

Totem Unified Database Access Service

Client-side Interfaces

Bartłomiej Alberski

Michał Idzik

Bartosz Niemczura

23 sierpnia 2012

1 Collimation Alignment Production

- `int saveOffsets(long startTime, string label, map<string, double> offsets)`

Description: Saves data from motor step counter offsets file into the database. Offsets can be further identified by a given validity interval and a label. The method returns version number of saved data (in case of multiple similar inserts the old one is not removed, but new version of the same data is created instead).

Parameters:

- *startTime* - a validity interval start time.
- *label* - a text label, connected with offsets data type, i. e. optics (*beta90*).
- *offsets* - offset data map. It should contain roman pot string labels in format: STATION_ARM_POSITION (i. e. *RP_220_Left_Det_Dist_0*) paired with offset values.

- `map<string, double> loadOffsets(long timestamp, string label)`

Description: Loads the latest version of offset data (bound to given timestamp and label) from the database and returns it as a roman pot label - offset value map.

Parameters:

- *timestamp* - a timestamp value, it should be between *startTime* and *endTime* specified in *saveOffsets* method.
- *label* - a text label, connected with offsets data type, i. e. optics (*beta90*).

- `map<string, double> loadOffsetsByVersion((long timestamp, string label, int version)`

Description: Loads the specified version of offset data (bound to given timestamp and label) from the database and returns it as a roman pot label - offset value map.

Parameters:

- *timestamp* - a timestamp value, it should be between *startTime* and *endTime* specified in *saveOffsets* method.
- *label* - a text label, connected with offsets data type, i. e. optics (*beta90*).
- *version* - a version number, positive integer value.

2 Geometry Production

- **int** *saveGeometry*(**long** timestamp, **string** label, **map**<**string**, **double**> geometryValues)

Description: Saves computed geometry values into the database. All data should be bound to specific timestamp and label in order to correctly identify in the database. The method returns version number of saved data (in case of multiple similar inserts the old one is not removed, but new version of the same data is created instead).

Parameters:

- *timestamp* - timestamp value in miliseconds, connected with this data.
- *label* - a text label, connected with geometry data type, i. e. optics (*beta90*).
- *geometryValues* - geometry data map. It should contain roman pot string labels in format: STATION_ARM_POSITION (i. e. *RP_220_Left_Det_Dist_0*) paired with geometry values.

- **map**<**string**, **double**> *loadGeometry*(**long** timestamp, **string** label)

Description: Loads the latest version of geometry data (bound to given timestamp and label) from the database and returns it as a roman pot label - geometry value map.

Parameters:

- *timestamp* - timestamp value in miliseconds, connected with this data.
- *label* - a text label, connected with geometry data type, i. e. optics (*beta90*).

- **map**<**string**, **double**> *loadGeometryByVersion*(**long** timestamp, **string** label, **int** version)

Description: Loads the specified version of geometry data (bound to given timestamp and label) from the database and returns it as a roman pot label - geometry

value map.

Parameters:

- *timestamp* - timestamp value in miliseconds, connected with this data.
- *label* - a text label, connected with geometry data type, i. e. optics (*beta90*).
- *version* - a version number, positive integer value.

3 Alignment Calculations

- **int** *saveAlignments*(**long** startTime, **long** endTime, **string** label, **vector**<**RomanPotAlignment**> alignments)

Description: Saves alignment values into the database. Alignment data is stored in *RomanPotAlignment* structure (described below) and is bound to specified validity interval and label. The method returns version number of saved data (in case of multiple similar inserts the old one is not removed, but new version of the same data is created instead).

Parameters:

- *startTime* - a validity interval start time.
- *endTime* - a validity interval end time.
- *label* - a text label, connected with alignments data
- *alignments* - the alignments vector, a set of *RomanPotAlignment* structures.

- **int** *saveAlignmentsByRun*(**int** startRun, **int** endRun, **string** label, **vector**<**RomanPotAlignment**> alignments)

Description: Saves alignment values into the database. Alignment data is stored in *RomanPotAlignment* structure (described below) and is bound to specified validity interval and label. The method returns version number of saved data (in case of multiple similar inserts the old one is not removed, but new version of the same data is created instead).

Parameters:

- *startRun* - a validity interval start run.
- *endRun* - a validity interval end run.
- *label* - a text label, connected with alignments data
- *alignments* - the alignments vector, a set of *RomanPotAlignment* structures.

- **map**<**IntervalOfValidity**, **vector**<**RomanPotAlignment**>> *loadAlignments*(**long** startTime, **long** endTime, **string** label)

Description: Loads the latest version of alignments data from the database (bound to a given time interval and label) and returns them as *RomanPotAlignment*

vectors connected with validity intervals.

Parameters:

- *startTime* - a time interval start.
- *endTime* - a time interval end.
- *label* - a text label, connected with alignments data

- **map<IntervalOfValidity, vector<RomanPotAlignment>>** *loadAlignmentsByRun*(**int** startRun, **int** endRun, **string** label)

Description: Loads the latest version of alignments data from the database (bound to a given run interval and label) and returns them as RomanPotAlignment vectors connected with validity intervals.

Parameters:

- *startTime* - a run interval start.
- *endTime* - a run interval end.
- *label* - a text label, connected with alignments data

- **map<IntervalOfValidity, vector<RomanPotAlignment>>** *loadAlignmentsByVersion*(**long** startTime, **long** endTime, **string** label, **int** version)

Description: Loads the specific version of alignments data from the database (bound to a given time interval and label) and returns them as RomanPotAlignment vectors connected with validity intervals.

Parameters:

- *startTime* - a time interval start.
- *endTime* - a time interval end.
- *label* - a text label, connected with alignments data
- *version* - a version

- **map<IntervalOfValidity, vector<RomanPotAlignment>>** *loadAlignmentsByRunAndVersion*(**int** startRun, **int** endRun, **string** label, **int** version)

Description: Loads the specific version of alignments data from the database (bound to a given run interval and label) and returns them as RomanPotAlignment vectors connected with validity intervals.

Parameters:

- *startTime* - a run interval start.
- *endTime* - a run interval end.
- *label* - a text label, connected with alignments data

- *version* - a version
- *RomanPotAlignment* - a structure representing single alignment.
Fields:
 - **string** *romanPotLabel* - a label describing a roman pot structure, in number format (replacing insignificant zeros in the beginning to spaces), i. e. ' 103'.
 - **map<string, double>** *alignmentValues* - a value map, describing alignment attributes and their values. The keys are attribute labels (i. e. *sh_r*).
- *IntervalOfValidity* - a structure representing single validity interval.
Fields:
 - **long** *startTime* - validity interval start time.
 - **long** *endTime* - validity interval end time.