# Database datasetindex

## Table “*datasetindex*”

Each record describes a dataset served by the Panoptes instances that reads from this database.

### Table columns

* *id*, varchar(20). Unique identifier for the data set.
* *name*, varchar(50). Display name of the data set.

## Table “*calculations*”

Each record correspond to a calculation that was done or is being done on the server. The state of the calculation (in progress, completed, failed) is also stored in the record.

### Table columns

* *id*, varchar(50). Unique identifier of the calculation.
* *user*, varchar(50). User id of the person who started the calculation.
* *timestamp*, varchar(50). Time stamp of the start of the calculation.
* *name*, varchar(300). Descriptive name.
* *status*, varchar(300). Text describing the current calculation activity.
* *progress*, float. Progress indication as a fraction.
* *completed*, int. Completed: 1, else: 0.
* *failed*, int. Failed: 1, else: 0.
* *scope*, varchar(100).

## Table “*storedviews*”

Each record contains a stored snapshot view of a Panoptes instance, that can be accessed by a permanent url.

### Table columns

* *dataset*, varchar(100), FK: *datasetindex.id*. Specifies the dataset the instance is based on.
* *workspace*, varchar(100), FK: *{datasetid}.workspaces.id*. Specifies the workspace in the dataset the instance is based on.
* *id*, varchar(100). Unique identifier of the stored view.
* *settings*, text. Serialised settings string, specifying the instance state.

## Table “*storage*”

Standard utility table used by the DQX server storage mechanism.

### Table columns

* *id*, varchar(50)
* *content*, text

# Database {datasetid}

## Table “*tablecatalog*”

A Panoptes dataset consists in a set of datatables being served. Each record in *tablecatalog* describes an individual datatable in the dataset.

### Table columns

* *id*, varchar(20). Unique identifier.
* *name*, varchar(50). Display name of the datatable.
* *primkey*, varchar(20). Name of the column that contains the primary key of the datatable.
* *IsPositionOnGenome*, int. Indicates if this datatable contains records that refer to positions on the genome (1 if true, 0 if not). Note that, if true, the corresponding table should have the columns “*chrom*” and “*pos*”.
* *settings*, varchar(2000). JSON object containing further settings.

### Settings tokens

* “*QuickFindFields*": A subset of the datatable properties that are used in the “Find record” wizard. Formatted as a comma separated string.
* "*GenomeMaxViewportSizeX*”. (Only applicable to datatables that contain genomic positions). Maximum viewport size (number, in bp) for which individual records are displayed in the genome browser.

## Table “*workspaces*”

### Table columns

* *id*, varchar(50)
* *name*, varchar(50)

## Table “*{tableid}*”

Table columns:

## Table “*{tableid}INFO\_{workspaceid}*”

Table columns:

## Table “*{tableid}CMB\_{workspaceid}*”

Table columns:

## Table “*annotation*”

Table columns:

* *chromid*, varchar(20)
* *fstart*, int
* *fstop*, int
* *fid*, varchar(20)
* *fparentid*, varchar(20)
* *ftype*, varchar(20)
* *fname*, varchar(100)
* *fnames*, varchar(200)
* *descr*, varchar(100)

## Table “*chromosomes*”

Table columns:

* *id*, varchar(20)
* *len*, float

## Table “*externallinks*”

Table columns:

* *linktype*, varchar(20)
* *linkname*, varchar(50)
* *linkurl*, varchar(200)

## Table “*propertycatalog*”

Table columns:

* *workspaceid*, varchar(50)
* *source*, varchar(50)
* *datatype*, varchar(20)
* *propid*, varchar(50)
* *tableid*, varchar(20)
* *name*, varchar(50)
* *ordr*, int
* *settings*, varchar(1000)

## Table “*settings*”

Table columns:

* *id*, varchar(20)
* *content*, varchar(2000)

## Table “*summaryvalues*”

Table columns:

* *workspaceid*, varchar(50)
* *source*, varchar(20)
* *propid*, varchar(20)
* *tableid*, varchar(20)
* *name*, varchar(50)
* *ordr*, int
* *settings*, varchar(2000)
* *minval*, float
* *maxval*, float
* *minblocksize*, int

## Table “*tablebasedsummaryvalues*”

Table columns:

* *tableid*, varchar(50)
* *trackid*, varchar(50)
* *trackname*, varchar(50)
* *settings*, varchar(5000)
* *minval*, float
* *maxval*, float
* *minblocksize*, int

# File structure

## Helper directories

Uploads

## Genome filterbank tracks

SummaryTracks/{datasetid}

## Table-related Genome filterbank tracks

SummaryTracks/{datasetid}/TableTracks/{tableid}/{trackid}/{recordid}