

QualDash 2.0 Developer Documentation

1 CONTENTS

2	Overview	2
3	Browser Support.....	2
4	User Interface.....	3
4.1	Getting started.....	3
4.2	The Home tab: main dashboard	4
4.2.1	Layout controls.....	4
4.2.2	Data Quality Pane.....	5
4.2.3	QualCard: Default View	6
4.2.3.1	Control Panel.....	6
4.2.3.2	Main View	7
	Visualization Design	7
	Mark types:	7
	Axes:	7
	Colour	8
	Interaction Design	8
	Selection	8
4.2.4	QualCard: Expanded View	8
4.2.5	QualCard Sub-views.....	10
4.2.5.1	The Pie sub-view.....	10
	GUI Customization	10
	Visualization Design	12
	Mark type.....	12
	Axes	12
	Colour	12
	Labels.....	12
	Interaction Design	12
	Mouse hover:	13
	Mouse selection:	13
4.2.5.2	The Bar sub-view	15
	GUI Customization	15
	Visualization Design	16
	Mark type.....	16

Axes	16
Colour	16
Interaction Design	16
Mouse hover:	16
Mouse selection:	17
4.2.5.3 The History sub-view	17
Visualization Design	17
Time Series View	17
Small Multiples View	17
Interaction Design	17
4.3 Tabular View	18
4.4 Data Exports	18
4.4.1 Exporting Raw Data	19
4.4.2 Exporting Visualizations	19
5 Backend and API	19
5.1 Preparing Data and Metadata	19
5.1.1 Metadata Files	19
5.1.2 Picanet Audit Files	20
5.1.3 MINAP Audit Files	20
5.2 Loading Data into QualDash	21
5.2.1 Metric Specification Structure (MSS)	21
5.3 Session Logging	22

2 OVERVIEW

QualDash is a web-based tool that generates adaptable visualization dashboards for clinical audit data. Given a dashboard specification file, QualDash generates a corresponding number of QualCards (i.e. draggable panels) which can be dragged and rearranged in different layouts and expanded / shrunk to reveal / hide visualizations. QualDash uses JavaScript libraries including [Bootstrap.js](#), [Muuri.js](#) and [D3.js](#). A live demo using a simulated dataset can be accessed [here](#).

3 BROWSER SUPPORT

QualDash 2.0 is supported and accessible from any computer in the site using a Google Chrome web browser. QualDash 2.0 has not been tested on any other web browsers and does not work on Internet Explorer.

4 USER INTERFACE

4.1 GETTING STARTED

To get started with QualDash, navigate to the dashboard's URL from a web browser. The URL will vary depending on server settings at your site. If you have any questions about where to access the dashboard, please contact your site's IT staff. Once loaded, QualDash's welcome page should appear:

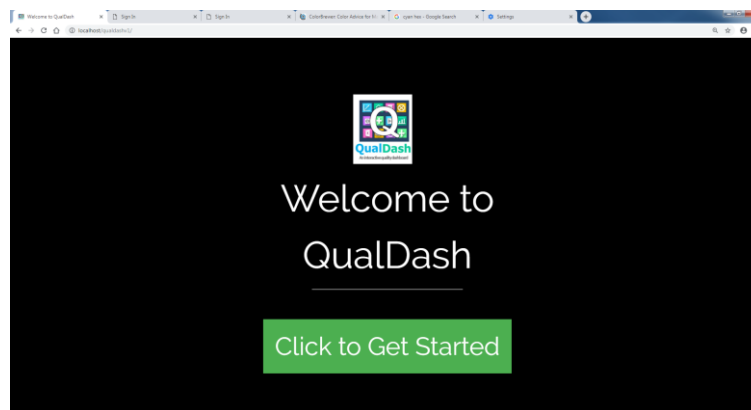


Figure 1. QualDash Welcome Page.

Once the “Click to get started” button is pressed, the user is prompted to select: (a) an audit to display, (b) a year for data to be loaded and (c) to type in their job title. All three fields are required to proceed.

Audit options are currently limited to two audits: PICANet or MINAP.

Year options depend on audit data available at your site. The selected year will cause the dashboard to load the data for the specified calendar year in the main view of each QualCard. For audits which require financial year to be visualized, the available financial year data will appear as two years separated with a ‘-’ in the dropdown menu for year selection (e.g. 2018-2019).

A history sub-view in each QualCard will also display historic data from two years prior to the selected year for comparison, if such data is available. If no historic data is available, a message will be displayed in the history sub-view instead. See Section 1.5 for more details on the history sub-view.

The job title selected does not affect the contents of the dashboard in any way. This field is only included for data collection purposes. To avoid entering your job title every time you launch QualDash, you may set your browser settings to remember your job title by performing two simple steps:

- a. Open Chrome Settings (this is done via the three dots on the top right corner of the Chrome window)
- b. Under "Autofill" --> "Addresses and more" --> check that the "save and fill addresses" option is switched on.

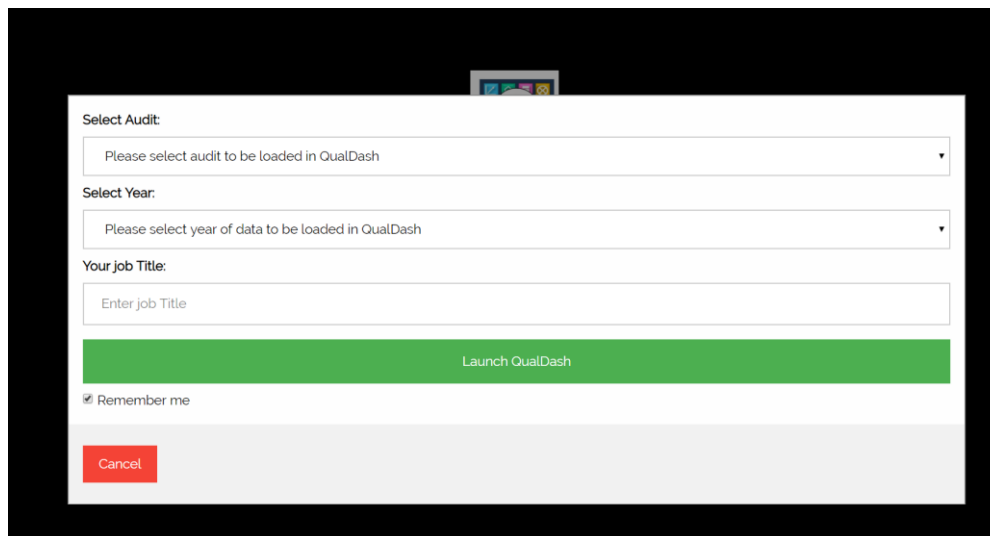


Figure 2. QualDash's launch screen

4.2 THE HOME TAB: MAIN DASHBOARD

Upon loading the dashboard, a number of QualCards will appear. In their initial form, QualCards occupy a subdivision of the screen depending on layout preferences that can be edited by the user. QualCards can be dragged, expanded and customized as will be described in this section.

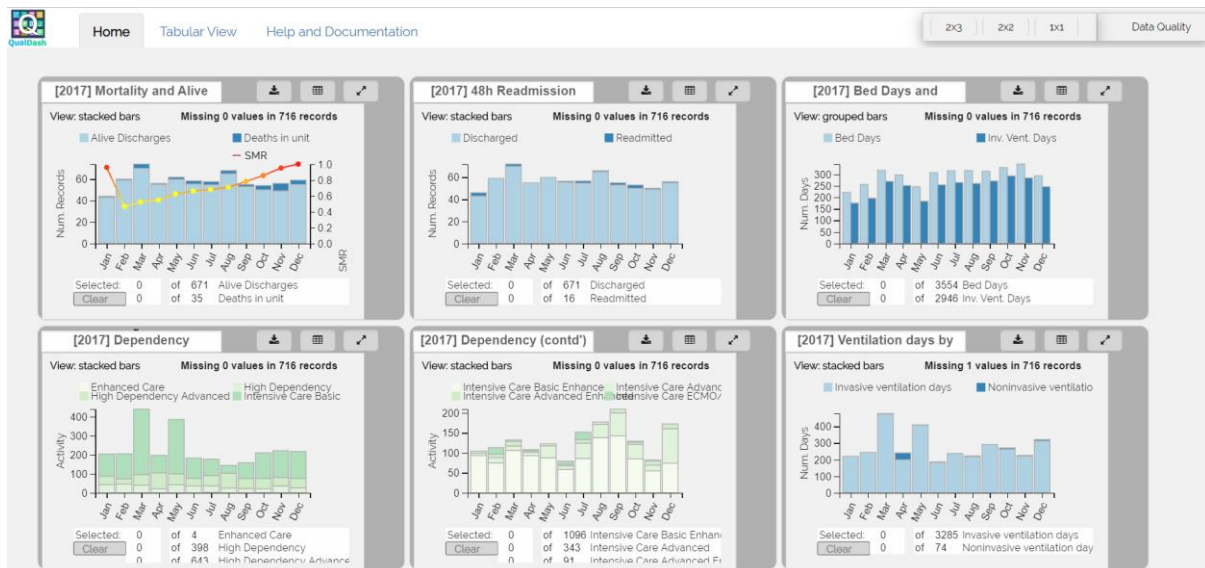


Figure 3. The main dashboard view of QualDash (based on simulated data only).

4.2.1 Layout controls

Three layout buttons exist at the top right corner of QualDash's interface. The first two buttons rearrange and resize QualCards on the screen to tile them in either a 2x3 or a 2x2 formation. Additional QualCards that do not fit within this formation can be viewed by scrolling down on the dashboard page.



Figure 4 Layout options 2x3 and 2x2

The third layout button creates a 1x1 arrangement, which means it maximizes and expands a QualCard to let it fill an entire screen. Navigating from one QualCard to another in this view is performed by scrolling down on the dashboard page.



Figure 5 Layout option 1x1

Note that in this expanded view, each card displays multiple visualizations in addition to the card's main visualization view. For a description of the different visualization views and their designs, please refer to Sections 2.4 and 2.5 of this documentation.

4.2.2 Data Quality Pane

Adjacent to the layout buttons described in the previous section, a "Data Quality" button is located at the top right corner of the main dashboard screen. This button displays a side pane that shows the missingness in the data. Specifically, the data quality pane assesses the fields in the database and sorts them in descending order by the number of missing values in each field.

For fields shown on the data quality pane, a horizontal bar shows the percentage of missing values in each one of the displayed fields. If a field is missing in less than 25% of the records, its bar is shown in

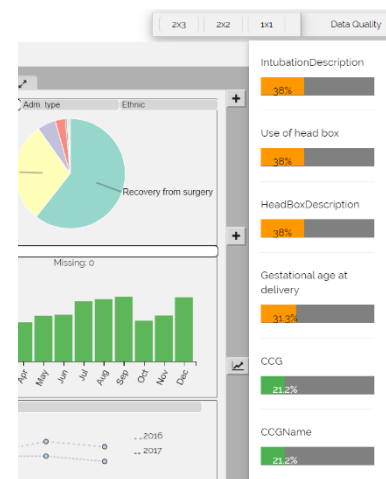


Figure 6 Data quality pane.

green. If this percentage increases to more than 25% but below 75%, the bar is shown in amber and if the percentage of missing values exceeds 75%, the bar is red.

4.2.3 QualCard: Default View

The default view of a QualCard has two main components: (a) a control panel and (b) a main visualization view (or “main view” for short).

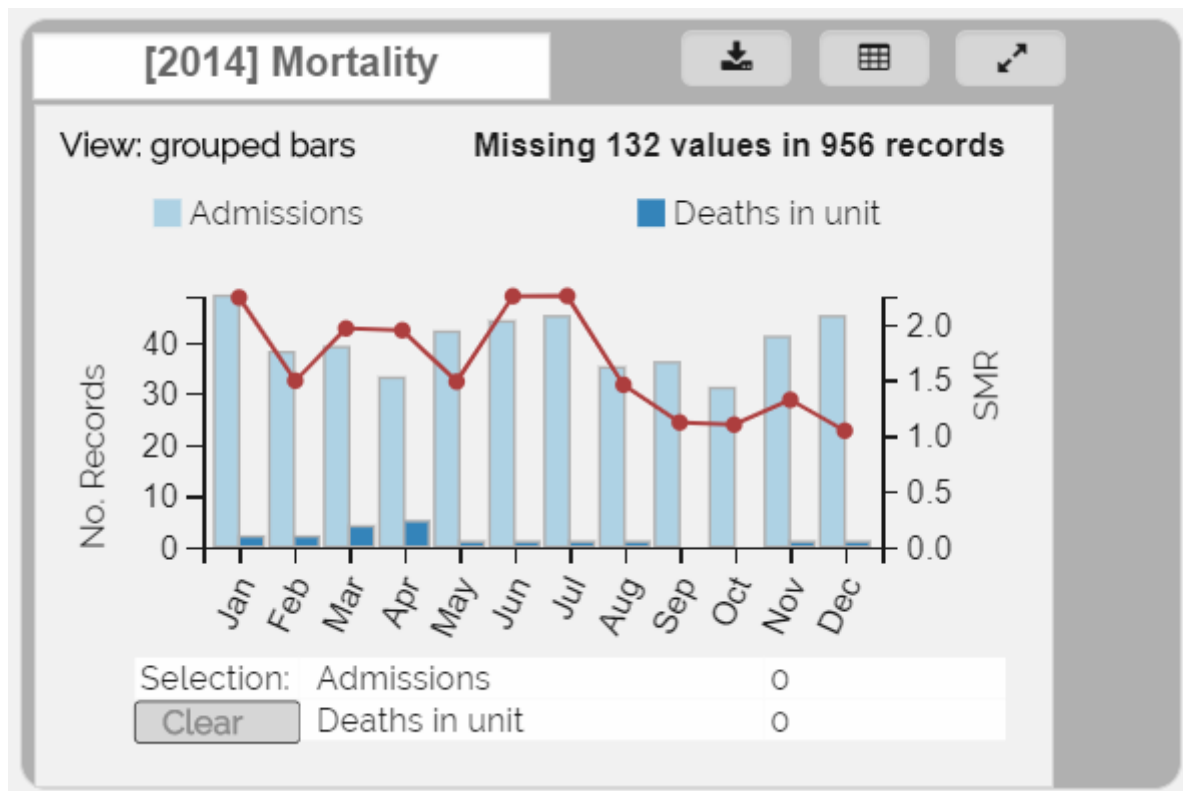


Figure 7 A QualCard's main view

4.2.3.1 Control Panel

This is the grey area that outlines the QualCard. It contains a label displaying the year of audit data loaded, the name of the metric that is shown on the card and a set of buttons (controls) to perform the following set of functions on the card:

- Download the visualization(s) shown in the main view.
- Export a selection to tabular view
- Expand / restore the card.

Changing a QualCard between its two states (expanded / restored) can be done by either double clicking anywhere on the grey space of the control panel or by clicking the expand button (top right button with two diverging arrows).

See Sections 2 and 3 of this part of the documentation for details of the functionality of the “Export to table” and “Download” buttons, respectively.

4.2.3.2 Main View

The main view of a QualCard displays:

- a) View information: the type of visualization displayed by the card. Chart types for the main view are currently restricted to either grouped bars or stacked bars. An additional y-axis can be specified, in which case a line is used for the measure displayed on the additional y-axis.
- b) Data missingness: the number of missing values encountered as the records used to generate this card's visualization were read from the audit database and the total number of records which exist in the database.
- c) Visualization legend: lists the names of variables (measures) shown in the main visualization and their colour encoding.
- d) Visualization chart: the main chart is displayed
- e) Selection pane: lists the names of the variables existing in the main visualization and the number of records selected in each from the visualization. The "Clear" button can be used to clear all selection.

Visualization Design

This section summarizes the technical design decisions made to visually encode data which can be loaded into the main view of a QualCard.

Mark types:

Bars (i.e. rectangles) are the main mark type of the main view. Data variables fed to each QualCard are displayed as aggregates (e.g. sum, average, count, etc.) on a monthly basis. Each aggregated value for a variable is assigned a bar in the chart and is given a colour. The number of colours used in each bar chart is therefore equal to the number of variables (measures) needed for this QualCard's corresponding metric.

In addition to the *bar* mark type, a *line* and *circle* mark types are added to a QualCard's main view if a QualCard's specification involves a variable that uses a different scale from the previously described aggregates. In this case, a dual axis is added to the chart and a line overlay is used to display the additional variable. See Figure 7 for an example.

Axes:

The x-axis in the main view represents time. By default, the time unit used in the main view of a QualCard is a month.

The y-axis represents a common scale used for all aggregates shown. We allow up to two different scales to be supported: one on the main y-axis and one on an optional dual axis.

Colour

A maximum of 9 different colours are allowed to be assigned to bars in the main view (Figure 8). This is the maximum number of colours that a qualitative printer-friendly palette can have on ColorBrewer (www.colorbrewer2.org). If the number of measures / categories that need to be shown on a QualCard exceeds this number, the chart is split into small multiples, where each chart independently displays one of the measures.

In addition to colours used for the bar mark type, the line and circle marks used for an overlay in case of a dual axis) always use a heatmap colour scheme which is distinct from the colour palette used for bars (Figure 5).

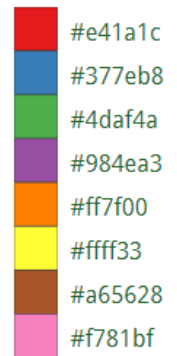


Figure 8 colour palette for bars in main view

Interaction Design

Interactions with the main view include highlights and selections. Highlights are performed by hovering the mouse on any of the visual marks on the chart. A highlighted visual mark is emphasized either by shading it with a cyan colour (in the case of bars) or increasing its size (in the case of circles on a line). Mouse hovers also display the value underlying each mark.

Selection

Selection is performed by clicking a visual mark. This functionality is currently only supported for bars in the main view. Selecting a bar will update the selection counts for the corresponding variable in the selection pane. Deselecting a bar can be done by clicking it again. A clear button can be used to clear all selections in a main view.

A selection can be exported to a tabular view to enable the user to view details-on-demand for the selected records. The export to table button on the QualCard's control panel is disabled by default. However, once the user selects records by interacting with the bar chart, the export button is enabled and clicking it will export the selection to the tabular view tab (See Section 2 of this documentation).

4.2.4 QualCard: Expanded View

The expanded view of a QualCard is obtained by double-clicking on the card's control panel (grey container) or by clicking the "expand button". To expand all cards at once, click on the 1x1 layout button (see Section 1.1 of this documentation). The expanded version of the QualCard reveals more variables that relate to the card's measure. Users can interact with these variables to understand categories in the data and make comparisons between different quantities. Therefore, in its expanded form a QualCard displays three sub-views in addition to the main view of the card.

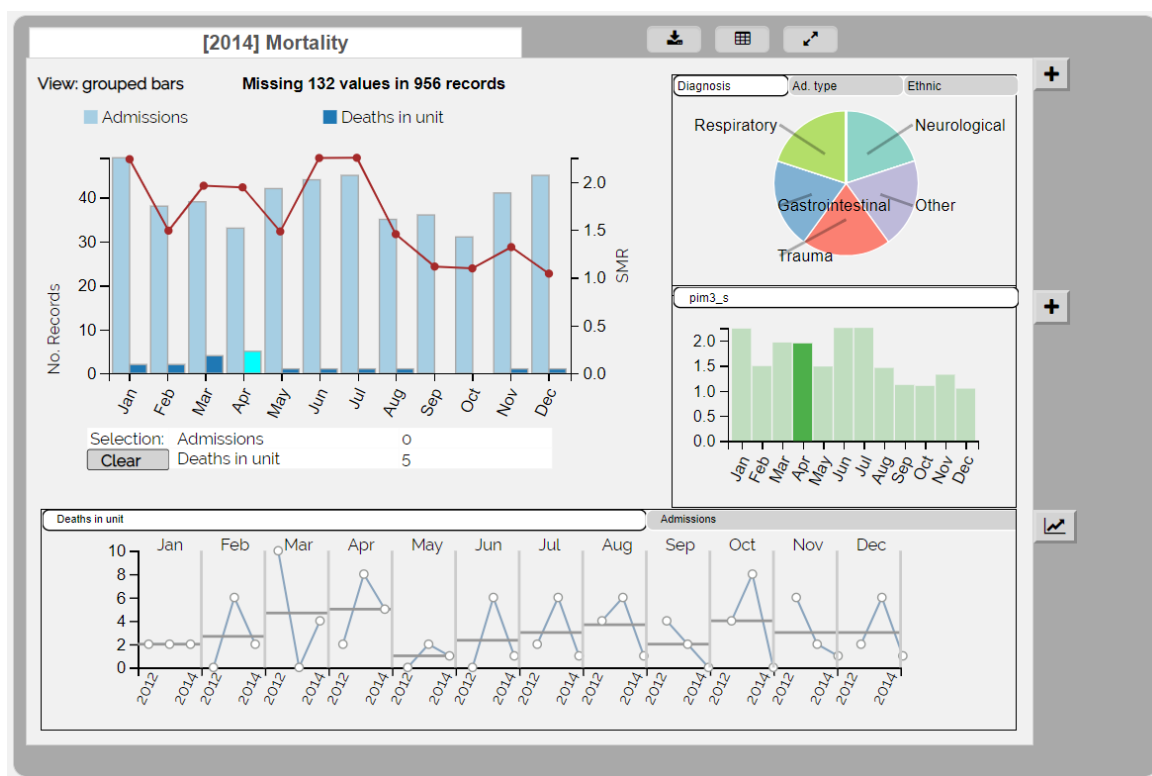


Figure 9 Expanded QualCard contains the main view and three sub-views and supports brushing and linking.

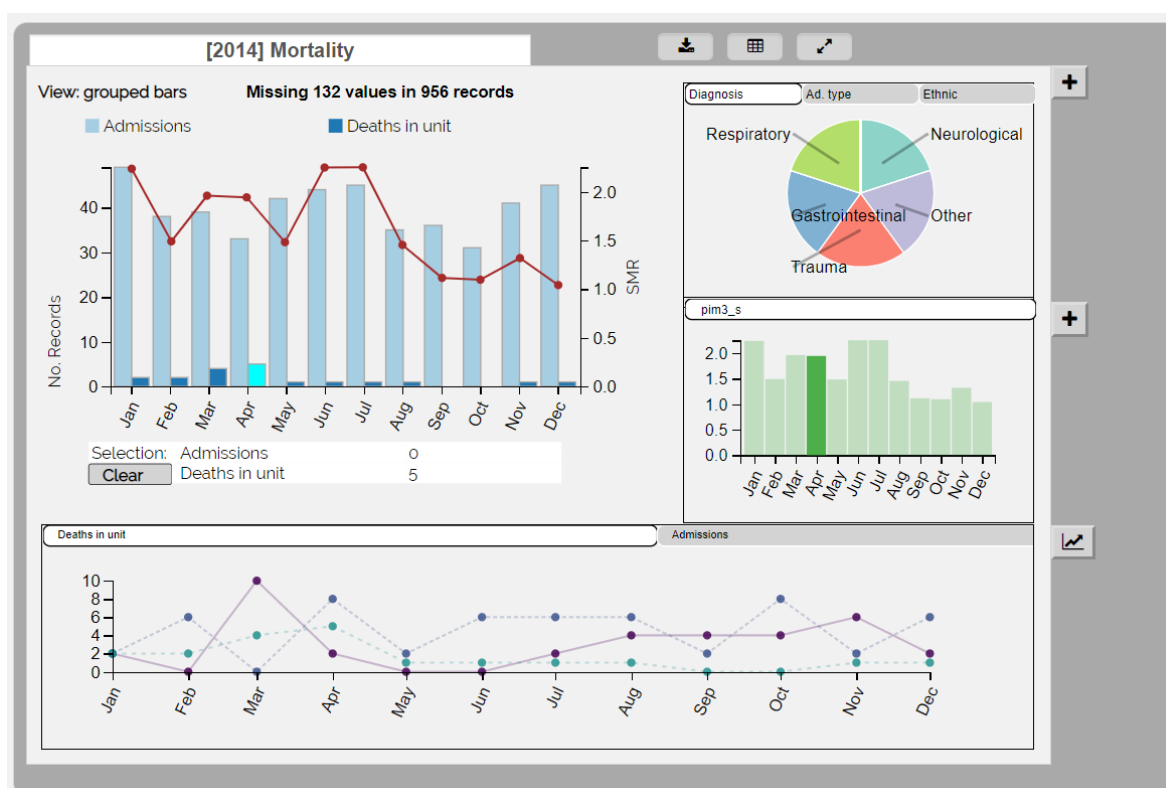


Figure 10 History sub-view can be toggled to show simple time series

4.2.5 QualCard Sub-views

Sub-views of a metric card contain customizable tabs that can be used to display additional variables from the dataset and link those variables to the measures shown in the main view of the card. The tabs of each sub-view are set to a default that is specified in a metric specification structure (MSS), which is loaded with the JavaScript of the dashboard. For details of the MSS please refer to Section IV in this documentation. These defaults are set according to the preferences of each site. Once loaded, the QualDash interface allows the user to further customize the tabs by selecting other variables to display in the corresponding sub-view. This section describes the visualizations and customization options for each one of these sub-views.

4.2.5.1 The Pie sub-view

The Pie sub-view is displayed on the top right corner of the expanded QualCard (see Figure 8). The tabs in the sub-view display a number of categorical variables from the database that have been specified in the MSS of the corresponding card. This default selection of variables can be further customized by the user through the Graphical User Interface (GUI) of the dashboard. In this section, we describe this GUI-based customization and explain the visualizations and interactions supported by QualDash in this sub-view.

GUI Customization

Clicking on a tab displays the variable's corresponding data column as a pie chart visualization in the Pie sub-view. A (+) button exists at the top right corner of the pie sub-view (See Figure 10). When clicked, this button displays a popover element that can be used to customize the variables displayed in the tabs. The popover element shows two multi-select lists of variables.

The list on the left contains all variables in the database that have a categorical (i.e. nominal or ordinal) data type (data types are specified in a metadata file that is stored in the /data folder. See Section 1 of Part IV of this documentation for details on data storage).

The list on the right hand side of the popover contains variables that are currently displayed in the Pie sub-view tabs. There are two required steps to update tabs in the sub-view:

Step 1: add / remove variables from selection:

To add new variables:

- Select one or more variable(s) from the left hand side list and click the right arrow button to move your selection into the selected variables list.

To remove variables:

- Select one or more variable(s) from the right hand side list and click the left arrow button to move your selection out of the selected variables list.

Step 2: confirm selection

Once happy with the list of variables to be displayed in the Pie sub-view, click on the "Update" button to confirm your selection.

Note that your changes will not be saved if you don't click the "Update" button. Clicking away from the popover will close the popover and will not save your changes. Therefore, it is crucial to press "Update" before navigating away from the popover.

Note also that your changes will not take effect if you select too many or too few variables to display on tabs. The Pie sub-view currently supports between 1 and 5 variables at a time. At the time when you press "Update", if the list of selected variables (right hand side) has less than 1 or more than 5 variables, QualDash will issue an error message and ask you to add the appropriate number of variables (See Figure 11).

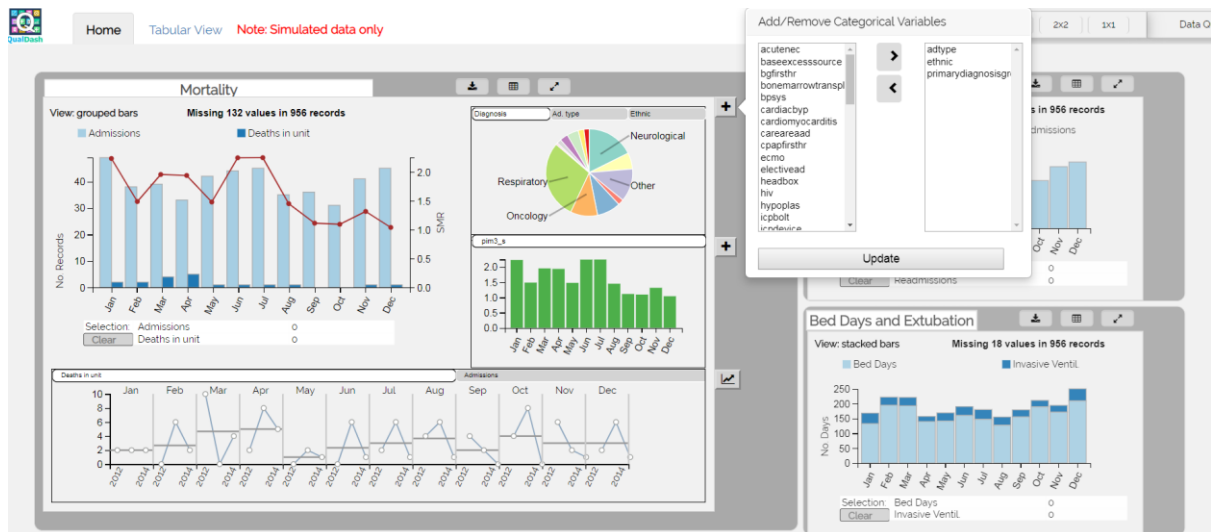


Figure 11 Customizing tabs of Pie sub-view

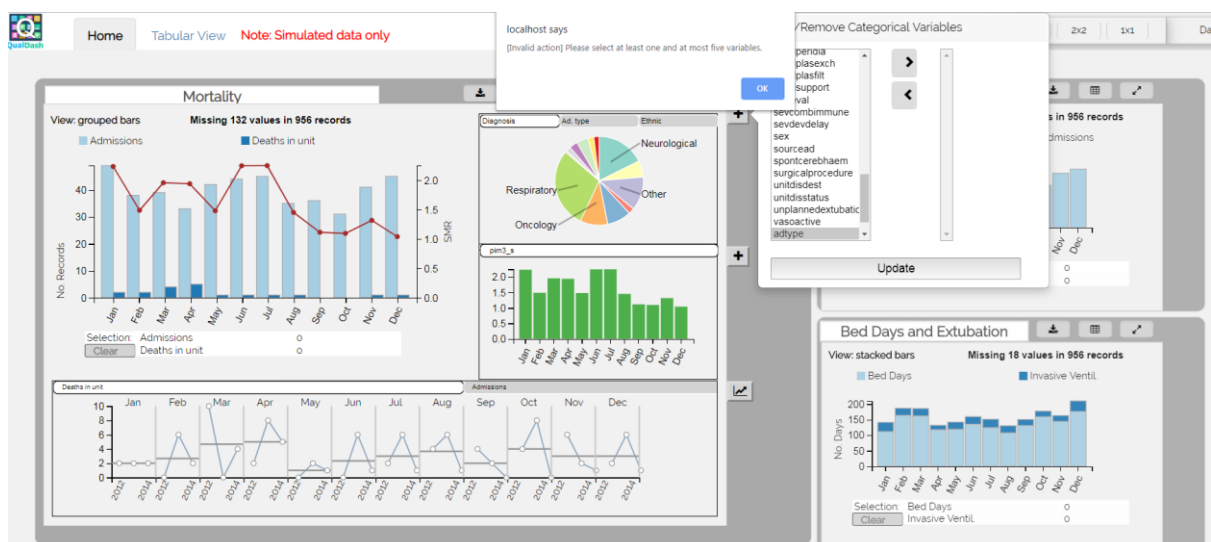


Figure 12 Error message is issued if the number of selected variables is outside valid range (1-5).

Visualization Design

Mark type

The only mark type supported in the Pie sub-view is “arc”, which makes up the slices of a pie chart

Axes

Each pie displays the distribution of values of one categorical variable. The size of each pie slice is proportional to an aggregated count of occurrences of a specific value (i.e. category) in that variable.

Colour

Colour is used to represent the different categories existing for a variable. Therefore, the colours used in different tabs are independent of one another and should not be linked to one another in any way. This means that if a slice (e.g. respiratory) in the “Diagnosis” tab, for example, is red and another slice (e.g. female) in another tab (e.g. gender) is also red, this does not mean that they are connected or interacting. We rely on the fact that these pie charts cannot be shown at the same time. Users are made aware that these tabs are independent and their colour palettes are repeated only because the number of colours that can be supported does not scale to the entire set of possible categories for up to 5 different variables. Therefore using unique colours across tabs is not feasible.

Within each tab, the colour palette for values of the pie is obtained from concatenating two palettes of 12 colours each from the 12-classes qualitative palettes on ColorBrewer (Figure 13). Colours from the first palette are exhausted first before colours from the second palette are used. It is less likely that a categorical variable in an audit will have more than 12 different categories. Therefore, in most cases only the first palette is used.

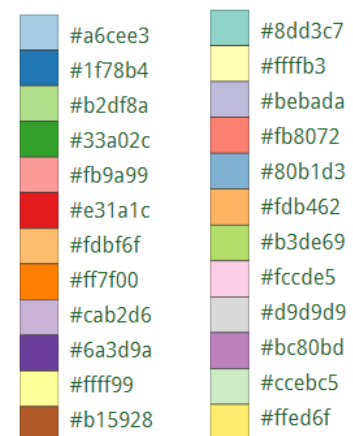


Figure 13 palettes 1 (left) and 2 (right) for categories in pie view

To see both palettes on ColorBrewer, please refer to:

<http://colorbrewer2.org/#type=qualitative&scheme=Paired&n=12>.

Labels

Pie slice labels are displayed in two different ways:

- 1- Hovering the mouse on a pie slice will display its label as a tooltip.
- 2- Textual labels are displayed by default on the periphery of each Pie sub-view as arrows pointed outward from the corresponding slice (see Figure 10 for example). To avoid clutter, these peripheral labels are only displayed for categories containing 10% or more of the total number of records displayed in the pie. An exception to this 10% rule occurs when selection is in place as will be described in the next sub-section (1.5.1.3 Interaction Design).

Interaction Design

Interaction in this sub-view is mainly aimed to link the variables shown in the main view to the Pie sub-view. This is to enable users to understand sub-groupings and categories within sets of records shown in monthly aggregates (in the main view).

Two forms of interaction are supported:

Mouse hover:

Hovering the mouse on a bar in the main view highlights it and animates slices of the pie. This animation is used to reflect changes in value distribution from the dataset that is originally shown to an updated subset. Specifically, mouse hovers in the main view filter the data records shown in the Pie sub-view to those currently highlighted in the main view only.

Figure 8 and 9 in this documentation show an example where the Pie sub-view is updated to show the case mix (i.e. Diagnosis) of five patients who died in April 2014. This selection is performed by hovering the mouse on the bar that represented mortality (Deaths in unit) in April in the main view.

Mouse hovers also work in the opposite direction. Namely, hovering the mouse on a pie slice shows the distribution of value counts of the corresponding category over time by overlaying highlights on top of the bar charts as shown in Figure 12.

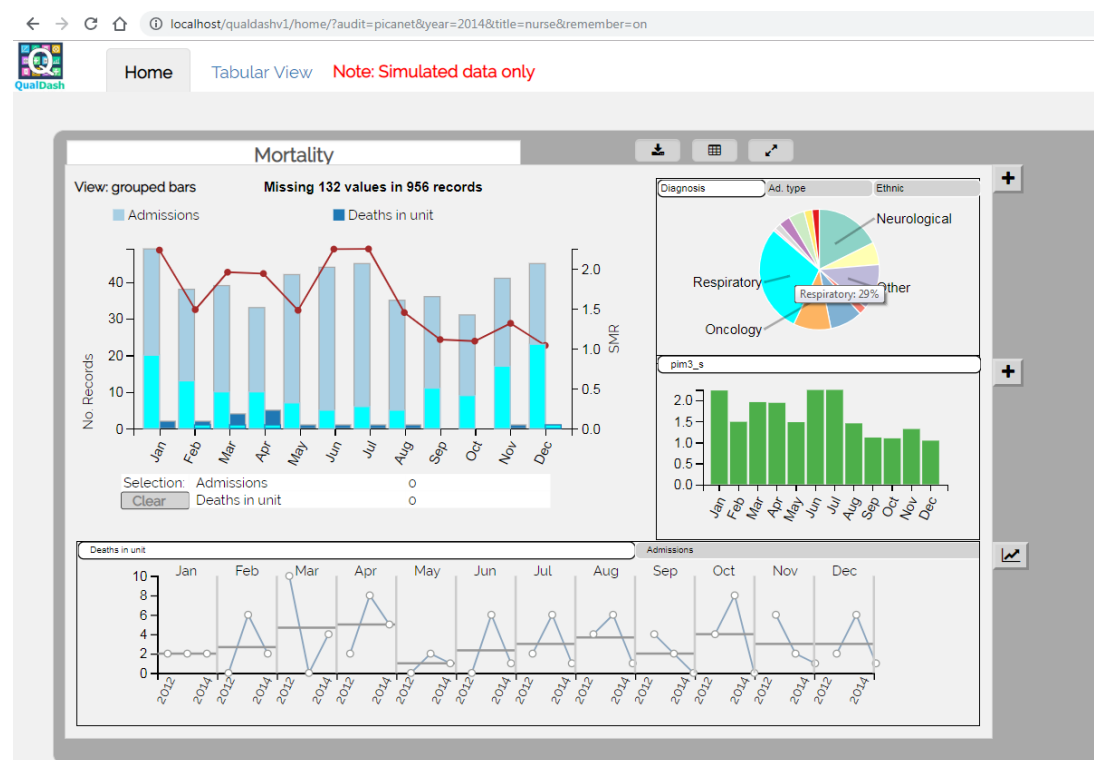


Figure 14 mouse hovers in Pie sub-view

Mouse selection:

Unlike mouse hovers which are bidirectional between the main view and the Pie sub-view, mouse selection only works in one direction. Specifically, users can click a bar in the bar chart view to select it. However the same is not true for pie slices.

Selecting a bar in the main view keeps the updated view of the pie chart after the mouse is moved away from the bar (unlike mouse hover). The main view enables users to select multiple bars together by clicking them one after another. Adding each bar will gradually update the pie chart to represent counts in the combined set of the different bars. A set union is performed across the different bars to obtain this updated distribution.

Note that if two bars contain overlapping sets of records, the repeated records are only considered once in the set union so counts shown on the pie chart will correctly display the distribution of categories while disregarding any repeated records (See for example Figure 13).

Note also that the labels of pie slices are always displayed if the corresponding category exists in a selection, regardless of the proportion of records in which the category exists (the 10% rule does not apply to selections).



Figure 15 Set of all patient admissions in April are selected (top) and set of deceased patients in April are added to the selection (bottom). In both cases, the pie chart displays the set union of the two bars which results in the same counts being displayed regardless of the fact that deceased records are a subset of admission records.

4.2.5.2 The Bar sub-view

The Bar sub-view is displayed in the middle of the right side of the expanded QualCard (see Figure 8). The tabs in the sub-view display a number of aggregated quantities from the database that have been specified in the MSS of the corresponding card. This default selection of variables can be further customized by the user through the Graphical User Interface (GUI) of the dashboard. In this section, we describe this GUI-based customization and explain the visualizations and interactions supported by QualDash in this sub-view.

GUI Customization

Clicking on a tab displays the variable's corresponding data column as a bar chart visualization in the Bar sub-view. A (+) button exists at the top right corner of the sub-view (See Figure 15). When clicked, this button displays a popover element that can be used to customize the variables displayed in the tabs. The popover element shows two multi-select lists of variables.

The list on the left contains all variables in the database that have a quantitative data type (data types are specified in a metadata file that is stored in the /data folder. See Section 1 of Part IV of this documentation for details on data storage).

The list on the right hand side of the popover contains variables that are currently displayed in the Bar sub-view tabs. There are two required steps to update tabs in the sub-view:

Step 1: add / remove variables from selection:

To add new variables:

- Select one or more variable(s) from the left hand side list and click the right arrow button to move your selection into the selected variables list.

To remove variables:

- Select one or more variable(s) from the right hand side list and click the left arrow button to move your selection out of the selected variables list.

Step 2: confirm selection

Once happy with the list of variables to be displayed in the Bar sub-view, click on the "Update" button to confirm your selection.

Note that your changes will not be saved if you don't click the "Update" button. Clicking away from the popover will close the popover and will not save your changes. Therefore, it is crucial to press "Update" before navigating away from the popover.

Note also that your changes will not take effect if you select too many or too few variables to display on tabs. The Bar sub-view currently supports between 1 and 5 variables at a time. At the time when you press "Update", if the list of selected variables (right hand side) has less than 1 or more than 5 variables, QualDash will issue an error message and ask you to add the appropriate number of variables (See Figure 11).

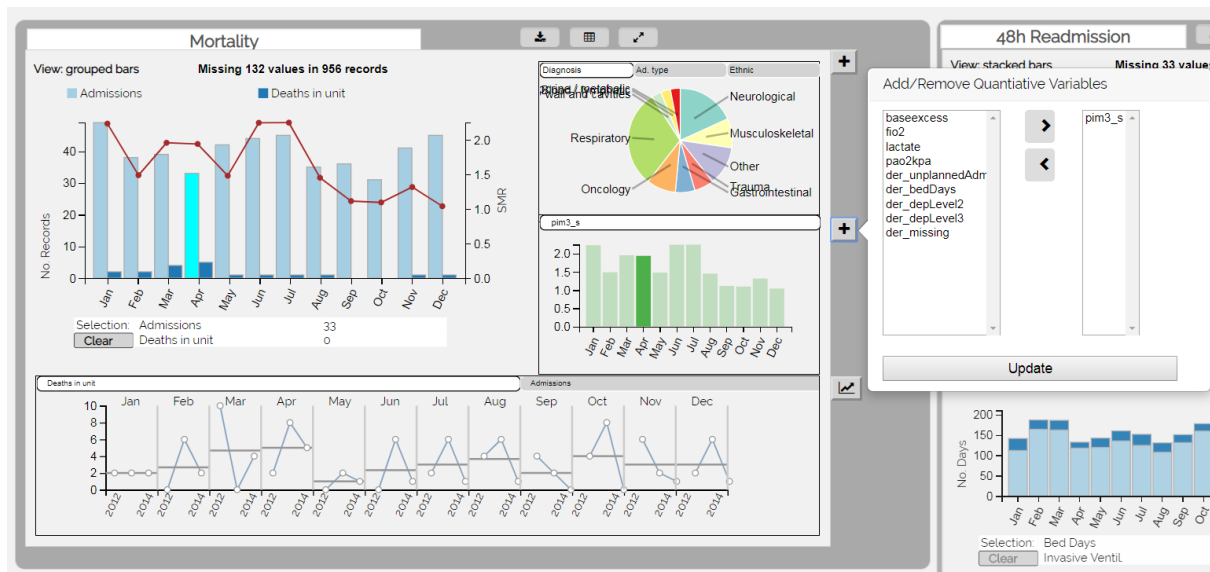


Figure 16 Customizing tabs of Bar sub-view

Visualization Design

Mark type

The mark type supported in the Bar sub-view is “rect” (i.e. bar).

Axes

Similar to the bar chart in the main view, the bar chart of the Bar sub-view displays time on the x-axis. The current aggregation level is also similar to the main view (monthly).

The y-axis displays aggregates which default to a sum in this view, unless otherwise specified in the MSS. Unlike the main view, however, the Bar sub-view only displays aggregates for one variable in each bar chart. Viewing a different variable in this sub-view requires the user to add this variable in a new tab in the sub-view.

Colour

Bars in each tab (i.e. for each variable) are displayed in a colour that is distinct from all other tabs in the Bar sub-view. The colours used for this sub-view are also distinct from ones which have already been used for bars in the main view. This is to avoid misleading the user. Specifically, no two bars are displayed in the same colour in the same QualCard unless they represent the same variable. To maintain this distinction, the QualDash engine selects the next available colour from the main palette shown in Figure 8.

Interaction Design

Interaction in this sub-view is mainly aimed to link the variables shown in the main view to the Bar sub-view, typically for comparison purposes.

Two forms of interaction are supported:

Mouse hover:

Mouse hover in the main view fades out all bars in the Bar sub-view and only emphasizes the bar that is currently hovered upon in the main view. For example, if the user hovers over a bar corresponding to the month of April in the main view, only April is emphasized in the Bar sub-view

and all other bars in this sub-view are faded out. Unlike the pie chart, however, this sub-view currently only supports this linked behaviour if the interaction takes place in the main view. Hovering the mouse on a bar in the Bar sub-view will only highlight this bar in the sub-view but will not affect the main view in any way. We intend to add this functionality in future versions.

Mouse selection:

Mouse selection only works in the main view. A selected bar in the main view will emphasize the corresponding bar in the Bar sub-view.

4.2.5.3 The History sub-view

While the main view of the QualCard and the two sub-views described thus far all display different facets of the same dataset (one for the year selected on the welcome page), the need to put this data into its historic context dictates the use of a history sub-view. This sub-view presents the variables shown in the main view and compares their monthly values with the corresponding time period in up to two previous years.

In theory, more years can be added to the comparison. However, we note that loading data for several years requires massive space and computations which may present scalability challenges to the web browser. In order to maintain good performance of the dashboard and avoid delays, this view currently presents three years for comparison: the current year and up to two previous years.

The tabs on top of this sub-view are not customizable, unlike the two previous sub-views. This is also a compromise that QualDash makes to maintain adequate performance, since displaying historic data for an uncontrolled number of variables requires computations to be performed on a large number of records, which may hinder interactivity. We therefore keep the tabs of this view to represent only the variables that are already displayed in the main view.

Visualization Design

Two visualization techniques are supported in this view, both make use of line and circle marks.

Time Series View

This is the default view once a card is expanded. It is a traditional time view where each year is represented as a line. Different colours and dashed patterns are used to draw each year's data. Circles are used to show exact values at specific time instances (See Figure 5). The point colour matches the colour of the corresponding bars in the main view for the variable shown in the history sub-view. The solid line represents data for the year which is currently loaded in the dashboard. All historic data (i.e. previous years) are shown as dashed lines with varying patterns.

Small Multiples View

It aims to cluster together data corresponding to a specific time unit. By default, data is clustered by month. Within each month, a line is used to represent the variability of the selected variable across different years (See Figure 16).

Interaction Design

Interaction in this view is limited to mouse hovers in the time series visualization. Hovering the mouse over a circle will emphasize it and display its value. Hovering the mouse on the line itself will emphasize the line and display its corresponding year.

4.3 TABULAR VIEW

In addition to the main “Home” tab which displays the main dashboard, a “Tabular View” tab enables users to inspect subsets of the audit data in a table format to obtain more information. This view is initially empty and is only populated once the user exports a record selection from one or more QualCards, using the “Export to table” button in the control panel of the card.

An example is shown in Figure 17, in which the user selects admission and mortality records for the months of March and April. The Tabular View understands that two different variables are selected and creates a corresponding number of sub-tabs. Each sub-tab displays the name of a variable and lists the table containing the set of records represented in the selected bars for that variable. Adding more variables in the selection will append more tabs in this view. Adding more records within the same variable will append more records to this variable’s corresponding tab in the Tabular View.

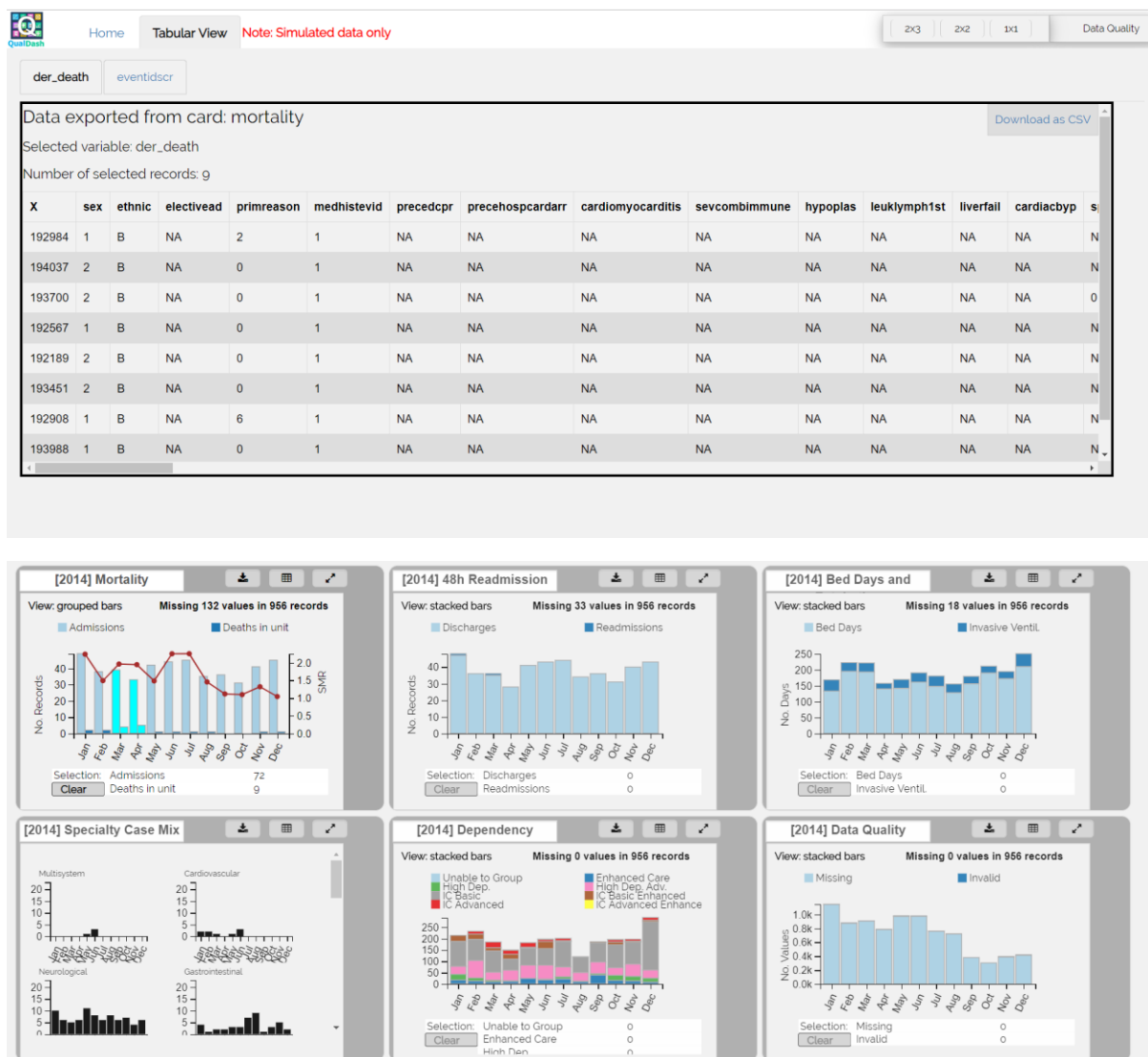


Figure 17 Exporting a selection to Tabular View

4.4 DATA EXPORTS

There are two main ways to export data from QualDash: Exporting data and exporting visualizations.

4.4.1 Exporting Raw Data

Once a data selection is exported from a visualization QualCard to the Tabular View (See previous section), the user can press the “Download as CSV” button on the top right corner of any table in the Tabular View (See Figure 17) to save the data in this table to a local file in Comma Separated Value (.csv) format. The downloaded file can then be opened in Excel for further processing.

4.4.2 Exporting Visualizations

The “Download” button on the control panel of each QualCard enables a user to download the visualization(s) shown in the card to local .PNG files. If the QualCard is in its default view when this button is pressed, then only the main view visualization gets downloaded. If the card was expanded at the point when the button was clicked, the browser will prompt the user to allow multiple downloads and if the user allows this process, then QualDash will download all visualizations in the view, including the sub-views.

5 BACKEND AND API

5.1 PREPARING DATA AND METADATA

For each audit, QualDash loads a metadata file that describes the data types included in the audit, and a set of audit data files. This section describes the backend process of preparing audit files to be loaded into QualDash and the formats that QualDash expects as input for each audit.

5.1.1 Metadata Files

For Picanet, QualDash loads the file “./data/picanet_meta.csv” and for MINAP it loads the file “./data/minap_meta.csv”. These files are essential for the dashboard to read the correct data types. If the metadata file is missing, an error message is issued and the dashboard stops loading.

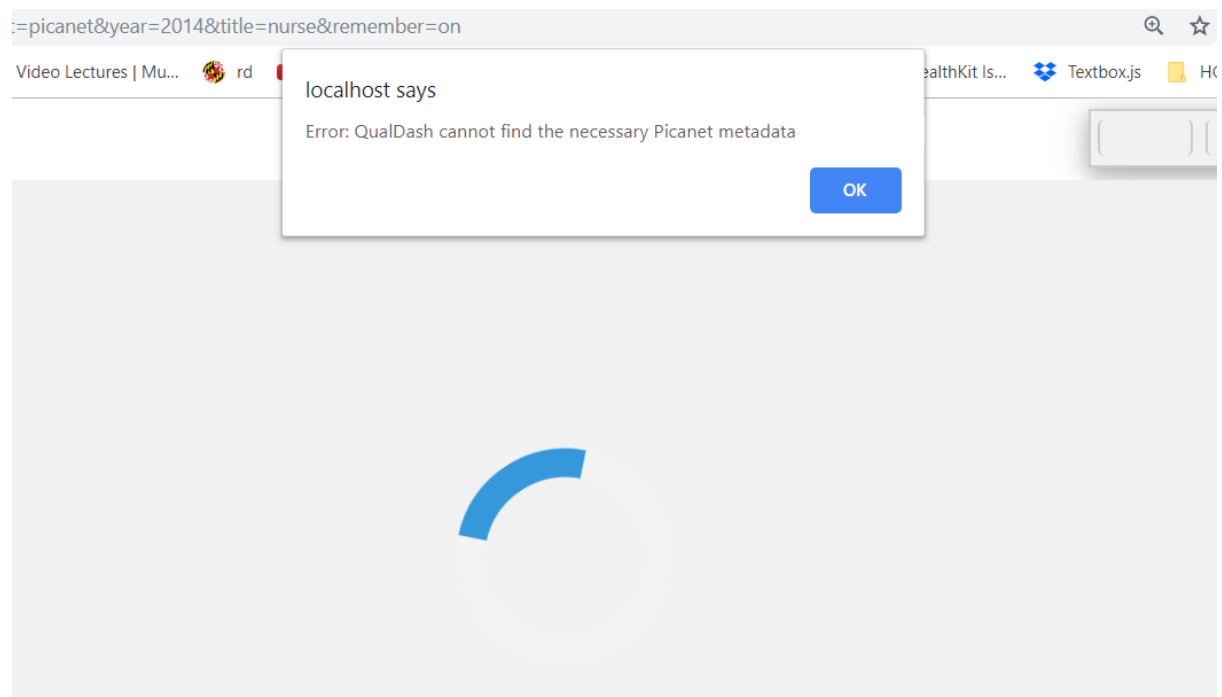


Figure 18 Error message due to missing metadata file

5.1.2 Picanet Audit Files

The Picanet audit files are uploaded to two folders: “./data/picanet_admission” and “./data/picanet_activity”. The admission folder contains a set of comma separated value files (.csv) that are named by year (for example “2012.csv”, “2013.csv”, etc.).

To prepare audit data in this format, we provide an R script that accepts raw admission and activity files as input and generates and stores the files in their proper locations. The R script for Picanet data is called “picanet_rscript.r” and can be found under the ./data folder. It can also be accessed online on Github at

https://github.com/maya70/Qualdashv1/blob/master/home/data/picanet_rscript.r

Please note that you would need to modify the script to the location of your raw audit files. For example, the line:

```
1 # read raw admission data from its current location
2 admission <- read_csv("C:/Users/Mai/Dropbox/Leeds/Qualdash_related/Data/admission.csv")
3
```

Must be modified to the file path and file name of your raw admissions file. The same applies for loading the activity file:

```
12 # read raw activity data
13 activity <- read_csv("C:/Users/Mai/Dropbox/Leeds/Qualdash_related/Data/activity.csv")
14
```

Once given the correct file locations, the R script will split the data into the separate files for each year and store them in the location that QualDash expects.

Please note that you may also need to modify the destination file paths if your server’s document root is different from the default set by Bitnami WAMP stack on Windows ('C:/Bitnami/wampstack-7.0.12-0/apache2/htdocs/'). If your path is different, please update the paths in lines 8, 21, 29, 33 and 43 in the R script to reflect your correct file path.

To check that your files are generated successfully, check that the folder “./data/picanet_admission” contains files for all the years included in your site’s audit database. For example, if your original audit file covers the period 2016 – 2018, you should see the files “2016.csv”, “2017.csv” and “2018.csv” in the admission folder. In the activity folder, you should also see a number of generated files names “shortactivYEAR.csv” where YEAR is a four digit number representing all the years that exist in the audit database.

5.1.3 MINAP Audit Files

The MINAP audit files are uploaded to the folder: “./data/minap_admission”. This folder contains a set of comma separated value files (.csv) that are named by year (for example “2012.csv”, “2013.csv”, etc.).

To prepare audit data in this format, we provide a MINAP-specific R script that accepts a raw admissions file as input and generates and stores the files in their proper location. The R script for MINAP data is called “minap_rscript.r” and can be found under the ./data folder. It can also be accessed online on Github at

https://github.com/maya70/Qualdashv1/blob/master/home/data/minap_rscript.r

Please note that you would need to modify the script to the location of your raw audit files. For example, the line:

```
1 # read raw admission data from its current location
2 madmission <- read_csv("C:/Users/Mai/Dropbox/Leeds/Qualdash related/Data/minap_dummy.csv")
3
```

Must be modified to the file path and file name of your raw admissions file.

Once given the correct file location, the R script will split the data into the separate files for each year and store them in the location that QualDash expects.

Please note that you may also need to modify the destination file paths if your server's document root is different from the default set by Bitnami WAMP stack on Windows ('C:/Bitnami/wampstack-7.0.12-0/apache2/htdocs/'). If your path is different, please update the paths in line 13 of the R script to reflect your correct file path.

To check that your files are generated successfully, check that the folder `./data/minap_admission` contains files for all the years included in your site's audit database. For example, if your original audit file covers the period 2016 – 2018, you should see the files `"2016.csv"`, `"2017.csv"` and `"2018.csv"` in the `minap_admission` folder.

5.2 LOADING DATA INTO QUALDASH

When loading the QualDash dashboard into a web browser, the start page displays a list of audits to choose from and a set of years for data available in the site for each audit. The set of years is dynamically populated based on the files available in the `./data` folder for the corresponding audit. For example, if the `./data/picanet_admission/` folder contains data for the years 2010 – 2014 then selecting this audit will cause the year selection to display 2010, 2011, 2012, 2013 and 2014 as options to choose from. Selecting a year displays its data in the main view of the QualCards and loads two previous years (if available) in the historic view of each QualCard. The decision of what variables get loaded in the main view and the sub-views of each individual QualCard is configured in the Metric Specification Structure (MSS).

5.2.1 Metric Specification Structure (MSS)

Metric Specification Structures are JSON (JavaScript Object Notation) objects that are kept in audit- and site-specific configuration files. These files are accessible under `"/js/config_mss_minap.js"` and

“/js/config_mss_picanet.js” on the server at each site.

```

41 "displayVariables": [{ "metric": "Mortality",
42                       "mark": "bar",
43                       "chart": "grouped",
44                       "colorScale": "categorical",
45                       "x": "addate",
46                       "y": ["eventidscr", "unitdisstatus", "der_smr"],
47                       "legend": ["Admissions", "Deaths in unit"],
48                       "yaggregates": ["count", "count", "runningAvg"],
49                       "yfilters": {"eventidscr": {"where": "*"},
50                                   "unitdisstatus": {"where": "unitdisstatus", "sign": "=", "value": "2", "valid": ["1", "9"]
51                                   }},
52                       "xType": "t",
53                       "yType": ["q", "q", "q"],
54                       "xspan": "year",
55                       "yspan": "unit",
56                       "ylabel": "No. Records",
57                       "tspan": 3,
58                       "granP": ["unit", "unit"],
59                       "ehr": "Admissions",
60
61                       /** Slave Tasks spec begin here **/
62                       "categories": ["primarydiagnosisgroup", "adtype", "ethnic"],
63                       "quantities": [{"q": "pim3_s", "granT": "addate", "granP": "unit", "yaggregates": "sum"}], // from t
64                       "granT": {"monthly-annual": ["eventidscr", "unitdisstatus"]} // the first element holds the master
65
66                       }
67

```

Figure 19 Example Metric Specification Structure for the Mortality card in the Picanet audit.

The key “displayVariables” holds an array of MSS objects to specify the contents of this audit’s QualCards. The number of QualCards that are loaded into the dashboard is equal to the number of entries in this array. **Error! Reference source not found.** shows an example MSS for the Mortality QualCard of the Picanet audit.

5.3 SESSION LOGGING

QualDash uses PHP to log user sessions. Session logging aims to collect anonymous usage data for QualDash. For each session, QualDash logs the timestamps of when a session is started and ended to calculate duration of use. It also logs the data entered by the user on the Getting Started page (audit, year, and job title). Additionally, we collect names of cards that were expanded in a session. Events are listed in the log file as shown in Figure 20.

We also log button clicks involving customization buttons for the Pie sub-view and the Bar sub-view, and buttons in the QualCard’s control panel.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	sessionID	evtOwned	evtType	evtTime	paramName	paramValue	paramValue	paramValue	paramValue				
2	2	application	start	7/17/2019	year				2018				
3	2	application	start	7/17/2019	audit				minap				
4	2	application	start	7/17/2019	jobTitle				analyst				
5	2	card	expand	7/17/2019	metric				Mortality				
6	2	layoutCtrl	2x3	7/17/2019, 3:58:56 PM									
7	2	bar	select	7/17/2019	metric				Mortality				
8	2	bar	select	7/17/2019	variable				NHS number				
9	2	bar	select	7/17/2019	metric				Mortality				
10	2	bar	select	7/17/2019	variable				NHS number				
11	2	bar	select	7/17/2019	metric				Mortality				
12	2	bar	select	7/17/2019	variable				NHS number				
13	2	card	tableExpor	7/17/2019	metric				Mortality				
14	2	layoutCtrl	2x3	7/17/2019, 4:01:07 PM									
15	2	bar	clear	7/17/2019	metric				Mortality				
16	2	card	expand	7/17/2019	metric				Mortality				
17	2	card	save	7/17/2019	filename				Mortalitycategory.png				
18	2	card	save	7/17/2019	filename				Mortalityquantity.png				
19	2	card	save	7/17/2019	filename				MortalitytimeView.png				
20	2	card	expand	7/17/2019	metric				Mortality				

Figure 20 sample log file

