Luminate Web Services Release 1.20

Version 1.20 of Luminate Web Services is now available. This release addresses several defects and exposes the Status field in CalendarEvent. It also increases the maximum number of rows returned by an incremental sync or query request from 200 to 1,000. Please see release notes below for details.

Defect Fixes

60471	Documentation: TeamRaiserTeam "isPrivate" returns "false" when a team has a password
63226	Expose CalendarEvent Status (Draft, Public, Private)
63823	Documentation: Product Description field values are not available
64636	Documentation: CalendarEvent Description field values are not available
65024	Documentation: SurveyResponse response values are not available for 'Unlimited Text' and 'Constituent Registration Info' question types
67651	Unexpected server error when querying SecurityCategoryId from both Donation and Donation.Payment
67740	Donation.GetIncrementalDeletes returns an empty result set when certain fields are included
67934	Unexpected server error when querying against SurveyQuestion
68087	Unexpected server error when including both HomeAddress.County, HomeCounty in a query

Application Behavior Changes

Added **CalendarEventStatus** (DRAFT | PUBLIC | PRIVATE) to **CalendarEvent**.

Added **ExchangeLogLastGoodSync** type for internal use to allow Blackbaud Data Warehouse to coordinate its syncyhronization state with the Blackbaud Professional Services custom DataSync between Luminate Online and Blackbaud CRM.

- **OperationId** - System generated unique ID for the DataSync operation.
- **DateStarted** - DataSync start time stamp.
- **DateCompleted** - DataSync completion time stamp.

Increased the maximum PageSize limit from 200 rows to 1,000 rows for download methods (Query, Find, GetIncrementalInserts, GetIncrementalUpdates, GetIncrementalDeletes).

Known Issues

Bug #56649 — Specifying a PageSize parameter for a GetIncrementalInserts, GetIncrementalUpdates, GetIncrementalDeletes operation that is different from the PageSize parameter specified for the previous corresponding GetIncrementalInsertsCount, GetIncrementalUpdatesCount, or GetIncrementalDeletesCount operation in a sync session results in unexpected behavior. If a larger PageSize is specified in subsequent calls, only the number of records specified in corresponding GetIncremental…Count call is returned. If a smaller PageSize=*n* is specified, only the first *n* records are returned for each Page resulting in records being skipped in the page sequence. To work around this issue, always specify the same PageSize for all GetIncremental* calls within the same sync session.