TIMi suite
integrated data mining solutions

kibella
unlimited self service BI

Kibella Quick User's guide *v1.03*

Creation date: July 2016
Last Edited: August 2016

1. Introduction

Welcome to the Quick user's guide to Kibella. Kibella is part of the "TIMi Suite" that is an integrated set of tools designed to solve Business Intelligence problems. The acronym "TIMi" stands for "The Intelligent Mining Machine (TIMi)".

Kibella is a "dashboarding tool" or "Dataviz tool" similar to Tableau, Qlickview, Kibana, Kyubit, etc. Basically, a "Dataviz tool" allows you to see different kind of visualizations (bart chart, pie chart, etc.) inside a web browser.

More precisely, Kibella allows you to create and deploy everywhere interactive dashboards (with full "drill-down" functionality) in a few mouse-clicks. Kibella is closely integrated with Anatella so that all data available inside Anatella can easily be exported to Kibella to create beautiful visualizations and dashboards.

Kibella is related to Kibana (from which it shares a part of the source code). …but Kibella adds many interesting features "on-top" of Kibana:

- **No complex infrastructure required.**
  Kibella works inside a simple LAMP/WAMP stack. In opposition, Kibana can only be deployed inside a complex hadoop-based infrastructure that typically implies many nodes (for distributed computations) and specific, non-standard, webservers (i.e. node.js).

- **Simpler to administrate**
  Kibella is built using a set of technology that is mastered by 100% of all technicians (i.e. Apache+Php+SQLite). In opposition, Kibana requires you to administrate "shards" (i.e. nodes with a specific elasticsearch stack installed on them) and to administrate "node.js".

- **Fast at Query-Time (the SQLite engine used by Kibella is based on efficient C++ code).**
  A very small server can handle a very large quantity of data. The SQLite engine used inside Kibella is using large in-memory data-cache, so that it can use all your RAM, if available. …but the SQLite engine from Kibella is not limited by the RAM of your PC (in opposition to Qlickview or Kibana): You can handle larger-than-RAM databases very fast & very easily.

  > *Note: We are still working on improving Kibella's speed. The final version of Kibella should display dashboards in a time divided by 5 compared to the Early-Access version of Kibella.*

  In opposition, the computing power delivered by Kibana on one server is much lower. Of course, you can install Kibana on many "shards/nodes" to still reach a large computing power.

- **Scalable**
  It's very unlikely that you ever need to take care about "scalability issues" when working with Kibella (e.g. because, in opposition to other solutions, you are not limited by the RAM of your server). However, if you are experiencing some "slow down" when "drilling down" inside a dashboard, you can still use Anatella to reduce the size of your datasets before exporting them to Kibella. To reduce your dataset size, you can use the many standard data transformations

available inside Anatella (i.e. mainly the "aggregation" box). With Anatella, you can easily handles databases containing several dozens of Terabytes on a "common grade" laptop.

The combination of Kibella and Anatella allows to cope with any situation, whatever the size, the complexity, the origin, the velocity or the format of the data.

Kibana typically scales in a different way: You'll typically add additional computing nodes (named "shards" in the elastic world) to "scale up". In comparison to Kibella, Kibana has a more expansive "scaling up" (because all these additional nodes involves additional costs in hardware and additional costs in salary for maintenance & support of this exotic components).

- **Fast at load-time.**
  Kibella has no load time: To update the data "behind" your dashboards, you just use Anatella to create/update your SQlite database files: This is usually done in a few seconds.
  In opposition, with Kibana, updating the data "behind" your dashboards is a long procedure because Kibana takes a huge amount of time to re-start (after the JSON dataset files have been updated): Sometime several hours (because it has to parse the complex JSON files to re-create many complex indexes).

- **Integration with Anatella.**
  This is not really a difference because both Kibella and Kibana are 100% integrated with Anatella: To export you data to Kibella or Kibana (or Tableau or Qlickview), you'll use:
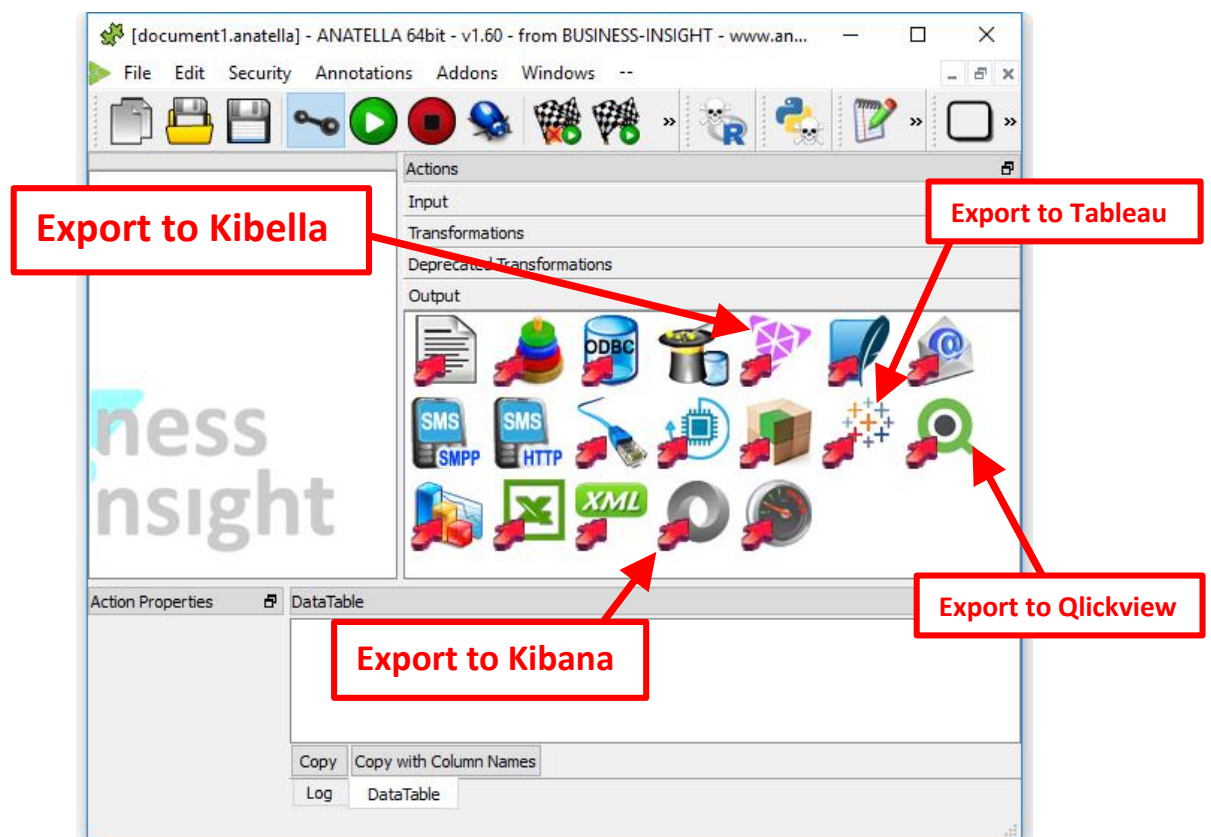
It should be noted that, up-to-now, there exists no other ETL tool that is able to create the complex JSON files used by Kibana: i.e. To use Kibana, you must provide all the data to query within a single, large & complex JSON file. This JSON file has a rather twisted structure (i.e. it has a multi-level structure). There is inside Anatella a small box (the "writeJSON" box) that is

exactly designed to generate these "multi-level" JSON files. You'll find more details about the integration between Anatella and Kibana in the section 4.8.17.13. of the "AnatellaQuickGuide.pdf".

An example: The data scientists from AXA Belgium (i.e. the guys from the AXA "Datalab") are using extensively the "writeJSON" box from Anatella to create various dashboards in Kibana (they have more than ten Kibana running all the time). In one of their dashboard, they generate a very complex 40 GB JSON file (4 levels deep) in less than 15 minutes with Anatella (in comparison, the other data scientists attempted to replicate the same performances in Hadoop and they had no results after 24 hours of computation). You'll find more details about the integration between Anatella and Kibana in the section 4.8.17.13. of the "AnatellaQuickGuide.pdf".

Why did we create Kibella? We loved Kibana but we recognized that not all companies are willing to invest time and efforts in an exotic technology such as the one used inside Kibana. Basically, most companies have a quite limited volume of data to visualize (we noticed that, after proceeding with the required aggregations with Anatella, 99% of the companies have a data volume that is less than 1GB for visualization). Thus, for 99% of the companies (i.e. for the companies that have "normal" dashboarding needs), using a complex distributed computing system for visualization (such as Kibana) does not make any sense: it's much easier & cheaper to use Kibella.

Question: Do I need distributed computations (such as the one included inside Kibana) to create my dashboards?

Response:
Let's take a first real-life example: We are currently working extensively with many telecoms all around the world. We provide them with advanced analytics (e.g. churn & cross-sell models) and "basic" reporting/dashboarding. We noticed that, for a small telecom with around 20 million subscribers, each dashboard in production is actually only using around 10 MB of data that is stored "on the visualization webserver" (after some data aggregations with Anatella. …but these aggregations are nevertheless always required, whatever the "dataviz tool" used, because of the large size of the "source" data). In this case, you don't need Kibana.

Another real-life example: The "DataLab" from AXA makes a non-traditional usage of Kibana: They use it as a CRM tool (and not really as a Dashboarding tool). It means that they inject inside Kibana (i.e. inside ElasticSearch engine) their whole CRM database (that contains a table with their "360° Customer view": This table has more than 1000 columns). To obtain an interactive, sub-second, drill-down on this large table, you need Kibana (more precisely, you need the ElasticSearch engine).

Are some examples of dashboards made with Kibella or Kibana:
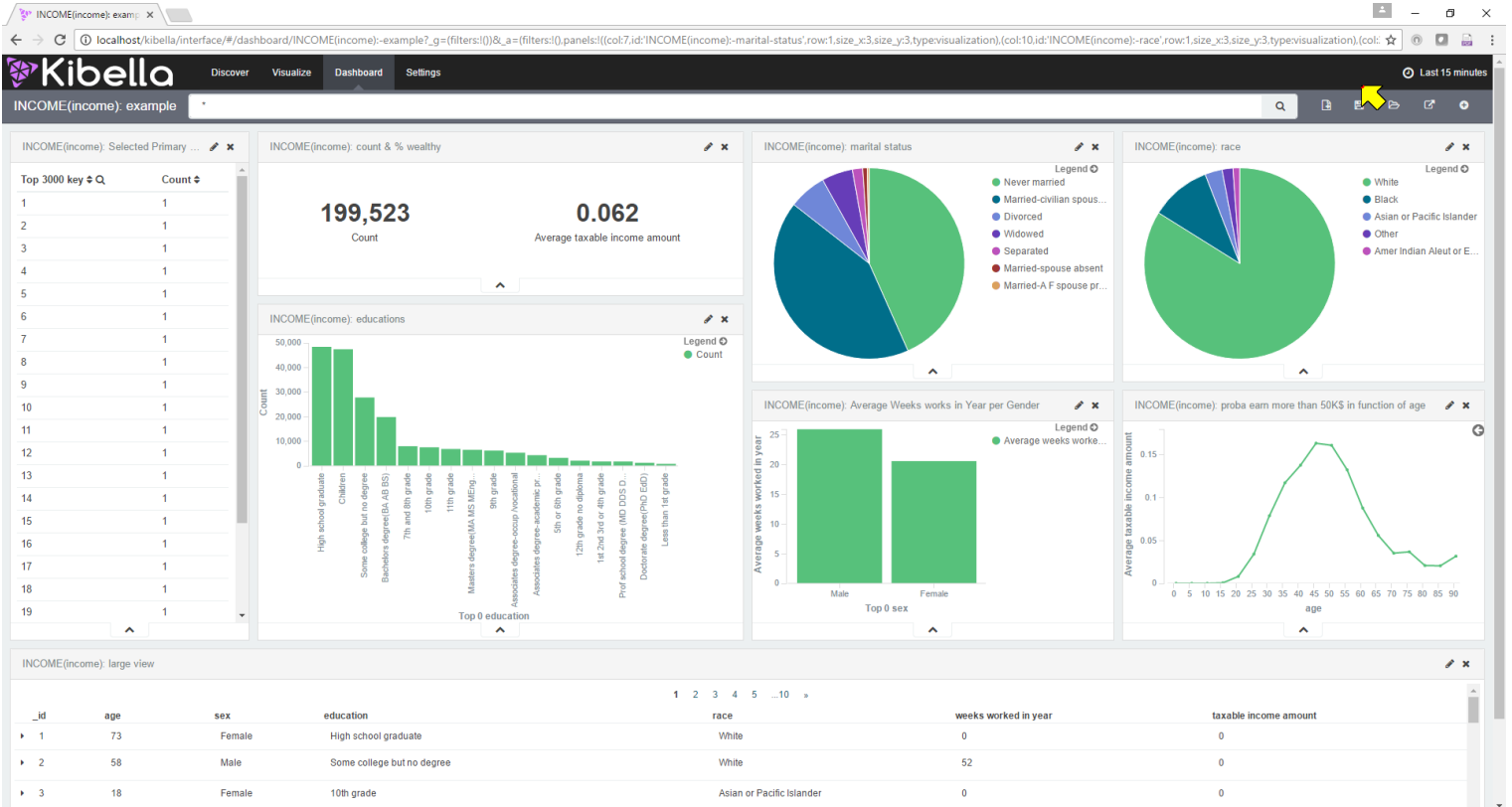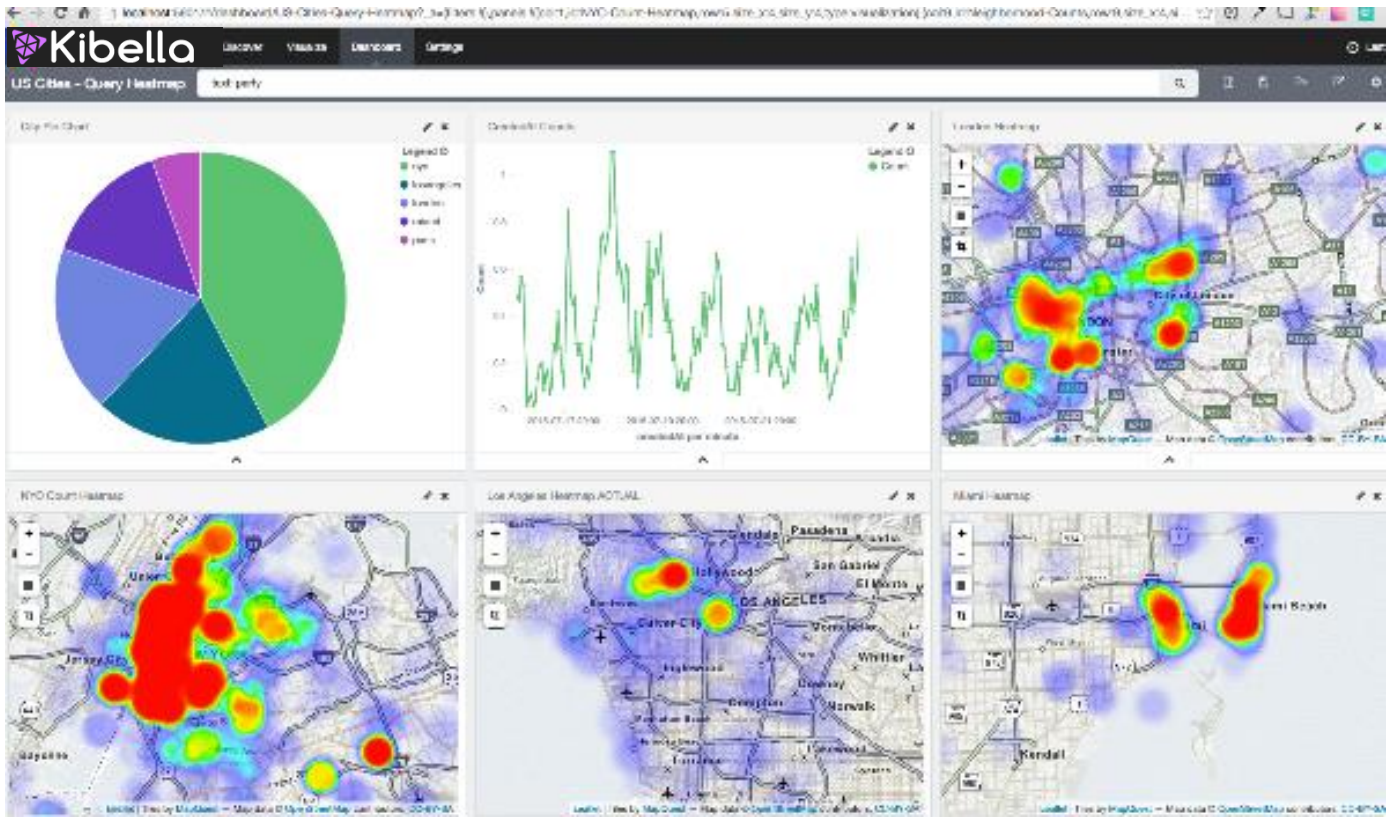
The main advantages of Kibella (over most other "Data Viz" tools) are:

- **Self-service BI**
  Anybody can create beautiful dashboards in a few mouse-click without writing any line of code.

- **Free & Unrestricted:**
  No limits on the numbers of viewers and designers.

> In particular, Tableau and Qlickview are limited in this regard because each additional person that simply "view" your dashboard costs you a thousand euro more. At this price, you'd better think twice before sharing anything with a new colleague!

- **In-Browser-Designer:**
  No need to install anything on the designer computer: The designer runs inside a browser. Anybody can become a "designer" and improve your dashboard design. This allows easy collaborative work.

In particular, Tableau and Qlickview are limited in this regard because each additional "designer" costs you a few thousand euro more. At this prices, you'd better think twice before collaborating on a dashboard with a new colleague!

- **Ethical solution:**
  The data format used by Kibella to store all the data for visualization is the simple and ubiquitous "SQLite" file format. The SQlite file format is readable & writable by 99% of the data-management tools currently available on the market. This means that you'll always be able to retrieve the data stored inside Kibella for further processing (e.g. for further processing inside a predictive modeling tool such as TIMi Modeler).

  In opposition to Kibella, Tableau and Qlickview are storing their data in a proprietary file format (TDE files for Tableau and QVX/QVB files for Qlikview) that no software (outside Tableau and Qlikview) can read (although Anatella can create these files, it cannot read them back).

  This means that, if you saw something interesting inside one of your dashboards and you want to investigate the matter in more details (e.g. using a predictive modeling tool such as TIMi Modeler), you simply cannot use the TDE or the QVB files to start your investigation.
  If you are lucky and you IT service kept a copy of the original data before exporting everything to Tableau or Qlickview, you can still work.
  …But, 99% of the time, this won't be the case. Most of the time, you won't be able to get access to your data anymore (only through a very limited aggregated view inside your browser, that is totally insufficient for any predictive modeling activities). This is a "lock-in" situation: i.e. You are now forced to use Tableau/Qlickview to make any data analysis (even if Tableau or Qlickview are not adapted to the nature of the investigation that you want to perform). This is why Tableau ang Qlikview are strictly forbidding anybody to create a data extraction tool from the proprietary format: They want a "lock-in" situation.

  Thanks to the open nature of Kibella (and within all the tools included in the TIMi Suite), you can have the 100% guarantee that we will never "take hostage" your data: You are safe with us. The only "lock-in" situation that we pursue is the one of Excellence: i.e. you are working with TIMi because that's technically the better solution.
  This is why we declare ourselves as an "Ethical Solution": No "lock-in" situation, just Excellence.

- **Cheap and Ubiquitous deployment:**
  Kibella runs on any common LAMP stack (Linux+Apache+MySQL+PHP) available in any datacenters. The LAMP stack is the most commonly used solution and the cheapest solution used to create Webservers.

  To still reduce costs, the same LAMP server can hosts several copies of Kibella (each copy is 100% isolated from the others). This allows to have one separated Kibella for each department inside your company at no additional cost (i.e. one kibella for accounting, one kibella for marketing, one kibella for engineering, etc.).

- **Fast at Query-Time (the SQLite engine used by Kibella is based on efficient C++ code).**
A very small server can handle a very large quantity of data. You can handle larger-than-RAM databases very fast & easily.

Note: We are still working on improving Kibella's speed. The final version of Kibella should display dashboards in a time divided by 5 compared to the Early-Access version of Kibella.

In any dashboarding solution, the bottleneck (that slows down everything) is always the webserver: i.e. The resources of the Webserver are limited and shared between all the "viewers" of the dashboards.

Thus, you need to have as much code running on the browser as possible (to have the smallest computing load on the webserver as possible). This is how Kibella is built and that's great.

…But Kibana is following another direction: The "ElasticSearch" company (that is developing Kibana) wants to promote their main product (that is "ElasticSearch"). To that end, they need to make sure that Kibana is as **in**compatible as possible with "classic & standard technologies" (such as an "Apache" server). This is why the "ElasticSearch" company restructured the code of the newest version of Kibana to have the largest part of the code as possible running on the webserver, inside "Node.js" (since the "Apache" webserver cannot run any Javascript code, this ensures that the newest releases of Kibana are **in**compatible with "Apache"). This is not a very good technical decision (although it makes sense commercially! 😉).

- **Fast at load-time.**
With all other solutions, updating the data "behind" your dashboards is a long procedure (involving painful & slow data extractions).

Especially, the "normal" Kibana takes a huge amount of time to start-up.

In opposition, updating the datasets displayed by Kibella is very fast & easy: Just use the new Anatella action named: "ExportToKibella": it's available here:
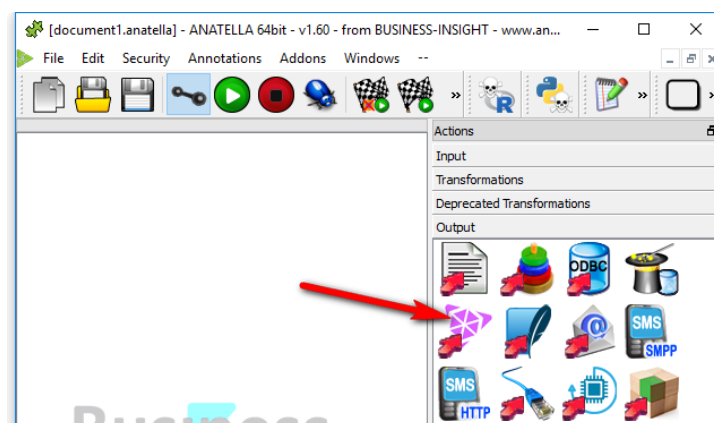

Usually, the speed of the "ExportToKibella" Action is only limited by the speed of your hard drive. This means that, 99% of the time, you can update the data "behind" your dashboards in a few seconds.

- **Incremental Loads:**
  The "ExportToKibella" Action supports a wide variety of "updates" mechanism. e.g. You can near-instantaneously add rows to (and/or delete rows from) an existing dataset.

- **Real-time display**
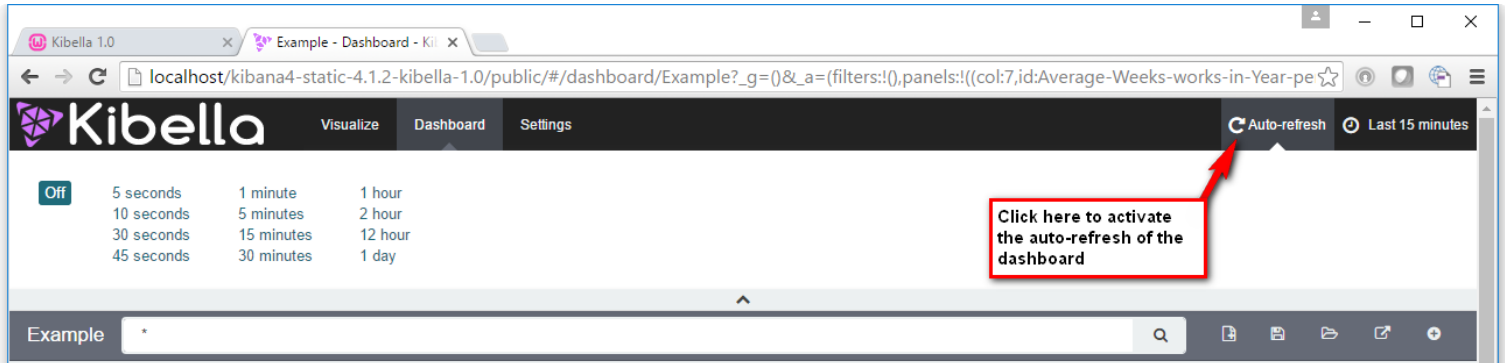
You can see inside your dashboard the latest data that just arrived (e.g. from the last 5 seconds). This functionality combined with the "Incremental Loads" allows you to obtain dashboards that update in "real-time", closely following the process that you are monitoring.

- **Easy deployment from lab to production:**
  When you finished developing some new dashboards locally (i.e. without any internet connection required), you can export your work to the production server by a simple drag&drop (i.e. just copy the "kibella.sqlite" file to the production server: This file contains all the visualizations and all the dashboards).

- **100% compatibility between the Windows & Linux executables:**
  You can design on Windows (locally) and publish on Linux everywhere.

- **Based on a 100% secure technology: i.e. the "Apache Web Server".**
  The "Apache Web Server" is the most common and the most trusted webserver currently available: Here are some market-share statistics:
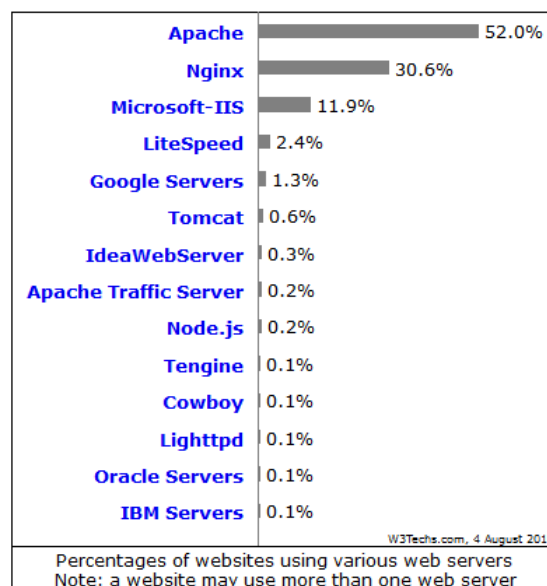
Thanks to Apache, Kibella is 100% secure: i.e. you can deploy all your dashboards to a wide audience without any fear of any "Russian Web Hackers".
All other dashboarding/BI solutions (Tableau, Qlickview, Kibana, Microsoft PowerBI,etc.) must be used exclusively on a private network, heavily secured (because these solutions are not robust against "Russian Web Hackers").

- **Large choice of Security options:**
  The "Apache Web Server" contains a large quantity of options to authenticate the Kibella dashboards users. You can manage users:
  - o "One-by-one" or at the "group level"
  - o Through standard LDAP protocol
  - o With many other options: More information here:
    http://httpd.apache.org/docs/current/howto/auth.html

- **Easily share your dashboards with anybody:**
  Just click the [icon] share icon and you receive an URL that you can send to anybody (in a mail, on skype, in whatsapp, etc.) to allow your correspondent to instantaneously see your results.
  The [icon] share icon is here:
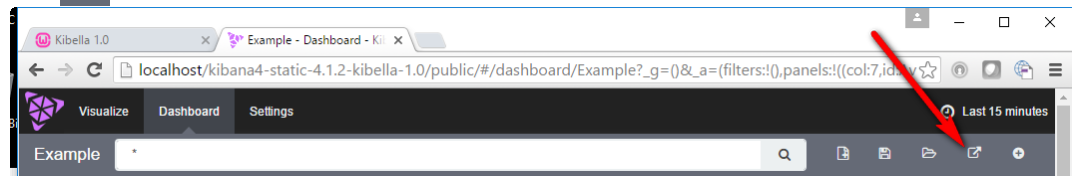
- Kibella is the only solution that allows you to distribute your dashboards to an unlimited audience (no financial limits, no barriers linked to the security of the server, very cheap infrastructure, instantaneous sharing).
- Somewhat Beautiful

The dis-advantages of Kibella are:
- On very, very large datasets, it's sometime better, to obtain a more reactive dashboard, to reduce the size of your dataset before exporting it to Kibella. To reduce your dataset size, you can use the many standard data transformations (i.e. mainly the "aggregation" box) available inside Anatella. Another solution is to use some distributed computations in Kibana.
- Not so beautiful compared to Tableau (but Kibella evolves rapidly! 😉)

We really think that the combination of Kibella with Anatella is just incredible. Anatella offer so much computing power and flexibility while Kibella allows you to share your results instantaneously to anybody, without any financial limits nor infrastructure limits.
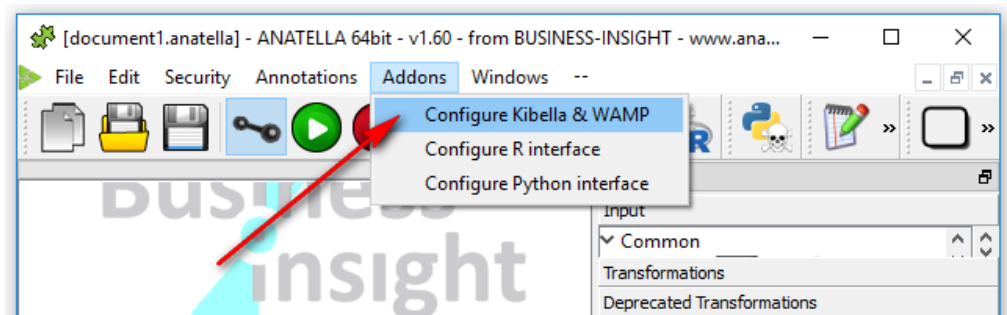
2. Installation

2.1. Windows Installation

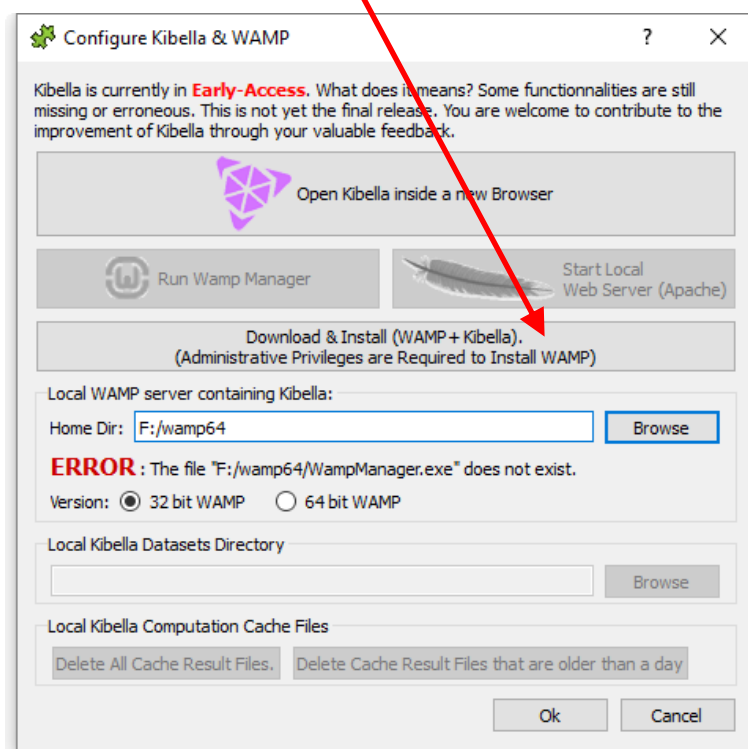### 2.1.1. Windows Automated Installation from Anatella

This is the preferred way of installing Kibella.

To install Kibella, open Anatella and click the "Configure Kibella & WAMP" inside the "Addons" menu:

Then, click on the "Download & Install" button:

Note that Kibella's installation directory must not contain any spaces or special characters (any character that requires a special treatment in an URL address is forbidden).

Anatella will then download & Install:
- the "Apache" webserver
- the Kibella files
- the "Wamp Manager" utility.

Once the above procedure is complete, click the button "Run Wamp Manager" here:
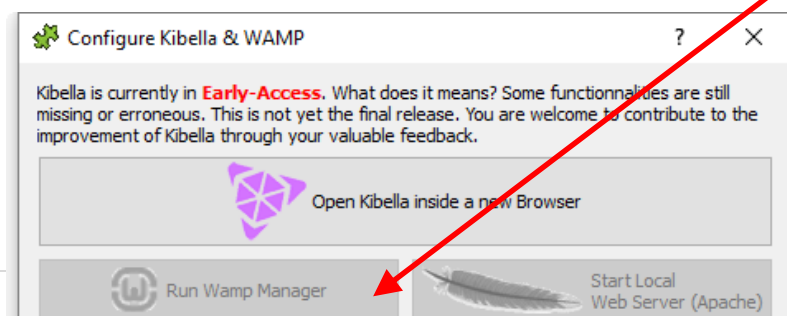
The "Wamp Manager" is a small utility that allows you to manage your "Apache" webserver. It installs itself inside the right-part of the MS-Windows Taskbar, next to the clock, on the bottom-right side of your screen. Once it's running you should see it here:

If the "Wamp Manager" icon is Red , it means that the Apache server could not start. Please refer to the section "2.3. Solving Installation Problems" to fix this.

If the "Wamp Manager" icon is Orange , everything is Ok: Your apache server is Up & Running: You can open the Kibella URL to start playing with Kibella: Go to: http://localhost/kibella
To open this URL, you can also simply click here:
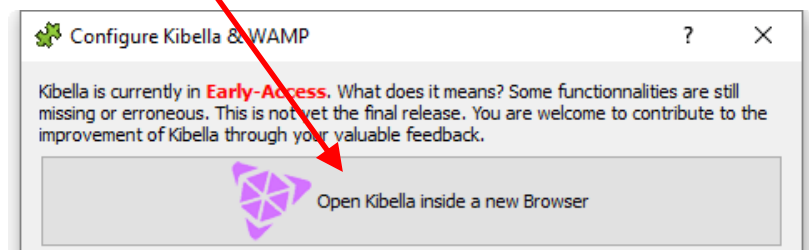
2.1.2. Windows Manual Installation

To manually install Kibella on a Windows machine:

1. Install a WAMP stack.
    Under windows, the default Apache+Php stack is the one here:
    http://www.wampserver.com/en/

2. Unzip the zip file downloaded from:
    http://www.anatella.com/downloads/kibella/kibella_alone_win.zip
    …inside your WAMP Installation directory.

    Note that Kibella's installation path must not contain any spaces or special characters (any character that requires a special treatment in an URL address is forbidden).

3. **IMPORTANT:** Edit the file "<WampInstallDir>/alias/kibella.conf"
    Only the first line of this file is really important: You must have:

```
AliasMatch "^/kibella/db(.*)$" "<WampInstallDir>/www/kibella/src/requests.php"

<Directory "<WampInstallDir>/www/kibella/">
    Options Indexes FollowSymLinks MultiViews
    AllowOverride all
        Order allow,deny
    Allow from all
</Directory>
```

    (where "<WampInstallDir>" must be edited to match your installation setup).

4. Restart your WAMP server so that it takes into account your new "Kibella.conf" file.

5. **Optional:** To enable the "Run Wamp Manager" button inside Anatella: Run:

   o For a 32-bit WAMP server:

```
schtasks /create /SC ONCE /ST 23:59 /TN "StartWampManager" /TR "cmd /c start \"\"
\""<WampInstallDir>/wampmanager.exe\"" /F /RL HIGHEST
```

   o For a 64-bit WAMP server:

```
schtasks /create /SC ONCE /ST 23:59 /TN "StartWampManager64" /TR "cmd /c start \"\"
\""<WampInstallDir>/wampmanager.exe\"" /F /RL HIGHEST
```

   (There are two .bat files inside the zip that might help you run the above command)

6. **Optional:** Configure the location of the databases: Edit the "<WampInstallDir>/www/kibella/kibella.ini" configuration file and change the datadir parameter to the location of the SQLite databases that are going to be queried. The default location (that is used if you don't do anything) is "<WampInstallDir>/www/data".

2.2. Linux Installation

To install Kibella on a Linux machine:

1. Install & start a LAMP stack
   (Inside this stack, you actually don't need MySQL because Kibella does not use it).

2. Unzip the zip file downloaded from:
   http://www.anatella.com/downloads/kibella/kibella_alone_lamp.zip
   …inside the directory that contains the webpages (e.g. "/var/www").

3. Verify (or edit as needed) that the Apache configuration file "/etc/apache2/sites-available/default" has the AllowOverride directive set to All inside the "Directory" directive block for the webserver directory (e.g. "/var/www").
   This means that that part of the file should (typically) look like this:

```
<Directory /var/www>
      Options Indexes FollowSymLinks MultiViews
      AllowOverride All
      Order allow,deny
      Allow from all
</Directory>
```

   This is needed to enable the .htaccess Apache configuration file specific to the Kibella application to work correctly.
   Restart your Apache server so that it takes into account the modifications done on its configuration file.

4. **Optional:** Configure the location of the databases: i.e. Edit the "kibella/kibella.ini" configuration file and change the datadir parameter to the location of the SQLite databases that are going to be queried.

5. Make sure that the directories "kibella/tempdata" and "kibella/tempdata/cache" have write (and read) permissions for the user that is running the webserver. This is usually achieved using:
   o your FTP client.
   o Some shell commands: e.g.

```
sudo chmod 777 –R /var/www/kibella/tempdata
```

   (Assuming that Kibella is installed in the webserver directory "/var/www/kibella", this should change the permissions of the "kibella/tempdata" directory and all its subdirectories and files to rw-rw-rw-).

2.3. Solving Installation Problems (Windows)

If the "Wamp Manager" icon is Red , it means that the Apache server could not start. To try again to start the Apache Server, left-click on the "Wamp Manager" icon (on the bottom-right side of your screen) and select the menus "Apache", "Service", "Start/Resume Service": Here is a screenshot of the whole procedure:
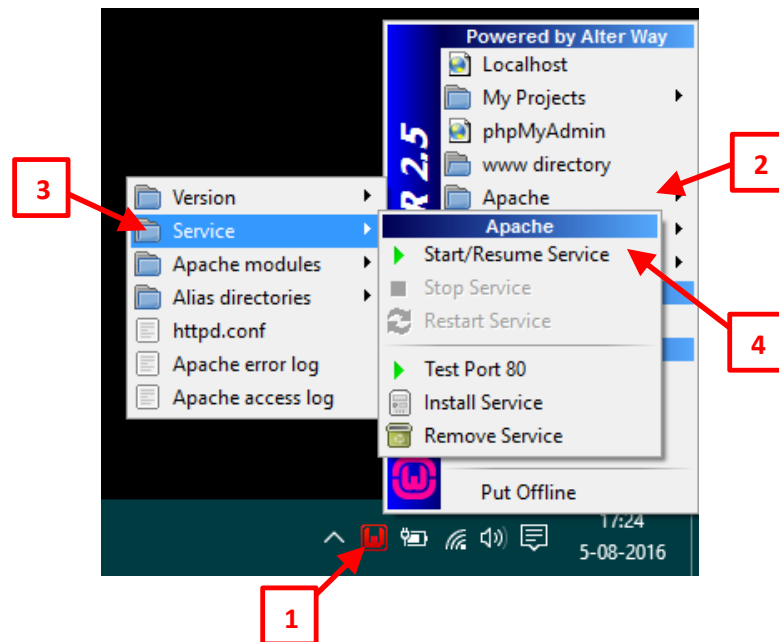
You'll get a message explaining why the "Apache" server could not start.

The most common cause of problem is Skype (that is blocking the TCP/IP port 80). To solve the Skype-related problem, open Skype and go to the "option window" (i.e. in the menu bar select "Tools" and then click on "Options") and then:
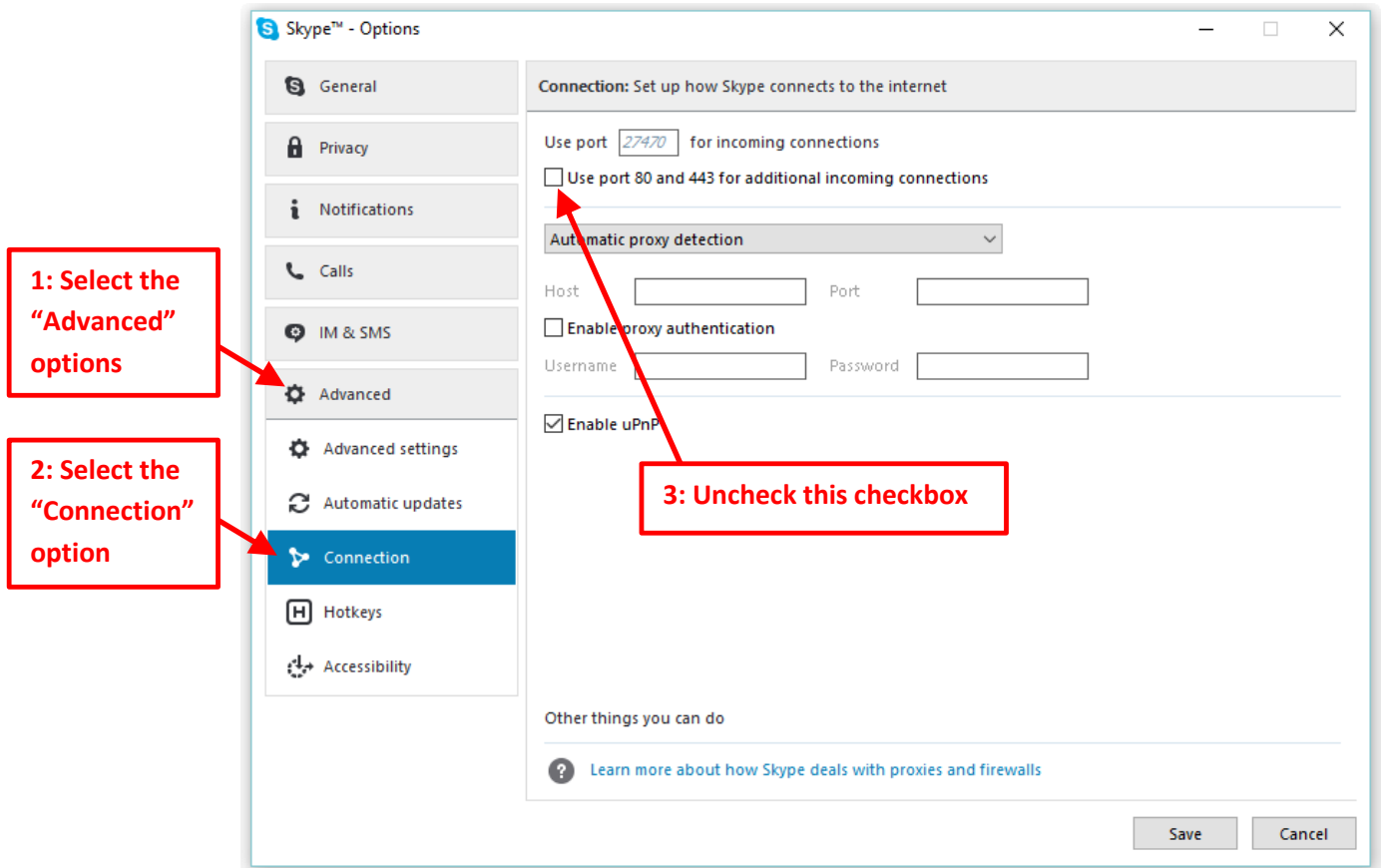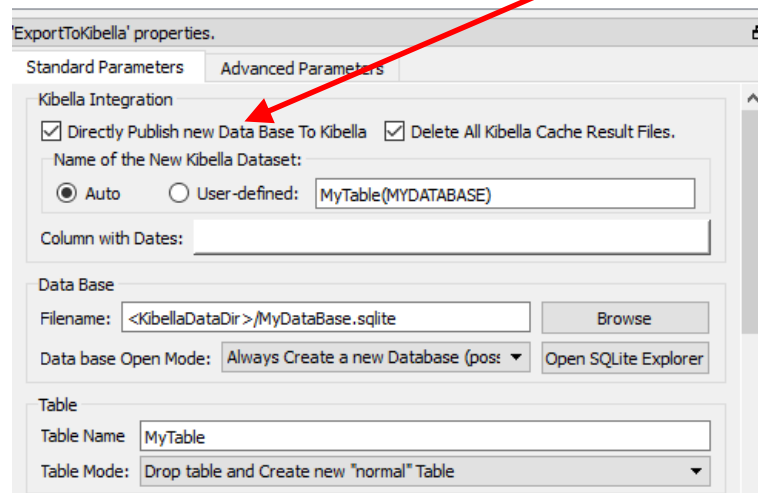

3. Configuration

The following table lists the parameters that can be defined in `kibella.ini`.

Parameter name	Default value	Group	Description	Comments
datadir	../data	User data directory	Directory given in absolute path to the location of the user databases.	Currently only SQLite databases are supported.
cachemode	day	Cache	Mode followed to delete old cache of query results. One of: - **day** - **rolling**	The cache modes are as follows: - **day**: cache files are deleted when they become more than one day old. - **rolling**: cache files are deleted when they have been modified earlier than '**cachehours**' before the same query is requested.
cachedaychange	04:00	Cache	Time at which a new day is considered to start.	Only used when cachemode is set to '**day**'.
cachehours	24	Cache	Number of hours after which a cache file is considered to be old and thus can be deleted.	Only used when cachemode is set to '**rolling**'.

## 4. Importing Tables inside Kibella

4.1. Importing Tables inside Kibella using Anatella

Just use the Action "ExportToKibella" and check the option "Directly Publish new Data Base to Kibella":
(This option is anyway checked by default)

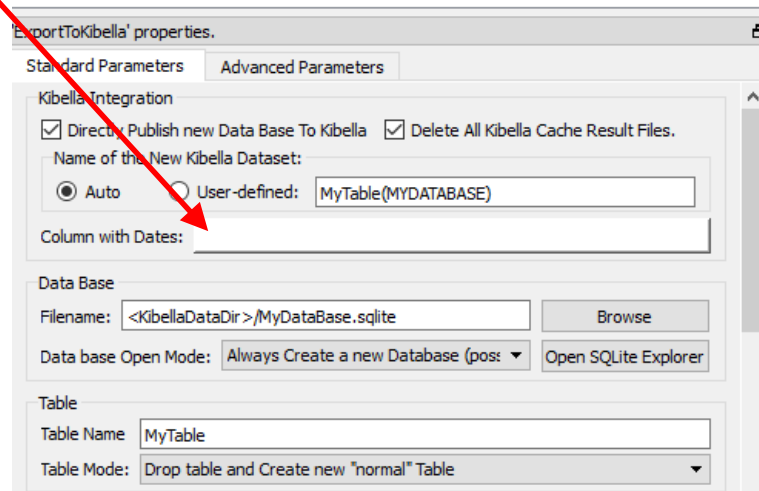### 4.2. Importing Tables inside Kibella using another ETL

Create a new SQLite database inside the "Kibella datadir" (the exact location of the "Kibella datadir" is defined inside the "Kibella.ini" file). Then, open a browser and go to the Kibella webpage (http://localhost/kibella/) and click on the "*Register a table to query on*" link:
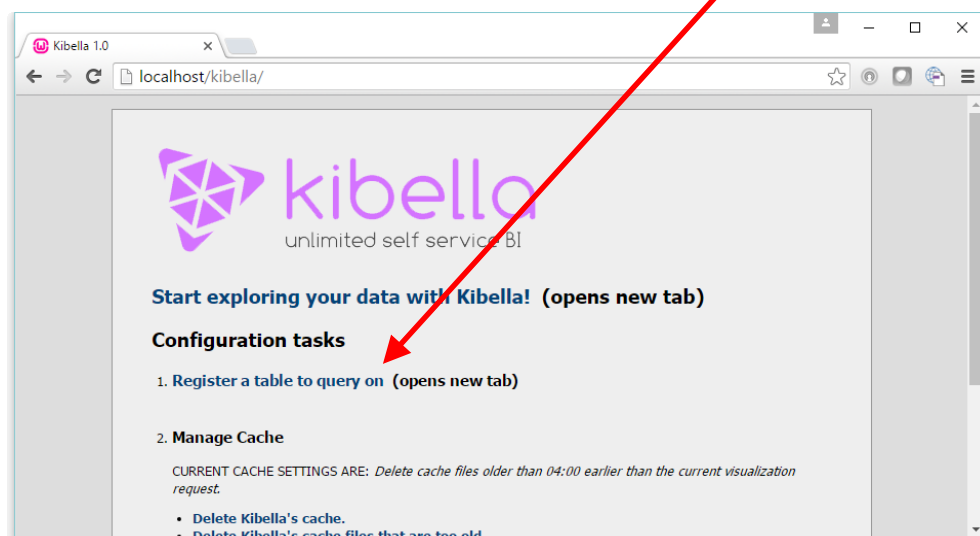
A registered table is identified by its ID which is made up of the *database filename in parenthesis followed by the table name*. Ex: (CENSUSINCOME)census.

4.3. Storing Dates and the "Date Format" inside the queried databases

All the columns containing dates (or any time-related value) must be listed here during exportation to Kibella:
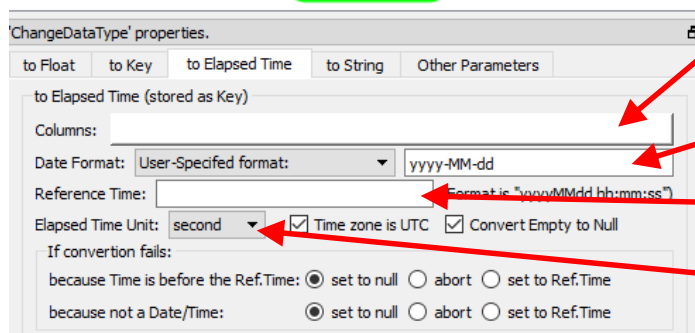
The "date" columns must contains dates using a specific format: The format to use is the "Elapsed Time" format: More Precisely: You can convert any date-format to the right format for Kibella using the "ChangeDataType" box inside Anatella: i.e. Use these parameters:

1: Select all the columns containing dates

2: Write the current format of the dates here

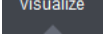**3: Leave this blank**

4: switch to "second"

(For Kibella, the Dates in the queried tables are stored as **seconds** elapsed since 1970/01/01 00:00:00).

5. Using Kibella

5.1. Creating new Visualizations

5.1.1. The New Visualization Wizard

You can use the Visualize tab to design data visualizations. You can save these visualizations, use them individually, or combine visualizations into a dashboard.

To start the **New Visualization** wizard, click on the Visualize tab `Visualize` at the top left of the page.

If you are already in the process of creating a visualization, you can click the New Visualization button in the toolbar to the right of the search bar to start a **New Visualization** wizard again.

The wizard guides you through the following steps:

5.1.1.1. Step 1: Choose the Visualization Type

Choose a visualization type when you start the New Visualization wizard:

	Area chart	Great for stacked timelines in which the total of all series is more important than comparing any two or more series. Less useful for assessing the relative change of unrelated data points as changes in a series lower down the stack will have a difficult to gauge effect on the series above it.
	Data table	The data table provides a detailed breakdown, in tabular format, of the results of a composed aggregation. Tip, a data table is available from many other charts by clicking grey bar at the bottom of the chart.
	Line chart	Often the best chart for high density time series. Great for comparing one series to another. Be careful with sparse sets as the connection between points can be misleading.
	Markdown widget	Useful for displaying explanations or instructions for dashboards.
	Metric	One big number for all of your one big number needs. Perfect for show a count of hits, or the exact average a numeric field.
	Pie chart	Pie charts are ideal for displaying the parts of some whole. For example, sales percentages by department.Pro Tip: Pie charts are best used sparingly, and with no more than 7 slices per pie.
	Tile map	Your source for geographic maps. Requires an elasticsearch geo_point field. More specifically, a field that is mapped as type:geo_point with latitude and longitude coordinates.
	Vertical bar chart	The goto chart for oh-so-many needs. Great for time and non-time data. Stacked or grouped, exact numbers or percentages. If you are not sure which chart your need, you could do worse than to start here.

You can also load a save visualization that you created earlier. The saved visualization selector includes a text field to filter by visualization name and a link to the Object Editor, accessible through **Settings > Edit Saved Objects**, to manage your saved visualizations.

If your new visualization is a Markdown widget, selecting that type takes you to a text entry field where you enter the text to display in the widget. For all other types of visualization, selecting the type takes you to data source selection.
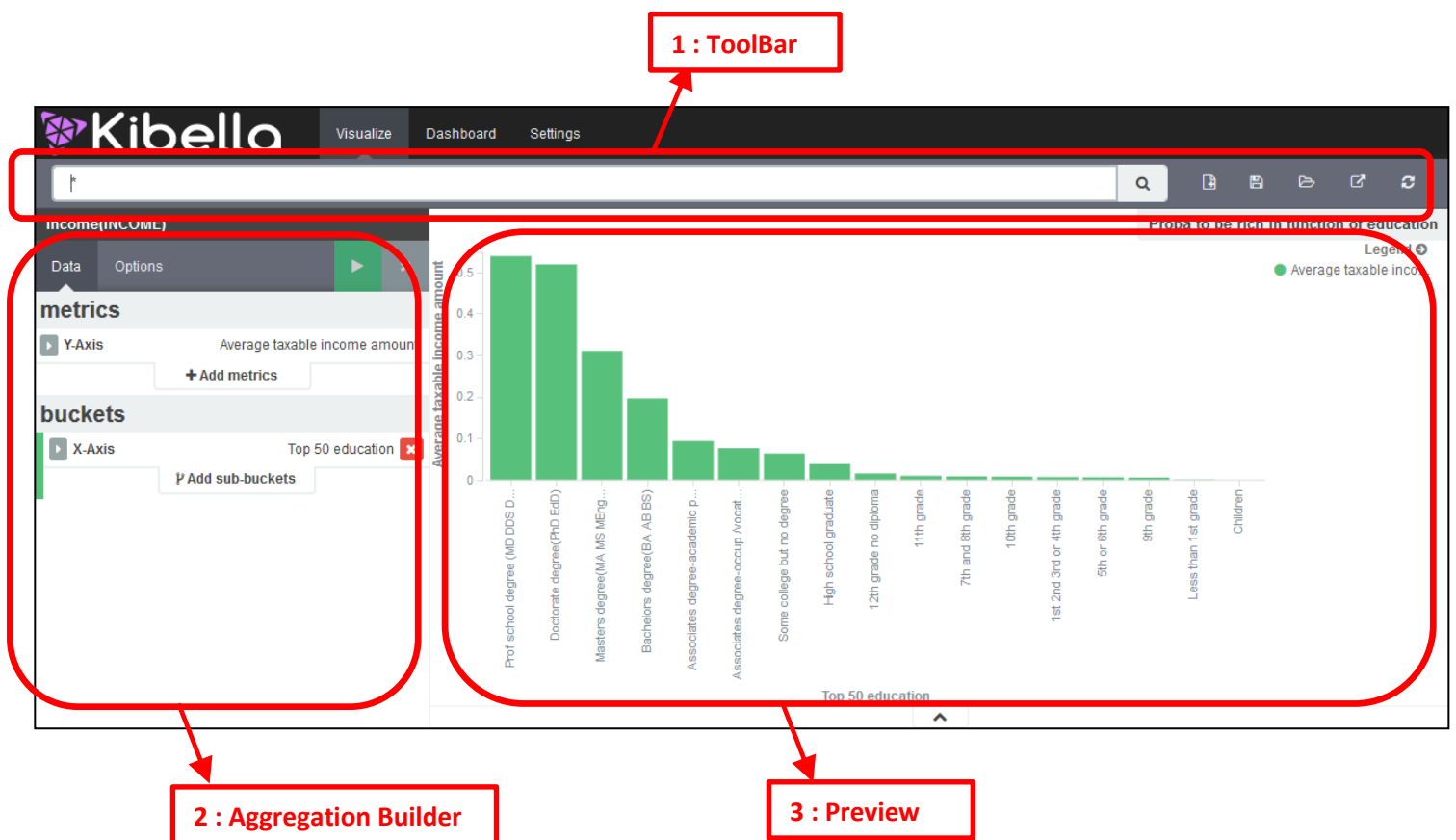
5.1.1.2. Step 2: Choose a Data Source

You can choose a new or saved search to serve as the data source for your visualization. Searches are associated with an index or a set of indexes. When you select new search on a system with multiple indices configured, select an index pattern from the drop-down to bring up the visualization editor.

When you create a visualization from a saved search and save the visualization, the search is tied to the visualization. When you make changes to the search that is linked to the visualization, the visualization updates automatically.

5.1.1.3. Step 3: The Visualization Editor

The visualization editor enables you to configure and edit visualizations. The visualization editor has the following main elements:

5.1.1.3.1. The Toolbar

The toolbar has a search field for interactive data searches, as well as controls to manage saving and loading visualizations. For visualizations based on saved searches, the search bar is grayed out. To edit the search, replacing the saved search with the edited version, double-click the search field.

The toolbar at the right of the search box has buttons for creating new visualizations, saving the current visualization, loading an existing visualization, sharing or embedding the visualization, and refreshing the data for the current visualization.
Aggregation Builder

5.1.1.3.2. The aggregation builder

Use the aggregation builder on the left of the page to configure the metric and bucket aggregations used in your visualization. Buckets are analogous to SQL GROUP BY statements.

Bar, line, or area chart visualizations use *metrics* for the y-axis and *buckets* are used for the x-axis, segment bar colors, and row/column splits. For pie charts, use the metric for the slice size and the bucket for the number of slices.

Choose the metric aggregation for your visualization's Y axis, such as count, average, sum, min, max, or cardinality (unique count). Use bucket aggregations for the visualization's X axis, color slices, and row/column splits. Common bucket aggregations include date histogram, range, terms, filters, and significant terms.

5.1.1.3.3. Preview

The preview canvas displays a preview of the visualization you've defined in the aggregation builder.
To refresh the visualization preview, clicking the Refresh button on the toolbar.

5.2. Area Chart Visualisation

This chart's Y axis is the *metrics* axis. The following aggregations are available for this axis:

- **Count:** The count aggregation returns a raw count of the elements in the selected index pattern.
- **Average:** This aggregation returns the average of a numeric field. Select a field from the drop-down.
- **Sum:** The sum aggregation returns the total sum of a numeric field. Select a field from the drop-down.
- **Min:** The min aggregation returns the minimum value of a numeric field. Select a field from the drop-down.
- **Max:** The max aggregation returns the maximum value of a numeric field. Select a field from the drop-down.
- **Unique Count:** The cardinality aggregation returns the number of unique values in a field. Select a field from the drop-down.

You can add an aggregation by clicking the **+ Add Aggregation** button.

The X axis of this chart is the buckets axis. You can define buckets for the X axis, for a split area on the chart, or for split charts.

This chart's X axis supports the following aggregations:

- **Date Histogram**
  A date histogram is built from a numeric field and organized by date. You can specify a time frame for the intervals in seconds, minutes, hours, days, weeks, months, or years. You can also specify a **custom** interval frame by selecting Custom as the interval and specifying a number and a time unit in the text field. Custom interval time units are **s** for seconds, **m** for minutes, **h** for hours, **d** for days, **w** for weeks, and **y** for years. Different units support different levels of precision, down to one second.

- **Histogram**
  A standard histogram is built from a numeric field. Specify an integer interval for this field. Select the Show empty buckets checkbox to include empty intervals in the histogram.

- **Range**
  With a range aggregation, you can specify ranges of values for a numeric field. Click **Add Range** to add a set of range endpoints. Click the red **(x)** symbol to remove a range.

- **Date Range**
  A date range aggregation reports values that are within a range of dates that you specify. You can specify the ranges for the dates using *date math* expressions. Click **Add Range** to add a set of range endpoints. Click the red **(/)** symbol to remove a range. The **From** and **To** values in Kibella must be specified as a string date using the template YYYY-MM-DD HH:MM:SS, YYYY/MM/DD HH:MM:SS, YYYY-MM-DD or YYYY/MM/DD. Some examples of valid dates are:
    - 2015/12/28 14:15:38
    - 2015-12-28 14:15:38
    - 2015/12/28
    - 2015-12-28

  The **from** and **to** values must *not* be enclosed in quotes.

- **Terms**
  A terms aggregation enables you to specify the top or bottom n elements of a given field to display, ordered by count or a custom metric.

- **Filters**
  You can specify a set of *filters* for the data. You can specify a filter as a query string or in JSON format, just as in the Discover search bar. Click **Add Filter** to add another filter. Click the Label button 🏷 to open the label field, where you can type in a name to display on the visualization.

Once you've specified an X axis aggregation, you can define sub-aggregations to refine the visualization. Click **+ Add Sub Aggregation** to define a sub-aggregation, then choose **Split Area** or **Split Chart**, then select a sub-aggregation from the list of types.

When multiple aggregations are defined on a chart's axis, you can use the up or down arrows to the right of the aggregation's type to change the aggregation's priority.

Enter a string in the **Custom Label** field to change the display label. For example, a chart of dates with incident counts can display dates in chronological order, or you can raise the priority of the incident-reporting aggregation to show the most active dates first. The chronological order might show a time-dependent pattern in incident count, and sorting by active dates can reveal particular outliers in your data.

The availability of these options varies depending on the aggregation you choose.

Select the Options tab to change the following aspects of the chart:
- **Chart Mode:** When you have multiple Y-axis aggregations defined for your chart, you can use this drop-down to affect how the aggregations display on the chart:

- o *Stacked*: Stacks the aggregations on top of each other.
- o *Overlap*: The aggregations overlap, with translucency indicating areas of overlap.
- o *Wiggle*: Displays the aggregations as a streamgraph.
- o *Percentage*: Displays each aggregation as a proportion of the total.
- o *Silhouette*: Displays each aggregation as variance from a central line.
- **Smooth Lines:** Check this box to curve the top boundary of the area from point to point.
- **Set Y-Axis Extents:** Check this box and enter values in the y-max and y-min fields to set the Y axis to specific values.
- **Scale Y-Axis to Data Bounds:** The default Y axis bounds are zero and the maximum value returned in the data. Check this box to change both upper and lower bounds to match the values returned in the data.
- **Show Tooltip:** Check this box to enable the display of tooltips.
- **Show Legend:** Check this box to enable the display of a legend next to the chart.

To display the raw data behind the visualization, click the bar at the bottom of the container. Tabs with detailed information about the raw data replace the visualization:
- **Table.** A representation of the underlying data, presented as a paginated data grid. You can sort the items in the table by clicking on the table headers at the top of each column.
- **Request.** The raw request used to query the server, presented in JSON format.
- **Response.** The raw response from the server, presented in JSON format.
- **Statistics.** A summary of the statistics related to the request and the response, presented as a data grid. The data grid includes the query duration, the request duration, the total number of records found on the server, and the index pattern used to make the query.
- **(at the bottom of any of the detailed information tabs) Raw or Formatted links**: Export the raw data behind the visualization as a comma-separated-values (CSV) file. A raw export contains the data as it is stored in Kibella. A formatted export contains the results of any applicable Kibella field formatters.

5.3. Data Table Visualization

The following aggregations are available:

- **Count:** The count aggregation returns a raw count of the elements in the selected index pattern.
- **Average:** This aggregation returns the average of a numeric field. Select a field from the drop-down.
- **Sum:** The sum aggregation returns the total sum of a numeric field. Select a field from the drop-down.
- **Min:** The min aggregation returns the minimum value of a numeric field. Select a field from the drop-down.
- **Max:** The max aggregation returns the maximum value of a numeric field. Select a field from the drop-down.
- **Unique Count:** The cardinality aggregation returns the number of unique values in a field. Select a field from the drop-down.

You can add an aggregation by clicking the **+ Add Aggregation** button.

Enter a string in the Custom Label field to change the display label.

The rows of the data table are called buckets. You can define buckets to split the table into rows or to split the table into additional tables.

Each bucket type supports the following aggregations:

- **Date Histogram**
  A date histogram is built from a numeric field and organized by date. You can specify a time frame for the intervals in seconds, minutes, hours, days, weeks, months, or years. You can also specify a **custom** interval frame by selecting Custom as the interval and specifying a number and a time unit in the text field. Custom interval time units are **s** for seconds, **m** for minutes, **h** for hours, **d** for days, **w** for weeks, and **y** for years. Different units support different levels of precision, down to one second.

- **Histogram**
  A standard histogram is built from a numeric field. Specify an integer interval for this field. Select the Show empty buckets checkbox to include empty intervals in the histogram.

- **Range**
  With a range aggregation, you can specify ranges of values for a numeric field. Click **Add Range** to add a set of range endpoints. Click the red **(x)** symbol to remove a range.

- **Date Range**
  A date range aggregation reports values that are within a range of dates that you specify. You can specify the ranges for the dates using *date math* expressions. Click **Add Range** to add a set of range endpoints. Click the red **(/)** symbol to remove a range. The **From** and **To** values in Kibella must be specified as a string date using the template YYYY-MM-DD HH:MM:SS, YYYY/MM/DD HH:MM:SS, YYYY-MM-DD or YYYY/MM/DD. Some examples of valid dates are:
  - 2015/12/28 14:15:38
  - 2015-12-28 14:15:38
  - 2015/12/28
  - 2015-12-28
  The **from** and **to** values must *not* be enclosed in quotes.

- **Terms**
  A terms aggregation enables you to specify the top or bottom n elements of a given field to display, ordered by count or a custom metric.

- **Filters**
  You can specify a set of *filters* for the data. You can specify a filter as a query string or in JSON format, just as in the Discover search bar. Click **Add Filter** to add another filter. Click the Label button 🏷 to open the label field, where you can type in a name to display on the visualization.

Once you've specified an bucket type aggregation, you can define sub-aggregations to refine the visualization. Click **+ Add Sub Aggregation** to define a sub-aggregation, then choose **Split Rows** or **Split Table**, then select a sub-aggregation from the list of types.

When multiple aggregations are defined on a chart's axis, you can use the up or down arrows to the right of the aggregation's type to change the aggregation's priority.

Enter a string in the **Custom Label** field to change the display label.

Select the Options tab to change the following aspects of the table:

- **Per Page:** This field controls the pagination of the table. The default value is ten rows per page.
- **Show metrics for every bucket/level:** Check this box to display the intermediate results for each bucket aggregation.
- **Show partial rows:** Check this box to display a row even when there is no result.

To display the raw data behind the visualization, click the bar at the bottom of the container. Tabs with detailed information about the raw data replace the visualization:

- **Table.** A representation of the underlying data, presented as a paginated data grid. You can sort the items in the table by clicking on the table headers at the top of each column.
- **Request.** The raw request used to query the server, presented in JSON format.
- **Response.** The raw response from the server, presented in JSON format.
- **Statistics.** A summary of the statistics related to the request and the response, presented as a data grid. The data grid includes the query duration, the request duration, the total number of records found on the server, and the index pattern used to make the query.
- **(at the bottom of any of the detailed information tabs) Raw or Formatted links**: Export the raw data behind the visualization as a comma-separated-values (CSV) file. A raw export contains the data as it is stored in Kibella. A formatted export contains the results of any applicable Kibella field formatters.

5.4. Line Chart Visualization

This chart's Y axis is the *metrics* axis. The following aggregations are available for this axis:

- **Count:** The count aggregation returns a raw count of the elements in the selected index pattern.
- **Average:** This aggregation returns the average of a numeric field. Select a field from the drop-down.
- **Sum:** The sum aggregation returns the total sum of a numeric field. Select a field from the drop-down.
- **Min:** The min aggregation returns the minimum value of a numeric field. Select a field from the drop-down.
- **Max:** The max aggregation returns the maximum value of a numeric field. Select a field from the drop-down.
- **Unique Count:** The cardinality aggregation returns the number of unique values in a field. Select a field from the drop-down.

You can add an aggregation by clicking the **+ Add Aggregation** button.

Enter a string in the Custom Label field to change the display label.

Before you choose a buckets aggregation, specify if you are splitting slices within a single chart or splitting into multiple charts. A multiple chart split must run before any other aggregations. When you split a chart, you can change if the splits are displayed in a row or a column by clicking the **Rows | Columns** selector.

The X axis of this chart is the buckets axis. You can define buckets for the X axis, for a split area on the chart, or for split charts.

This chart's X axis supports the following aggregations:
- **Date Histogram**
  A date histogram is built from a numeric field and organized by date. You can specify a time frame for the intervals in seconds, minutes, hours, days, weeks, months, or years. You can also specify a **custom** interval frame by selecting Custom as the interval and specifying a number and a time unit in the text field. Custom interval time units are **s** for seconds, **m** for minutes,

h for hours, **d** for days, **w** for weeks, and **y** for years. Different units support different levels of precision, down to one second.

- **Histogram**
  A standard histogram is built from a numeric field. Specify an integer interval for this field. Select the Show empty buckets checkbox to include empty intervals in the histogram.

- **Range**
  With a range aggregation, you can specify ranges of values for a numeric field. Click **Add Range** to add a set of range endpoints. Click the red **(x)** symbol to remove a range.

- **Date Range**
  A date range aggregation reports values that are within a range of dates that you specify. You can specify the ranges for the dates using *date math* expressions. Click **Add Range** to add a set of range endpoints. Click the red **(/)** symbol to remove a range. The **From** and **To** values in Kibella must be specified as a string date using the template YYYY-MM-DD HH:MM:SS, YYYY/MM/DD HH:MM:SS, YYYY-MM-DD or YYYY/MM/DD. Some examples of valid dates are:
  - o 2015/12/28 14:15:38
  - o 2015-12-28 14:15:38
  - o 2015/12/28
  - o 2015-12-28

  The **from** and **to** values must *not* be enclosed in quotes.

- **Terms**
  A terms aggregation enables you to specify the top or bottom n elements of a given field to display, ordered by count or a custom metric.

- **Filters**
  You can specify a set of *filters* for the data. You can specify a filter as a query string or in JSON format, just as in the Discover search bar. Click **Add Filter** to add another filter. Click the Label button 🏷 to open the label field, where you can type in a name to display on the visualization.

Once you've specified an X axis aggregation, you can define sub-aggregations to refine the visualization. Click **+ Add Sub Aggregation** to define a sub-aggregation, then choose **Split Area** or **Split Chart**, then select a sub-aggregation from the list of types.

When multiple aggregations are defined on a chart's axis, you can use the up or down arrows to the right of the aggregation's type to change the aggregation's priority.

Enter a string in the **Custom Label** field to change the display label.

The availability of these options varies depending on the aggregation you choose.

Select the Options tab to change the following aspects of the chart:

- **Y-Axis Scale:** You can select linear, log, or square root scales for the chart's Y axis. You can use a log scale to display data that varies exponentially, such as a compounding interest chart, or a square root scale to regularize the display of data sets with variabilities that are themselves highly variable. This kind of data, where the variability is itself variable over the domain being examined, is known as heteroscedastic data. For example, if a data set of height versus weight has a relatively narrow range of variability at the short end of height, but a wider range at the taller end, the data set is heteroscedastic.

- **Smooth Lines:** Check this box to curve the line from point to point. Bear in mind that smoothed lines necessarily affect the representation of your data and create a potential for ambiguity.

- **Show Connecting Lines:** Check this box to draw lines between the points on the chart.

- **Show Circles:** Check this box to draw each data point on the chart as a small circle.

- **Current time marker:** For charts of time-series data, check this box to draw a red line on the current time.
- **Set Y-Axis Extents:** Check this box and enter values in the y-max and y-min fields to set the Y axis to specific values.
- **Show Tooltip:** Check this box to enable the display of tooltips.
- **Show Legend:** Check this box to enable the display of a legend next to the chart.

After changing options, click the green **Apply changes** button to update your visualization, or the grey **Discard changes** button to keep your visualization in its current state.

To display the raw data behind the visualization, click the bar at the bottom of the container. Tabs with detailed information about the raw data replace the visualization:

- **Table.** A representation of the underlying data, presented as a paginated data grid. You can sort the items in the table by clicking on the table headers at the top of each column.
- **Request.** The raw request used to query the server, presented in JSON format.
- **Response.** The raw response from the server, presented in JSON format.
- **Statistics.** A summary of the statistics related to the request and the response, presented as a data grid. The data grid includes the query duration, the request duration, the total number of records found on the server, and the index pattern used to make the query.
- **(at the bottom of any of the detailed information tabs) Raw or Formatted links**: Export the raw data behind the visualization as a comma-separated-values (CSV) file. A raw export contains the data as it is stored in Kibella. A formatted export contains the results of any applicable Kibella field formatters.

5.5. Bubble Chart Visualization

You can convert a line chart visualization to a bubble chart by performing the following steps:

1. Click Add Metrics for the visualization's Y axis, then select Dot Size.
2. Select a metric aggregation from the drop-down list.
3. In the Options tab, uncheck the Show Connecting Lines box.
4. Click the Apply changes button.

5.6. Markdown Widget Visualization

The Markdown widget is a text entry field that accepts GitHub-flavored Markdown text. Kibella renders the text you enter in this field and displays the results on the dashboard. You can click the Help link to go to the help page for GitHub flavored Markdown. Click Apply to display the rendered text in the Preview pane or Discard to revert to a previous version.

5.7. Metric Visualization

A metric visualization displays a single number for each aggregation you select:

- **Count:** The count aggregation returns a raw count of the elements in the selected index pattern.

- **Average:** This aggregation returns the average of a numeric field. Select a field from the drop-down.
- **Sum:** The sum aggregation returns the total sum of a numeric field. Select a field from the drop-down.
- **Min:** The min aggregation returns the minimum value of a numeric field. Select a field from the drop-down.
- **Max:** The max aggregation returns the maximum value of a numeric field. Select a field from the drop-down.
- **Unique Count:** The cardinality aggregation returns the number of unique values in a field. Select a field from the drop-down.

You can add an aggregation by clicking the **+ Add Aggregation** button.

Enter a string in the Custom Label field to change the display label.

Click the Options tab to display the font size slider.

To display the raw data behind the visualization, click the bar at the bottom of the container. Tabs with detailed information about the raw data replace the visualization:
- **Table.** A representation of the underlying data, presented as a paginated data grid. You can sort the items in the table by clicking on the table headers at the top of each column.
- **Request.** The raw request used to query the server, presented in JSON format.
- **Response.** The raw response from the server, presented in JSON format.
- **Statistics.** A summary of the statistics related to the request and the response, presented as a data grid. The data grid includes the query duration, the request duration, the total number of records found on the server, and the index pattern used to make the query.
- **(at the bottom of any of the detailed information tabs) Raw or Formatted links**: Export the raw data behind the visualization as a comma-separated-values (CSV) file. A raw export contains the data as it is stored in Kibella. A formatted export contains the results of any applicable Kibella field formatters.

5.8. Pie Charts Visualization

The slice size of a pie chart is determined by the metrics aggregation. The following aggregations are available for this axis:

- **Count:** The count aggregation returns a raw count of the elements in the selected index pattern.
- **Sum:** The sum aggregation returns the total sum of a numeric field. Select a field from the drop-down.
- **Unique Count:** The cardinality aggregation returns the number of unique values in a field. Select a field from the drop-down.

Enter a string in the Custom Label field to change the display label.

You can specify any of the following bucket aggregations for your pie chart:
- **Date Histogram**
  A date histogram is built from a numeric field and organized by date. You can specify a time frame for the intervals in seconds, minutes, hours, days, weeks, months, or years. You can also specify a **custom** interval frame by selecting Custom as the interval and specifying a number

and a time unit in the text field. Custom interval time units are **s** for seconds, **m** for minutes, **h** for hours, **d** for days, **w** for weeks, and **y** for years. Different units support different levels of precision, down to one second.

- **Histogram**
A standard histogram is built from a numeric field. Specify an integer interval for this field. Select the Show empty buckets checkbox to include empty intervals in the histogram.

- **Range**
With a range aggregation, you can specify ranges of values for a numeric field. Click **Add Range** to add a set of range endpoints. Click the red **(x)** symbol to remove a range.

- **Date Range**
A date range aggregation reports values that are within a range of dates that you specify. You can specify the ranges for the dates using *date math* expressions. Click **Add Range** to add a set of range endpoints. Click the red **(/)** symbol to remove a range. The **From** and **To** values in Kibella must be specified as a string date using the template YYYY-MM-DD HH:MM:SS, YYYY/MM/DD HH:MM:SS, YYYY-MM-DD or YYYY/MM/DD. Some examples of valid dates are:
  - 2015/12/28 14:15:38
  - 2015-12-28 14:15:38
  - 2015/12/28
  - 2015-12-28

The **from** and **to** values must *not* be enclosed in quotes.

- **Terms**
A terms aggregation enables you to specify the top or bottom n elements of a given field to display, ordered by count or a custom metric.

- **Filters**
You can specify a set of *filters* for the data. You can specify a filter as a query string or in JSON format, just as in the Discover search bar. Click **Add Filter** to add another filter. Click the Label button 🏷 to open the label field, where you can type in a name to display on the visualization.

Once you've specified an X axis aggregation, you can define sub-aggregations to refine the visualization. Click **+ Add Sub Aggregation** to define a sub-aggregation, then choose **Split Area** or **Split Chart**, then select a sub-aggregation from the list of types.

When multiple aggregations are defined on a chart's axis, you can use the up or down arrows to the right of the aggregation's type to change the aggregation's priority.

Enter a string in the **Custom Label** field to change the display label.

The availability of these options varies depending on the aggregation you choose.

Select the Options tab to change the following aspects of the chart:
- **Donut:** Display the chart as a sliced ring instead of a sliced pie.
- **Show Tooltip:** Check this box to enable the display of tooltips.
- **Show Legend:** Check this box to enable the display of a legend next to the chart.

After changing options, click the green **Apply changes** button to update your visualization, or the grey **Discard changes** button to keep your visualization in its current state.

To display the raw data behind the visualization, click the bar at the bottom of the container. Tabs with detailed information about the raw data replace the visualization:
- **Table.** A representation of the underlying data, presented as a paginated data grid. You can sort the items in the table by clicking on the table headers at the top of each column.

- **Request.** The raw request used to query the server, presented in JSON format.
- **Response.** The raw response from the server, presented in JSON format.
- **Statistics.** A summary of the statistics related to the request and the response, presented as a data grid. The data grid includes the query duration, the request duration, the total number of records found on the server, and the index pattern used to make the query.
- **(at the bottom of any of the detailed information tabs) Raw or Formatted links**: Export the raw data behind the visualization as a comma-separated-values (CSV) file. A raw export contains the data as it is stored in Kibella. A formatted export contains the results of any applicable Kibella field formatters.

5.9. Tile Maps Visualization

Tile Maps are currently not supported Inside the Early-Access version of Kibella.

A tile map displays a geographic area overlaid with circles keyed to the data determined by the buckets you specify.

The default metrics aggregation for a tile map is the Count aggregation. You can select any of the following aggregations as the metrics aggregation:

- **Count:** The count aggregation returns a raw count of the elements in the selected index pattern.
- **Average:** This aggregation returns the average of a numeric field. Select a field from the drop-down.
- **Sum:** The sum aggregation returns the total sum of a numeric field. Select a field from the drop-down.
- **Min:** The min aggregation returns the minimum value of a numeric field. Select a field from the drop-down.
- **Max:** The max aggregation returns the maximum value of a numeric field. Select a field from the drop-down.
- **Unique Count:** The cardinality aggregation returns the number of unique values in a field. Select a field from the drop-down.

The buckets aggregations determine what information is being retrieved from your data set.

Before you choose a buckets aggregation, specify if you are splitting the chart or displaying the buckets as **Geo Coordinates** on a single chart. A multiple chart split must run before any other aggregations.

Tile maps use the **Geohash** aggregation as their initial aggregation. Select a field, typically coordinates, from the drop-down. The **Precision** slider determines the granularity of the results displayed on the map.

Once you've specified a buckets aggregation, you can define sub-aggregations to refine the visualization. Tile maps only support sub-aggregations as split charts. Click **+ Add Sub Aggregation**, then **Split Chart** to select a sub-aggregation from the list of types:
- **Date Histogram**
  A date histogram is built from a numeric field and organized by date. You can specify a time frame for the intervals in seconds, minutes, hours, days, weeks, months, or years. You can also specify a **custom** interval frame by selecting Custom as the interval and specifying a number

and a time unit in the text field. Custom interval time units are **s** for seconds, **m** for minutes, **h** for hours, **d** for days, **w** for weeks, and **y** for years. Different units support different levels of precision, down to one second.

- **Histogram**
A standard histogram is built from a numeric field. Specify an integer interval for this field. Select the Show empty buckets checkbox to include empty intervals in the histogram.

- **Range**
With a range aggregation, you can specify ranges of values for a numeric field. Click **Add Range** to add a set of range endpoints. Click the red **(x)** symbol to remove a range.

- **Date Range**
A date range aggregation reports values that are within a range of dates that you specify. You can specify the ranges for the dates using *date math* expressions. Click **Add Range** to add a set of range endpoints. Click the red **(/)** symbol to remove a range. The **From** and **To** values in Kibella must be specified as a string date using the template YYYY-MM-DD HH:MM:SS, YYYY/MM/DD HH:MM:SS, YYYY-MM-DD or YYYY/MM/DD. Some examples of valid dates are:
  - 2015/12/28 14:15:38
  - 2015-12-28 14:15:38
  - 2015/12/28
  - 2015-12-28

The **from** and **to** values must *not* be enclosed in quotes.

- **Terms**
A terms aggregation enables you to specify the top or bottom n elements of a given field to display, ordered by count or a custom metric.

- **Filters**
You can specify a set of *filters* for the data. You can specify a filter as a query string or in JSON format, just as in the Discover search bar. Click **Add Filter** to add another filter. Click the Label button 🏷 to open the label field, where you can type in a name to display on the visualization.

- **Geohash**
The geohash aggregation displays points based on the geohash coordinates.

The availability of these options varies depending on the aggregation you choose.

Select the Options tab to change the following aspects of the chart:
- **Map type:** Select one of the following options from the drop-down.
  - *Scaled Circle Markers*: Scale the size of the markers based on the metric aggregation's value.
  - *Shaded Circle Markers*: Displays the markers with different shades based on the metric aggregation's value.
  - *Shaded Geohash Grid*: Displays the rectangular cells of the geohash grid instead of circular markers, with different shades based on the metric aggregation's value.
  - *Heatmap*: A heat map applies blurring to the circle markers and applies shading based on the amount of overlap. Heatmaps have the following options:
    - Radius: Sets the size of the individual heatmap dots.
    - Blur: Sets the amount of blurring for the heatmap dots.
    - Maximum zoom: Tilemaps in Kibella support 18 zoom levels. This slider defines the maximum zoom level at which the heatmap dots appear at full intensity.
    - Minimum opacity: Sets the opacity cutoff for the dots.
    - Show Tooltip: Check this box to have a tooltip with the values for a given dot when the cursor is on that dot.

- **Desaturate map tiles:** Desaturate the map's color in order to make the markers stand out more clearly.

After changing options, click the green **Apply changes** button to update your visualization, or the grey **Discard changes** button to keep your visualization in its current state.

Navigating the Map

Once your tilemap visualization is ready, you can explore the map in several ways:

- Click and hold anywhere on the map and move the cursor to move the map center. Hold Shift and drag a bounding box across the map to zoom in on the selection.

- Click the **Zoom In/Out** buttons to change the zoom level manually.

- Click the **Fit Data Bounds** button to automatically crop the map boundaries to the geohash buckets that have at least one result.

- Click the **Latitude/Longitude Filter** button, then drag a bounding box across the map, to create a filter for the box coordinates.

To display the raw data behind the visualization, click the bar at the bottom of the container. Tabs with detailed information about the raw data replace the visualization:

- **Table.** A representation of the underlying data, presented as a paginated data grid. You can sort the items in the table by clicking on the table headers at the top of each column.
- **Request.** The raw request used to query the server, presented in JSON format.
- **Response.** The raw response from the server, presented in JSON format.
- **Statistics.** A summary of the statistics related to the request and the response, presented as a data grid. The data grid includes the query duration, the request duration, the total number of records found on the server, and the index pattern used to make the query.
- **(at the bottom of any of the detailed information tabs) Raw or Formatted links**: Export the raw data behind the visualization as a comma-separated-values (CSV) file. A raw export contains the data as it is stored in Kibella. A formatted export contains the results of any applicable Kibella field formatters.

5.10. Vertical Bar chart Visualization

This chart's Y axis is the *metrics* axis. The following aggregations are available for this axis:

- **Count:** The count aggregation returns a raw count of the elements in the selected index pattern.
- **Average:** This aggregation returns the average of a numeric field. Select a field from the drop-down.
- **Sum:** The sum aggregation returns the total sum of a numeric field. Select a field from the drop-down.
- **Min:** The min aggregation returns the minimum value of a numeric field. Select a field from the drop-down.
- **Max:** The max aggregation returns the maximum value of a numeric field. Select a field from the drop-down.
- **Unique Count:** The cardinality aggregation returns the number of unique values in a field. Select a field from the drop-down.

You can add an aggregation by clicking the **+ Add Aggregation** button.

The *buckets* aggregations determine what information is being retrieved from your data set.

Before you choose a buckets aggregation, specify if you are splitting slices within a single chart or splitting into multiple charts. A multiple chart split must run before any other aggregations. When you split a chart, you can change if the splits are displayed in a row or a column by clicking the **Rows | Columns** selector.

The X axis of this chart is the buckets axis. You can define buckets for the X axis, for a split area on the chart, or for split charts.

This chart's X axis supports the following aggregations:
- **Date Histogram**
  A date histogram is built from a numeric field and organized by date. You can specify a time frame for the intervals in seconds, minutes, hours, days, weeks, months, or years. You can also specify a **custom** interval frame by selecting Custom as the interval and specifying a number and a time unit in the text field. Custom interval time units are **s** for seconds, **m** for minutes, **h** for hours, **d** for days, **w** for weeks, and **y** for years. Different units support different levels of precision, down to one second.
- **Histogram**
  A standard histogram is built from a numeric field. Specify an integer interval for this field. Select the Show empty buckets checkbox to include empty intervals in the histogram.
- **Range**
  With a range aggregation, you can specify ranges of values for a numeric field. Click **Add Range** to add a set of range endpoints. Click the red **(x)** symbol to remove a range.
- **Date Range**
  A date range aggregation reports values that are within a range of dates that you specify. You can specify the ranges for the dates using *date math* expressions. Click **Add Range** to add a set of range endpoints. Click the red **(/)** symbol to remove a range. The **From** and **To** values in Kibella must be specified as a string date using the template YYYY-MM-DD HH:MM:SS, YYYY/MM/DD HH:MM:SS, YYYY-MM-DD or YYYY/MM/DD. Some examples of valid dates are:
  - 2015/12/28 14:15:38
  - 2015-12-28 14:15:38
  - 2015/12/28
  - 2015-12-28
  The **from** and **to** values must *not* be enclosed in quotes.
- **Terms**
  A terms aggregation enables you to specify the top or bottom n elements of a given field to display, ordered by count or a custom metric.
- **Filters**
  You can specify a set of *filters* for the data. You can specify a filter as a query string or in JSON format, just as in the Discover search bar. Click **Add Filter** to add another filter. Click the Label button to open the label field, where you can type in a name to display on the visualization.

Once you've specified an X axis aggregation, you can define sub-aggregations to refine the visualization. Click **+ Add Sub Aggregation** to define a sub-aggregation, then choose **Split Area** or **Split Chart**, then select a sub-aggregation from the list of types.

When multiple aggregations are defined on a chart's axis, you can use the up or down arrows to the right of the aggregation's type to change the aggregation's priority.

The availability of these options varies depending on the aggregation you choose.

Select the Options tab to change the following aspects of the chart:
- **Bar Mode:** When you have multiple Y-axis aggregations defined for your chart, you can use this drop-down to affect how the aggregations display on the chart:
  - Stacked: Stacks the aggregations on top of each other.
  - Percentage: Displays each aggregation as a proportion of the total.
  - Grouped: Groups the results horizontally by the lowest-priority sub-aggregation.
- **Show Tooltip:** Check this box to enable the display of tooltips.
- **Show Legend:** Check this box to enable the display of a legend next to the chart.
- **Scale Y-Axis to Data Bounds:** The default Y axis bounds are zero and the maximum value returned in the data. Check this box to change both upper and lower bounds to match the values returned in the data.

To display the raw data behind the visualization, click the bar at the bottom of the container. Tabs with detailed information about the raw data replace the visualization:
- **Table.** A representation of the underlying data, presented as a paginated data grid. You can sort the items in the table by clicking on the table headers at the top of each column.
- **Request.** The raw request used to query the server, presented in JSON format.
- **Response.** The raw response from the server, presented in JSON format.
- **Statistics.** A summary of the statistics related to the request and the response, presented as a data grid. The data grid includes the query duration, the request duration, the total number of records found on the server, and the index pattern used to make the query.
- **(at the bottom of any of the detailed information tabs) Raw or Formatted links**: Export the raw data behind the visualization as a comma-separated-values (CSV) file. A raw export contains the data as it is stored in Kibella. A formatted export contains the results of any applicable Kibella field formatters.

5.11. Dashboards

A Kibella dashboard displays a set of saved visualizations in groups that you can arrange freely. You can save a dashboard to share or reload at a later time.
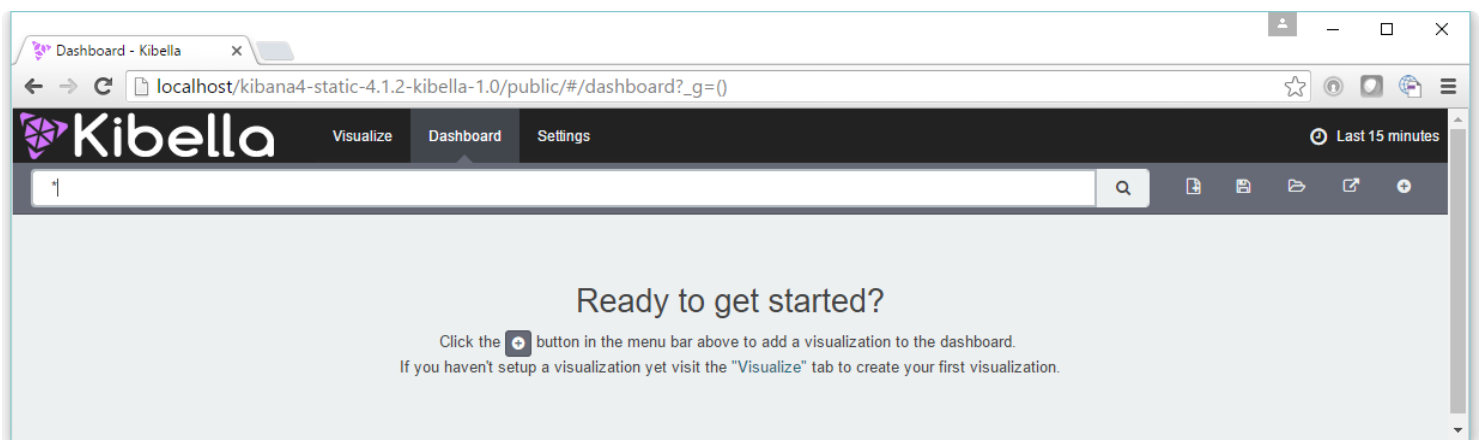
Here is a sample dashboard:
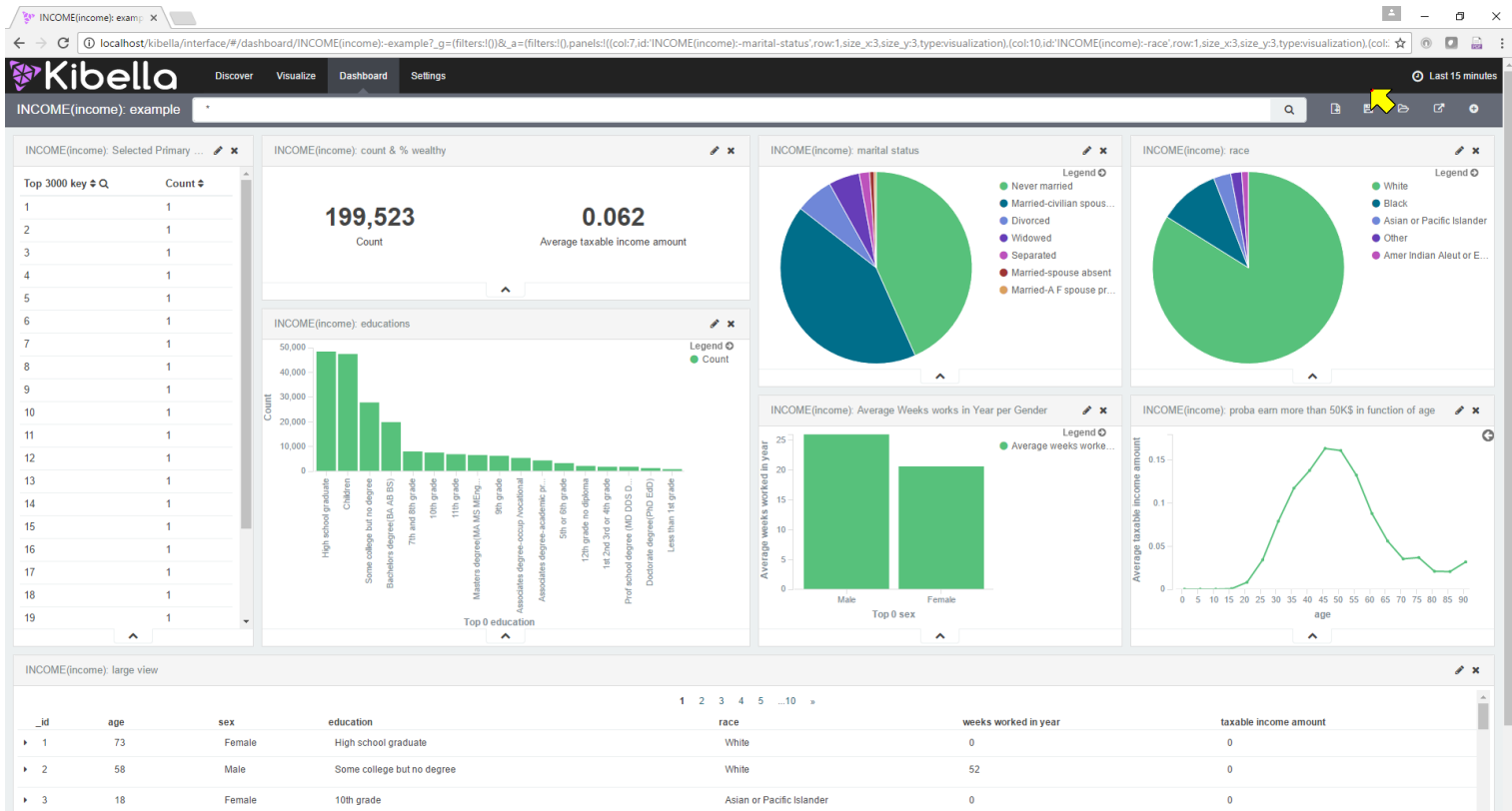
You need at least one saved visualization to use a dashboard.

5.11.1.1. Building a New Dashboard

The first time you click the Dashboard tab, Kibella displays an empty dashboard:

Build your dashboard by adding visualizations.

5.11.1.2. Adding Visualizations to a Dashboard

To add a visualization to the dashboard, click the **Add Visualization** ⊕ button in the toolbar panel. Select a saved visualization from the list. You can filter the list of visualizations by typing a filter string into the **Visualization Filter** field.

The visualization you select appears in a *container* on your dashboard.

5.11.1.3. Saving Dashboards

To save the dashboard, click the Save Dashboard button in the toolbar panel, enter a name for the dashboard in the Save As field, and click the Save button. By default, dashboards store the time period specified in the time filter when you save a dashboard. To disable this behavior, clear the Store time with dashboard box before clicking the Save button.

5.11.1.4. Loading a Saved Dashboard

Click the Load Saved Dashboard button to display a list of existing dashboards. The saved dashboard selector includes a text field to filter by dashboard name and a link to the Object Editor for managing your saved dashboards. You can also access the Object Editor by clicking Settings > Edit Saved Objects.

5.11.1.5. Sharing Dashboards

You can share dashboards with other users. You can share a direct link to the Kibella dashboard or embed the dashboard in your Web page.

Click the **Share** button to display HTML code to embed the dashboard in another Web page, along with a direct link to the dashboard. Click the copy button Copy to Clipboard button next to either option to copy the code or the link to your clipboard.

5.11.1.6. Embedding Dashboards

To embed a dashboard, copy the embed code from the Share display into your external web application.

5.11.2. Customizing Dashboard Elements

The visualizations in your dashboard are stored in resizable containers that you can arrange on the dashboard. This section discusses customizing these containers.

5.11.2.1. Moving Containers

Click and hold a container's header to move the container around the dashboard. Other containers will shift as needed to make room for the moving container. Release the mouse button to confirm the container's new location.

5.11.2.2. Resizing Containers

Move the cursor to the bottom right corner of the container until the cursor changes to point at the corner. After the cursor changes, click and drag the corner of the container to change the container's size. Release the mouse button to confirm the new container size.

5.11.2.3. Editing the Visualization inside a Container

Click the Edit button ✏ at the top right of a container to open the visualization in the Visualize page.

5.11.2.4. Removing Containers

Click the x icon at the top right corner of a container to remove that container from the dashboard. Removing a container from a dashboard does not delete the saved visualization in that container.

5.11.3. The raw data behind the visualization

To display the raw data behind the visualization, click the bar at the bottom of the container. Tabs with detailed information about the raw data replace the visualization:

- **Table.** A representation of the underlying data, presented as a paginated data grid. You can sort the items in the table by clicking on the table headers at the top of each column.
- **Request.** The raw request used to query the server, presented in JSON format.
- **Response.** The raw response from the server, presented in JSON format.
- **Statistics.** A summary of the statistics related to the request and the response, presented as a data grid. The data grid includes the query duration, the request duration, the total number of records found on the server, and the index pattern used to make the query.
- **(at the bottom of any of the detailed information tabs) Raw or Formatted links**: Export the raw data behind the visualization as a comma-separated-values (CSV) file. A raw export contains the data as it is stored in Kibella. A formatted export contains the results of any applicable Kibella field formatters.

5.11.4. Working with Filters.

5.11.4.1. Simple Filters

When you create a filter anywhere in Kibella, the filter conditions display in a green oval under the search text entry box:
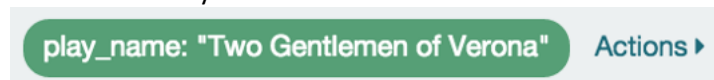

Hovering on the filter oval displays the following icons:

Here is a description of each icon:

- **Enable Filter**
  Click this icon to disable the filter without removing it. You can enable the filter again later by clicking the icon again. Disabled filters display a striped shaded color, green for inclusion filters and red for exclusion filters.

- **Pin Filter**
  Click this icon to pin a filter. Pinned filters persist across Kibella tabs. You can pin filters from the Dashboard tab, click on the Visualize tab, and those filters remain in place.
  Note: If you have a pinned filter and you're not seeing any query results, that your current tab's index pattern is one that the filter applies to.

- **Toggle Filter**
  Click this icon to toggle a filter. By default, filters are inclusion filters, and display in green. Only elements that match the filter are displayed. To change this to an exclusion filters, displaying only elements that don't match, toggle the filter. Exclusion filters display in red.

- **Remove Filter**
  Click this icon to remove a filter entirely.

- **Custom Filter**
  Click this icon to display a text field where you can customize the JSON representation of the filter. Click the Done button to update the filter with your changes.

To apply any of the filter actions to all the filters currently in place, click the Global Filter Actions button and select an action.

5.11.4.2. Filters on dates

Date Filters Inside the Visualization Panel:

> Inside the Visualization Panel, the calendar interface is not available. The only way to filter dates is by using the free box for normal filters at the top and filtering the date using ranges defined on seconds elapsed since 1970/01/01 00:00:00. An example of such a filter is:
> `1430611200 <= date and date <= 1462406400`
> …which filters the date variable `date` between `'03-May-2015 02:00:00'` and `'05-May-2016 02:00:00'`.

Inside the Dashboard Panel, the calendar interface is available to easily create different queries on dates: Click here:
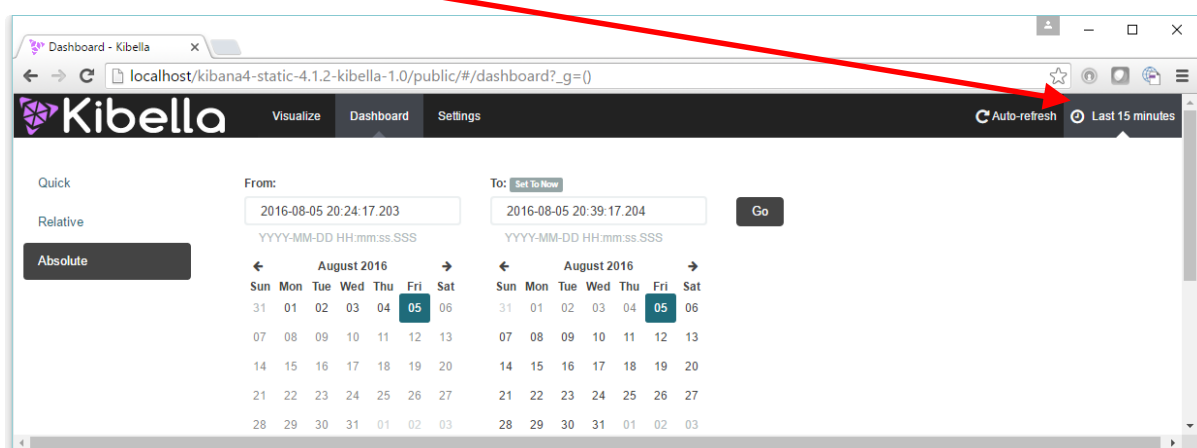
Inside the Early-Access version of Kibella, relative date values like "now-1w/w" (which appears as default in Kibella) are currently not supported.

## 5.11.5. Automatic Refresh of the Page

You can configure a refresh interval to automatically refresh the page with the latest index data. This periodically resubmits the search query.

When a refresh interval is set, it is displayed to the left of the Time Filter in the menu bar.

To set the refresh interval:

1. Click the **Time Filter** ⏰ in the upper right corner of the menu bar:
2. Click the **Refresh Interval** tab.
3. Choose a refresh interval from the list.

To automatically refresh the data, click the ⟳ **Auto-refresh** **Auto-refresh** button and select an autorefresh interval:
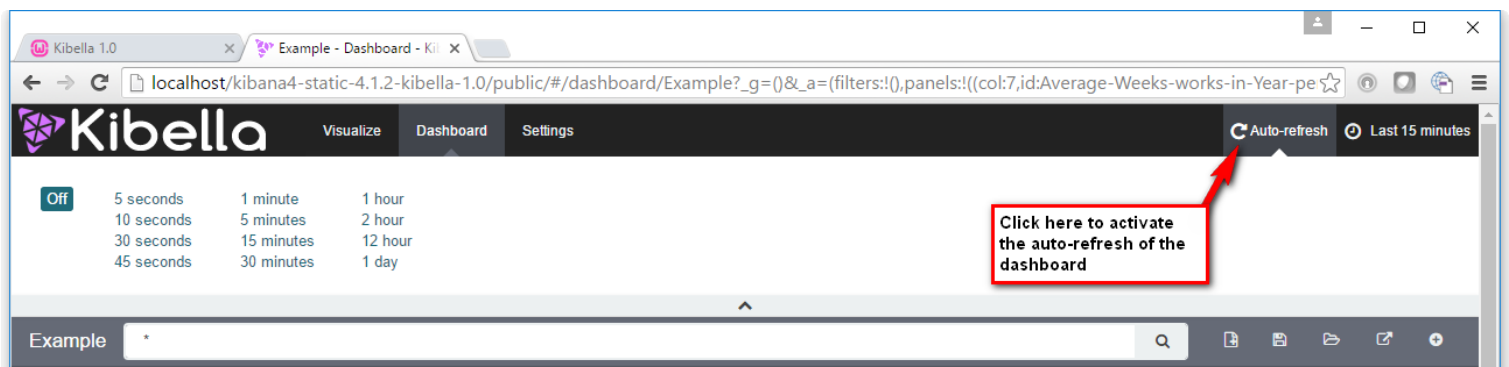
When auto-refresh is enabled, Kibella's top bar displays a pause button and the auto-refresh interval: ⏸ 1 hour . Click the **Pause** button to pause auto-refresh.

6. Deploying your dashboards in production

You need to copy to the production server:
*   the ""**<WampInstallDir>**/www/kibella/kibella.sqlite" file: This file contains the definition of all the visualizations and all the dashboards.
*   the .sqlite files that contains the queried datasets (from the kibella **data** dir)

7. Managing Several Kibella's installed on the same server

Already possible.
Still need to write Documentation.