\Client”.

This feature requires the use of ibaPDA 7.3.8, ibaAnalyzer 7.3.3 and ibaDatCoordinator 2.4.2.