Enhancements Release 1.06

DDMS

IVCC

LSTM
LIVERPOOL SCHOOL
OF TROPICAL MEDICINE

ddms@lstmed.ac.uk
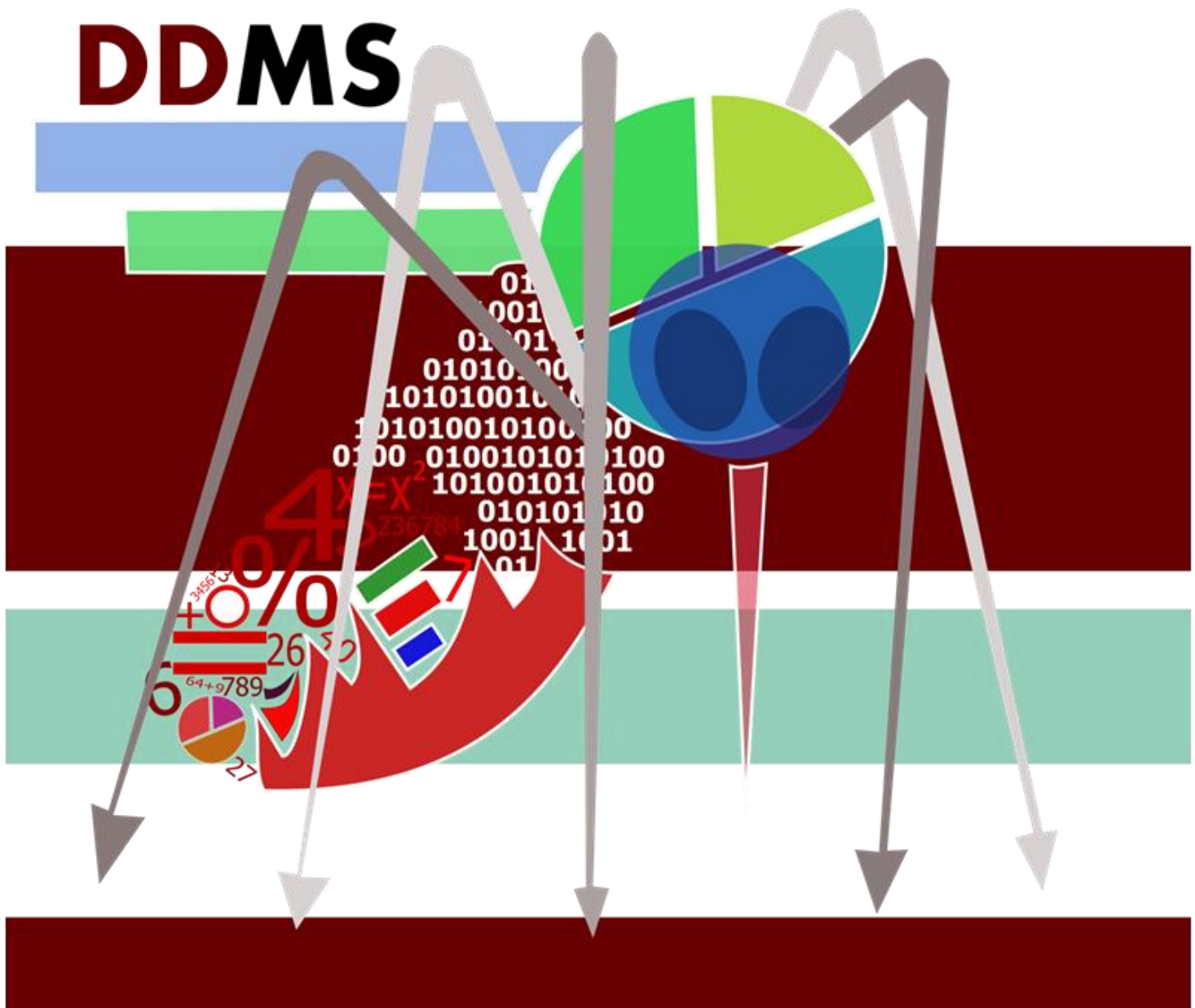
Contents

1.   Introduction

1.1.   Scope of the Training

DDMS 1.06 is a major release and contains four significant enhancements

1.   Kaleidoscopes: interactive drill-down maps and reports that are linked to the Kaleidoscope
2.   ETL data import technology: the ability to drop a spreadsheet into the DDMS with a wizard to guide you through the import of the data and the creation of a DDMS form based on the data in the spreadsheet
3.   Indicators: the definition of a ratio of two data fields, which are aggregated in query builders and Kaleidoscopes before the ratio is calculated
4.   DHIS2 interface: technology that uses a DDMS query result to automatically create the metadata in a DHIS2 install and push the aggregated data into DHIS2, once off or periodically

This 2 day course will explain these DDMS enhancements. The installation, configuration and use of DHIS2 is outside the scope of this course.

1.2.   Prerequisites

Participants are expected to be familiar with the DDMS, in particular the query builders and form creation.

This training is based on a copy of the Zambian database. For the Day 1 training participants should have a copy of the Zambian database on their laptop, patched to release 1.06.0010.

For day 2, the participants should have an empty DHIS2 install on their laptop. Note that having the DDMS and DHIS2 on a single computer can cause conflicts, both in the PostgreSQL configuration and possibly the Tomcat configuration. The trainer will assist in resolving these conflicts but the technicalities of configuring both systems is not within the scope of this course.

2. Landing Page

Although the login page has not changed, the DDMS has a new landing page.

✓ Log into the DDMS using username 'train' and password 'train'
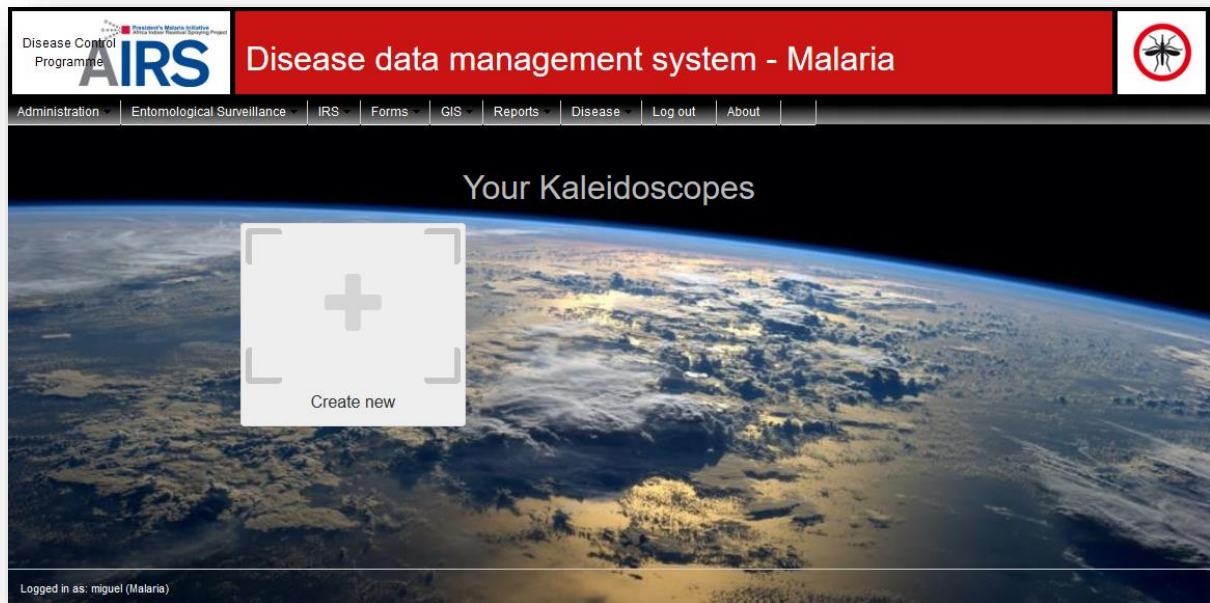
It replaces the 1.05 *Welcome* page which had little functionality. The intention of this page is to provide users with a basic dashboard. The use and creation of Kaleidoscopes will be detailed later in the course.

For optimal use of the landing page, a screen resolution of 1920 x 1080 is recommended. This will allow for 6 Kaleidoscope thumbnails without needing to scroll.

The menu structure has been enhanced but functions the same way as before.

3. Persisted Queries and Data sets

DDMS 1.06 introduces the new concepts of persisted queries and data sets.
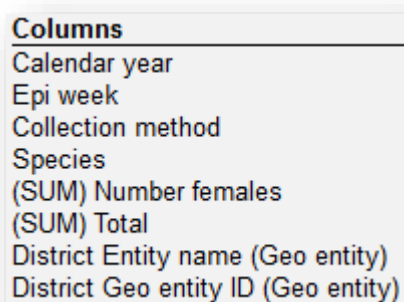
Technically, a data set is a metadata construct that adds metadata to a Class (or database table). Practically, this is a new table in the DDMS database, which can be created by the user in a variety of ways, i.e. by:

- Persisting a query
- Dropping a spreadsheet in the ETL data set manager
- Creating a form

Previously it was only possible to save a query. Saving a query creates a view in the database. A view is not a physical database table but a virtual table storing the SQL of the query. These views are accessible by the BIRT report writer but each time they are used the query is run to provide the results. This can be time consuming, depending on the query, the amount of data in the database and the amount of views used by the report.

In DDMS 1.06 it is additionally possible to persist a query. Persisting a query creates a data set, which includes a physical table in the database containing the query result.

- ✓ Open the *Mosquito collections and abundance* query builder
- ✓ Create the following query:

- ✓ Run the query

As before, you can save this query, but now you can also persist the query by ticking the *Persist result* check box:

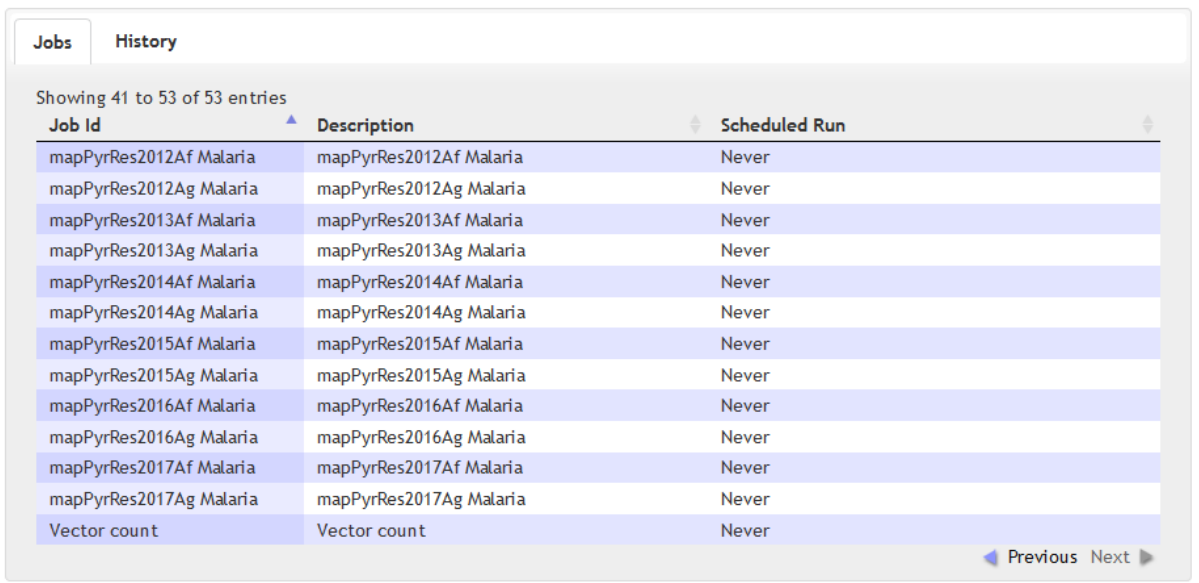

Ticking the box on its own does nothing, but now when you save the query a data set is created.

✓ Tick the *Persist result* check box and save the query as 'Vector count'

By doing this, the resulting data set can be used in a Kaleidoscope, a BIRT report and the DHIS2 interface.

The data set is a static table, meaning that it is not automatically updated when the data are changed. When you use the data set in a report, Kaleidoscope or the DHIS2 interface, you must remember to save the query so that it contains the most recent data. This can be done automatically from time to time by scheduling the refresh in the DDMS scheduler. A job for the scheduler is created automatically when you persist a query result.

✓ Open the Scheduler and find the newly created job. It will be at the end of the job list.

Jobs	History		
Showing 41 to 53 of 53 entries			
Job Id ▲	**Description**	**Scheduled Run**	
mapPyrRes2012Af Malaria	mapPyrRes2012Af Malaria	Never	
mapPyrRes2012Ag Malaria	mapPyrRes2012Ag Malaria	Never	
mapPyrRes2013Af Malaria	mapPyrRes2013Af Malaria	Never	
mapPyrRes2013Ag Malaria	mapPyrRes2013Ag Malaria	Never	
mapPyrRes2014Af Malaria	mapPyrRes2014Af Malaria	Never	
mapPyrRes2014Ag Malaria	mapPyrRes2014Ag Malaria	Never	
mapPyrRes2015Af Malaria	mapPyrRes2015Af Malaria	Never	
mapPyrRes2015Ag Malaria	mapPyrRes2015Ag Malaria	Never	
mapPyrRes2016Af Malaria	mapPyrRes2016Af Malaria	Never	
mapPyrRes2016Ag Malaria	mapPyrRes2016Ag Malaria	Never	
mapPyrRes2017Af Malaria	mapPyrRes2017Af Malaria	Never	
mapPyrRes2017Ag Malaria	mapPyrRes2017Ag Malaria	Never	
Vector count	Vector count	Never	
		Previous Next	
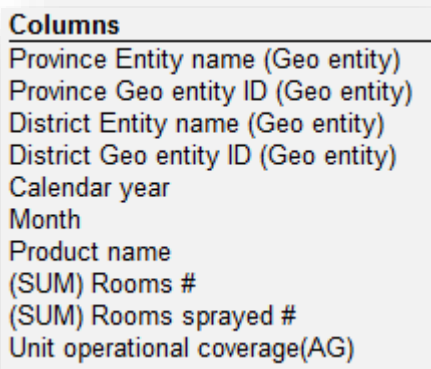
The main advantage of a persisted result is performance. Some queries can run for a long time because they access several linked tables. An IRS query can access hundreds of thousands of rows and for each row complex calculations might be performed to ultimately provide a query result which in turn might contain thousands of rows, but if this result is stored in a data set it becomes a single indexed table. Any operation using this table will be fast.

The BIRT report writer can continue to use views (still created by saving the query) and can additionally use the data sets.

Kaleidoscopes and the DHSI2 interface only use data sets.

Exercise

✓ Create the following query in the IRS query builder:

Columns
Province Entity name (Geo entity)
Province Geo entity ID (Geo entity)
District Entity name (Geo entity)
District Geo entity ID (Geo entity)
Calendar year
Month
Product name
(SUM) Rooms #
(SUM) Rooms sprayed #
Unit operational coverage(AG)

✓ Persist the query with the name 'IRS'

Note that this query takes quite a long time to run because it involves a dozen tables, does a fair amount of aggregation and performs complex calculations to determine the application ratio for each aggregated row in the query result.

Important note: When a persisted query is edited it is sometimes necessary to delete the data set and create a new one. Examples are:

- Removing columns
- Changing the aggregation function on a column

This has serious repercussions on Kaleidoscopes and reports that are already using the data set. If the data set is destroyed, it is also automatically removed from the Kaleidoscope and any layers that were using data elements from the data set will also be removed. Reports will be corrupt.

When creating a persisted query for Kaleidoscopes and reports, it can be useful to design a single query containing all data elements in the query builder, at the lowest useful geographic level, without aggregation, rather than multiple specific queries.

4. Kaleidoscopes

4.1. Creating a Kaleidoscope

The DDMS landing page provides access to the Kaleidoscopes, but you can also access it from the menu.

✓ Open *GIS > Kaleidoscopes*
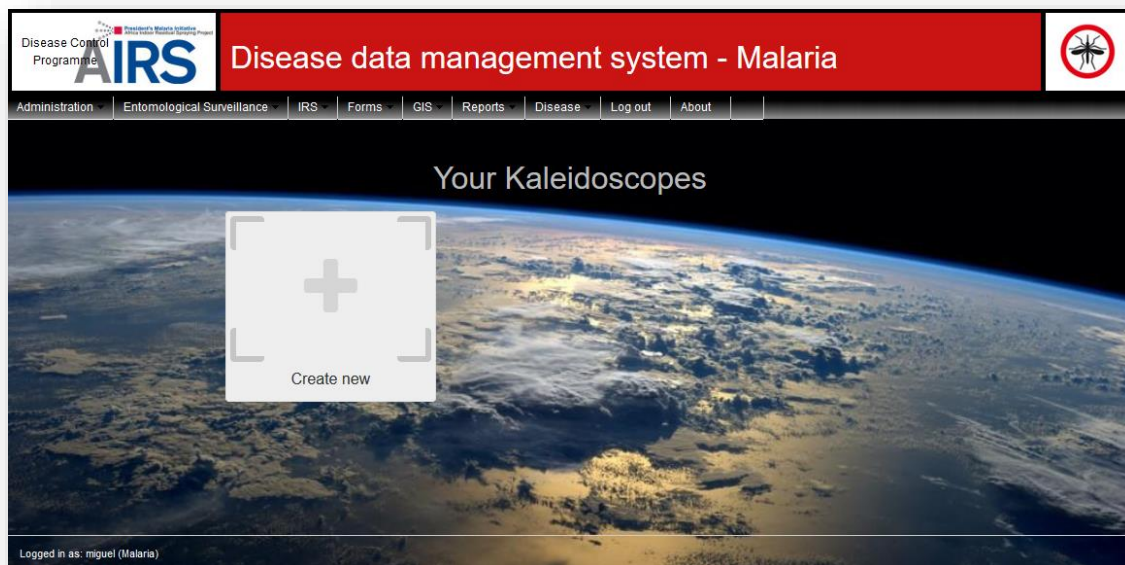
In a fresh install or patched system there are no Kaleidoscopes. The Kaleidoscopes will replace the previous DDMS maps over time, but there is no upgrade path from an existing map to a Kaleidoscope.

Because Kaleidoscopes are intended to replace the old mapping functionality, a fresh install of 1.06 will not have the *Generate maps* menu option anymore. Nevertheless, the old technology still works and existing maps are still accessible, in particular to existing BIRT reports that use maps. The deprecated map cycle job is also still supported in the Scheduler. If you need to access the deprecated map functionality and you don't find it in the menu, you can add it back using the standard menu configuration tool.

✓ Click on the *Create new* thumbnail on the Kaleidoscopes page

When you create a new Kaleidoscope, you are only defining which data sets it will use. The detailed configuration of the Kaleidoscope is a subsequent step.

A pop-up allows you to name the Kaleidoscope (required) and provide a description (optional).

✓ Label your Kaleidoscope 'IRS impact on vector abundance'
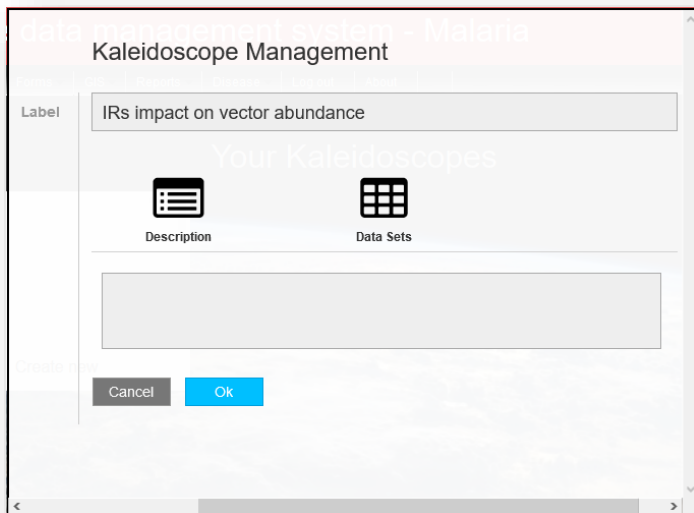
✓ Click on the *Data Sets* icon
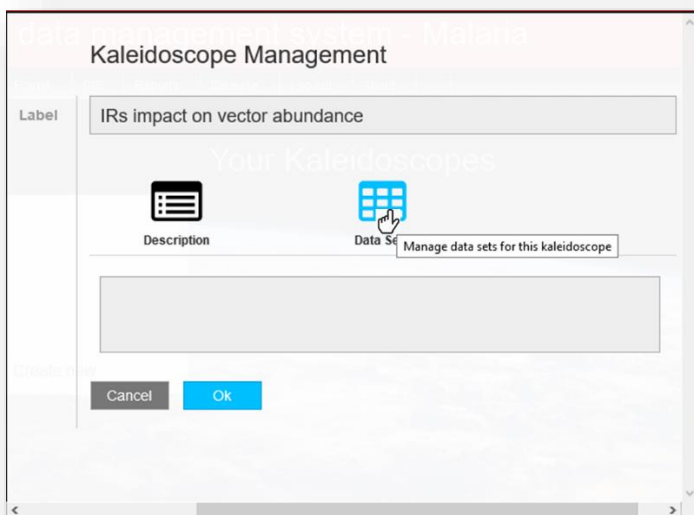
All saved data sets are available here for selection – not only persisted queries but also forms and ETL generated data sets.

You can select the whole data set by ticking the checkbox or select individual data elements by expanding the data set and ticking the columns you want to use.

✓ Expand *IRS* by clicking on the grey arrow to the left of the *IRS* label
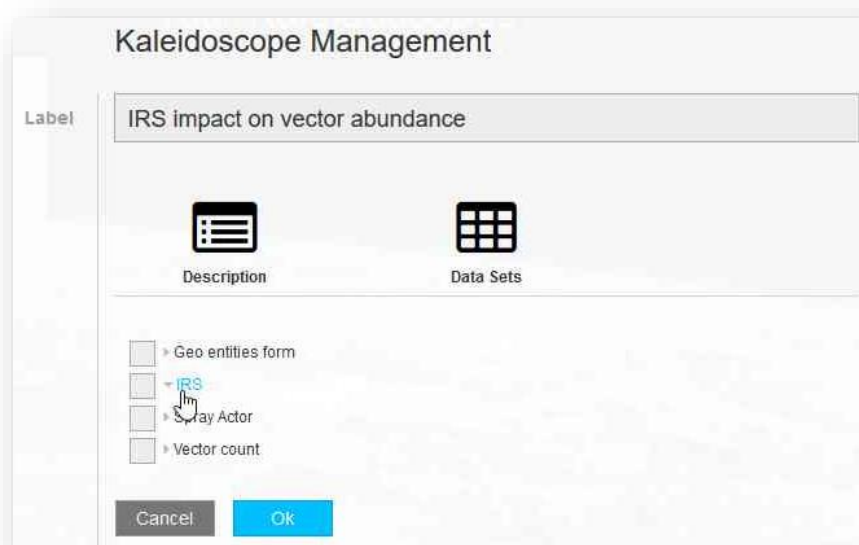
✓ Tick the *IRS* check box

All data elements are automatically selected but you can remove data elements from the Kaleidoscope by clicking in the individual check boxes.

✓ Tick the *Vector count* data set as well

✓ Click on the *Ok* button

The Kaleidoscope is created:
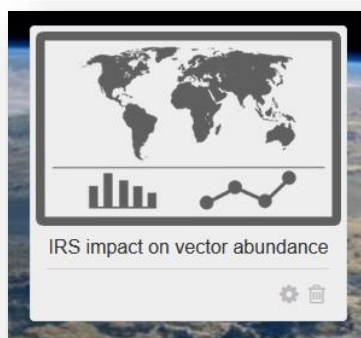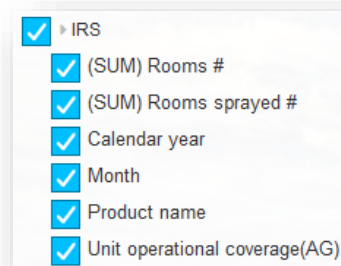
4.2. The Kaleidoscope Layout

A Kaleidoscope is a shared interactive map with optionally a linked report. The map uses the full page and all tools are accessed in collapsible panes.

✓ Open the Kaleidoscope by clicking on the thumbnail

The menu is removed to provide more real estate for the map.   A Kaleidoscope is always opened in its own tab, so unless you close the other DDMS tabs in the browser, you can always get back to the menu by clicking on another DDMS tab. If you have closed all tabs by mistake, you can get back to the landing page by using Kaleidoscope menu icon and selecting Home.

✓ Hover over the menu icon at the top right corner of the Kaleidoscope and click on *Home*
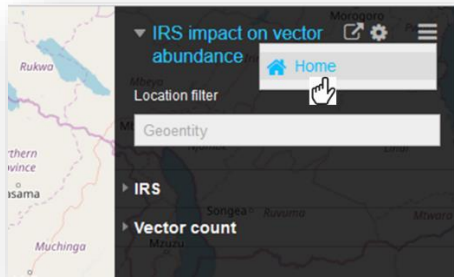
✓ Open the Kaleidoscope again

There are three collapsible panes:

1. The data set pane (by default open)
2. The map layer pane (by default open)
3. The report pane (by default closed)
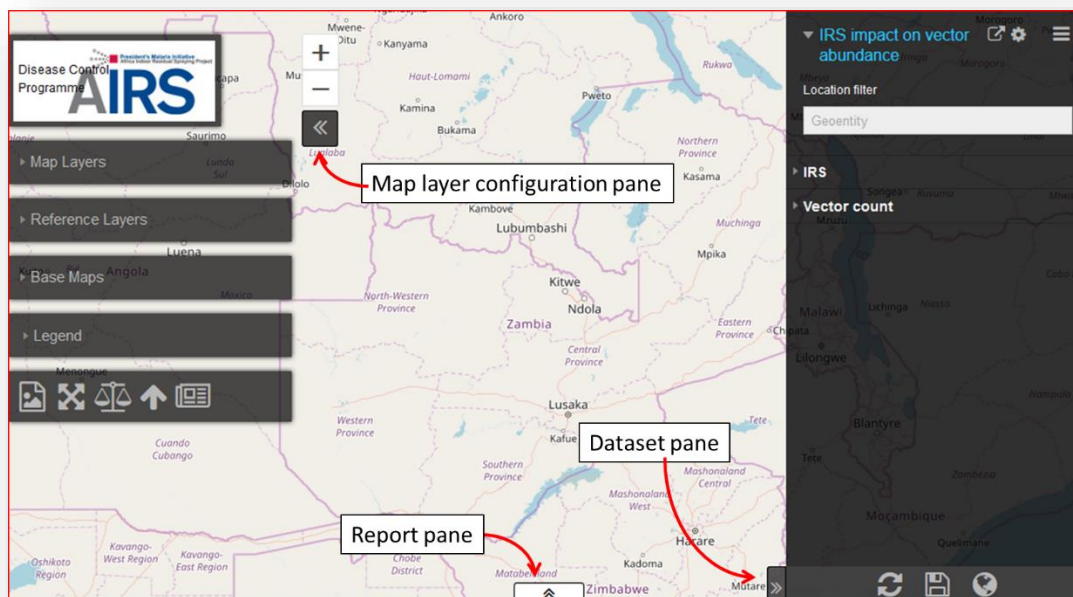
4.3.   The Data set Pane

You can change the initial configuration of the Kaleidoscope by clicking on the *Kaleidoscope options* icon at the top right. This is if you want to change the Kaleidoscope name or change the selected data sets or data elements.

Remember that the Kaleidoscope is using the data set, and not the original data (if the data set originates from a persisted query). Any filters or aggregation that you configure apply to the data set. Disaggregation of the data set is not possible. If you have saved the query aggregated by province, you will not be able to drill down below province.
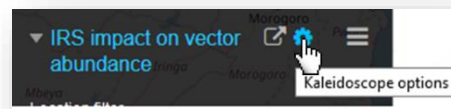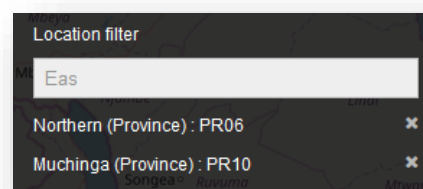
If you want to limit the map to specific geographical boundaries, you can set a location filter. The location filter uses the geo tree. It is possible to filter on geography that is not included in the data set. This is because the data set could contain different geo entities each time it is saved. You can filter on one or several geo entities.

You can expand or collapse the data set by clicking on the arrow to the left of the data set name.
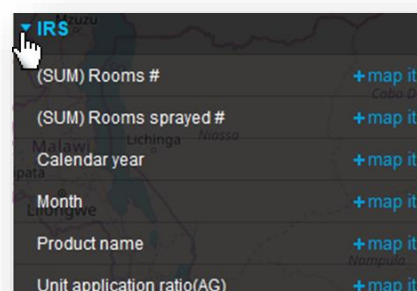
Each data element of the included data sets can be mapped in a layer. A layer can only reference one data element (except for *Bubble* layer types).

You can configure a filter on each data element. This is similar to filters in the query builder but there are a few useful differences:

- Using filters in the query means that these filters apply to any Kaleidoscope using the resulting data set, whilst multiple Kaleidoscopes can use the same data set with different saved filters

- Query filters also apply to any user accessing the Kaleidoscope, whilst users can set filters on a Kaleidoscope and save the setting for themselves only

- If you change the filter on the query, you must save the persisted query before the result can be used in Kaleidoscopes

Taking account of all of the above, it is often useful to persist a query for use in Kaleidoscopes at the lowest geographical level possible and without filters. One drawback of Kaleidoscope filters is that period groupings such as *Calendar year* and *Epi week* or simply

recognised as text in the data set and so you cannot put intelligent period filters on them. You can get around this by putting a date in your persisted query.

- ✓ Open the IRS query builder in a different tab and add *Spray date* to the query. Then save it.
- ✓ Go back to the Kaleidoscope tab, open the Kaleidoscope settings and add the *Spray date* to the Kaleidoscope
- ✓ Check the filter options on the *Spray date*

To set a filter on a data element, click on the data element.

- ✓ Click on *Product name* in the *IRS* data set
- ✓ Type 'fi' in the search box
- ✓ Select FICAM VC
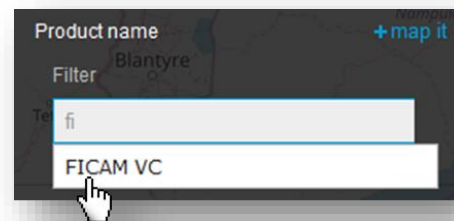
The data set is now filtered on FICAM VC and any map layers you add will only apply to data rows containing FICAM VC.

Filters are not automatically applied to rendered map layers. To refresh the map using the configured filters, click on the Refresh icon at the bottom of the pane.
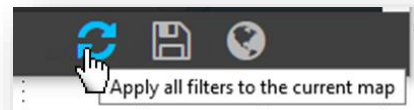
Filters are not automatically saved to the Kaleidoscope. If you would like to save the filter for yourself only, click on the Save icon at the bottom of the pane.
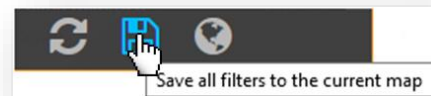
If you want to save the filters to the Kaleidoscope for all users, click on the Globe icon at the bottom of the pane.
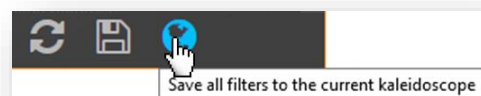
Only the person who created the globally saved filter can remove the filter by clicking on the x.
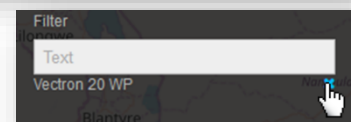
To create a thematic map layer based on a specific data element, click on next to the data element.

4.4. Layers

There are three types of layers:

1. Base maps
2. Reference layers
3. Map layers

Base maps

You have the choice to use one or no base maps. The options for base maps are *Open Street Map* or *Bing Satellite*. It is necessary to be online to use either of these, but for environments where the DDMS server has limited connectivity, you can also configure *Offline Open Street Maps*.

Open Street Map

This is an open source repository maintained by the user community and tends to be detailed even in areas with low population density. These maps are vector based tiles that are much smaller than satellite images. Maps typically update quickly when zooming or panning, even with low bandwidth. The *Open Street Map* base map can also be used when generating maps to be used in reports.

Bing Satellite

This is a free repository of satellite images.

Open Street Map (offline)

This uses a downloaded repository of Open Street Map tiles. It is always available, even if the server has no connection to the internet. When using *Open Street Map (offline)* the user only has access to the OSM tiles that have been downloaded by the administrator. Use of *Open Street Map (offline)* is detailed later in the course.

Reference Layers

Reference layers are map layers that are not based on a thematic variable (data element). These provide a reference background of e.g. the country or provinces. You can select the geo level you want to use as reference layer, the fill colour of the polygons and the border. The configuration of reference layers is a simplified version of map layers and is not discussed in any detail in this course.

Reference layers also drill down and up automatically as you drill down and up on thematic layers.

Map Layers

Map layers are the thematic layers of the map. You can create as many as you like. Each map layer uses a single data element as thematic variable (except buuble type layers). Map layers are discussed in detail later in this course.
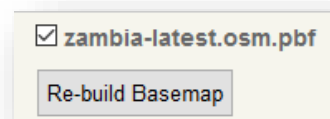
4.5. Open Street Map (offline)

DDMS 1.06 includes the ability to use downloaded OSM tiles. In a fresh installation of the DDMS this is included under the GIS menu. If it is not in the menu, it can be added using the standard menu configuration tools. The functional item is called *Offline basemap management*.

- ✓ If *Offline basemap management* is not in the GIS menu, add it there
- ✓ Open the *Offline basemap management* page

The page contains a lot of explanation and a single button. Essentially you need to download pbf files from geofabrik.de, put them in C:\MDSS\basemaps. Then you can select the files you want to add to the DDMS offline OSM basemaps.

Note that every time you rebuild the offline basemap, the previous offline basemap is deleted. This means that if you already had Zambia as your offline basemap and you want to add Zimbabwe, you need to have both pbf files in C:\MDSS\basemaps and select them both before re-building the basemap.

- ✓ Click on the http://download.geofabrik.de/ link in the page
- ✓ Browse to Zambia and download the pbf file
- ✓ Move the file into C:\MDSS\basemaps
- ✓ Refresh the *Offline basemap management* page.
  You will see the pbf file with a check box next to it
- ✓ Tick the check box and click on the *Re-build Basemap* button

Building the basemap takes some time. Once the process is completed the *Open Street Map (offline)* basemap will be available in Kaleidoscopes and report map images. It is not necessarily identical to the online OSM basemap.

4.6. Map Layers

Map layers are added by clicking on [+map it] next to the data element you want to map.

- ✓ Open the IRS impact on vector abundance Kaleidoscope
- ✓ Expand the *IRS* data set
- ✓ Make sure all filters are removed
- ✓ Click on [+map it] next to (SUM) Rooms sprayed #

A full page pop-up allows you to configure the layer.
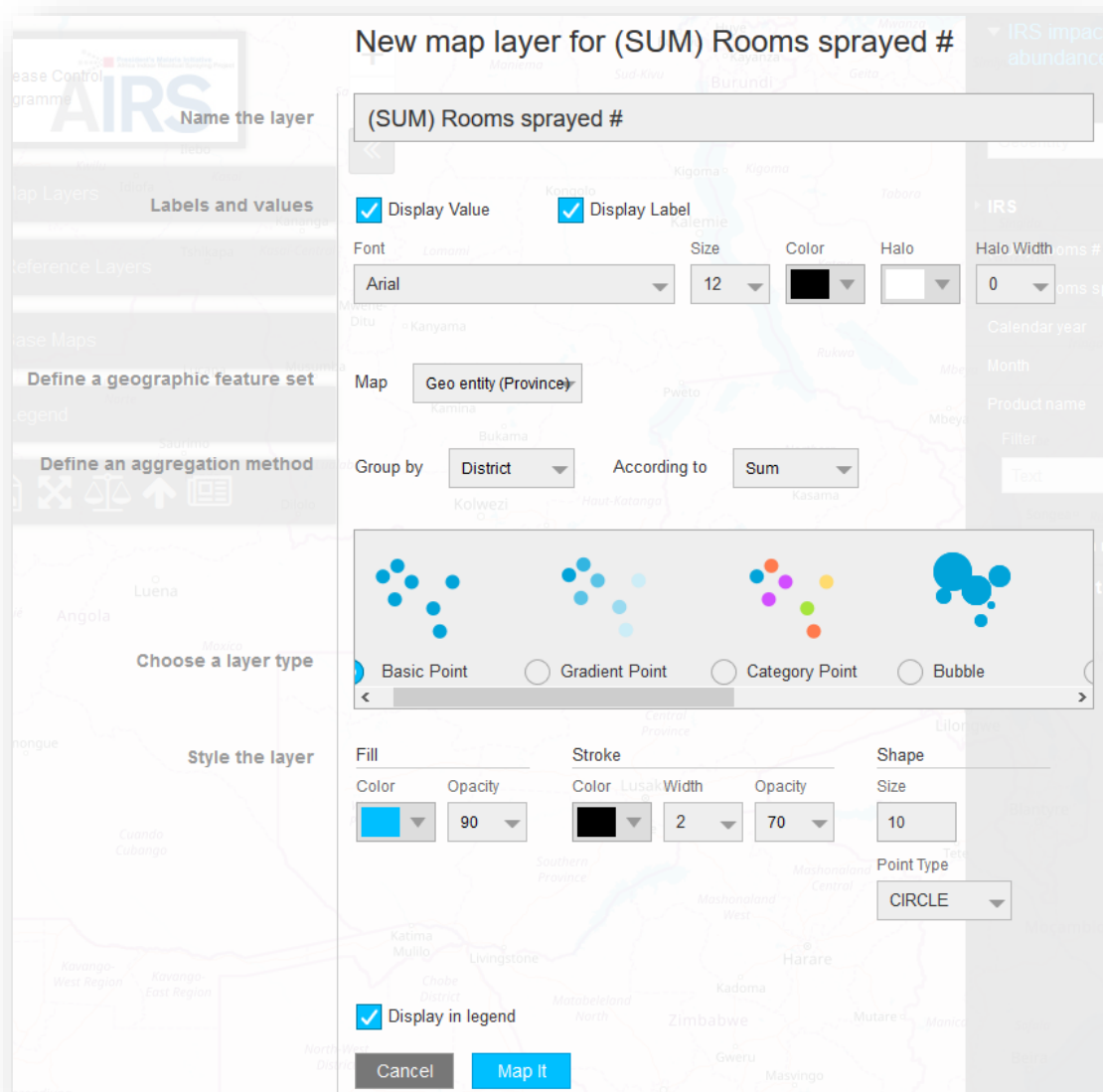
This is what the different settings do:

o *Name the layer*: the localisable display label shown in the Kaleidoscope. The default is the data element name

✓ Change this name to 'Number of rooms sprayed'
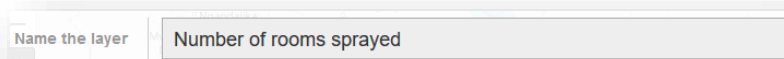
o *Display Value check box:* Will show the aggregated value of the thematic variable in the map layer
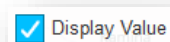
- o *Display Label* check box: Will show the geo entity name in the map layer
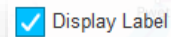
✓ Tick both of these

- o *Font*, *Size*, *Color*, *Halo* and *Halo Width* are attributes of the displayed label and value

- o *Define a geographic feature set*: This is only relevant if your data set has more than one geo column defined. It tells the map layer which column to use. A useful example is in Case Surveillance, if your persisted query contains both the source location and the health facility. You can use the same data set and even data element to create two different layers using different geo columns.
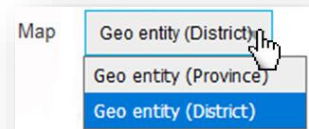
✓ Since our query contains both *District* and *Province*, you will see both these options in the select list. Select *District*. If you select *Province*, you will not be able to drill down lower than province.

- o *Group by*: Defines the aggregation level that the layer is rendered at initially. It is useful to select the highest (least granular) geo level here. Once rendered, it is possible to drill down in the map. Rolling up is only possible after first drilling down.
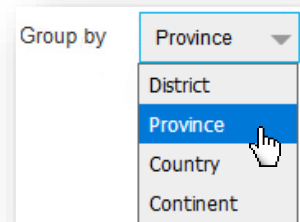
✓ Select Province for *Group by*

- o *According to:* Select the aggregation function that must be applied to the data element when aggregated in the map layer. This is typically the same aggregation function you selected (if any) in the persisted query. A new feature is that textual data elements can also be aggregated on *Majority* and *Minority*. An example is *Product name* in the IRS data set. If you aggregate using *Majority*, it will show the *Product name* that appears most on the data set rows for each geo entity.
  Note that *Min*, *Max* and *Sum* aggregate well even if the persisted query is already aggregated with the same function, but *Avg* is only an approximation if the persisted query is also aggregated.
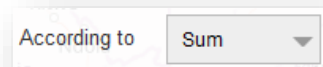
- o *Choose a layer type*: Essentially you can choose between using the centroid of a geo entity (point) or the polygon, if polygon data are available. Then, for each of these, you can choose Basic, Gradient or Category. For points, you also have Bubble, which

is particularly interesting because you can add a second attribute to it. To select a layer type, click in the corresponding radio button.

- ❖ *Basic Point*: Select a *Color, Size* and *Type*. Each geo entity will be represented by a point in the layer. *Stroke* is the outline of the point.
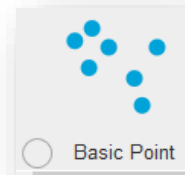
- ❖ *Gradient Point*: All points will have the same size, but their colour will vary according to the aggregated thematic value. You can select the number of *Buckets* and define the *Min* and *Max Fill* colour. Equal sized buckets will be created automatically and the colour of each bucket will follow a gradient path from the *Min* colour to the *Max* colour. It is usually best to similar colours for Min and Max, such as light blue and dark blue. A gradient path between green and red can produce ugly results.
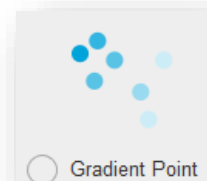
- ❖ *Category Point*: This differs for textual data and numeric data. For textual data elements you can define up to 5 exact values and set their chosen colour. For numeric data elements you can define lower and upper thresholds for each category (1.06.0011). The points will all have the same size and type, but their colour will differ according to which category is represented.
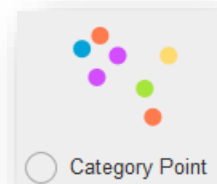
❖ *Bubble*: This layer type can have two attributes defined by two different data elements. The first is the size of the bubble, and the second is the colour of the bubble.

You can set the *Fill Color* and *Min*/*Max* point size of the bubbles. By default the bubble size will be continuous, but if you prefer distinct sizes you can unselect *Continuous Size*. This will give you the option to define an exact number of buckets.

To add a secondary attribute, select a data element from *Secondary attribute* select list. You can only choose from data elements in the same data set. You will get the option to define categories and the colour of the bubble. You must also define the *Aggregation* for this data element.

✓ Select *Bubble* as the *Layer type*.
Set the *Min* value to 10 and the *Max* value to 100.
Select *Product name* as the *Secondary attribute*.
Select *Majority* as the *Aggregation* function.
Create the 'Actellic 300CS' and 'FICAM VC' categories as shown here.
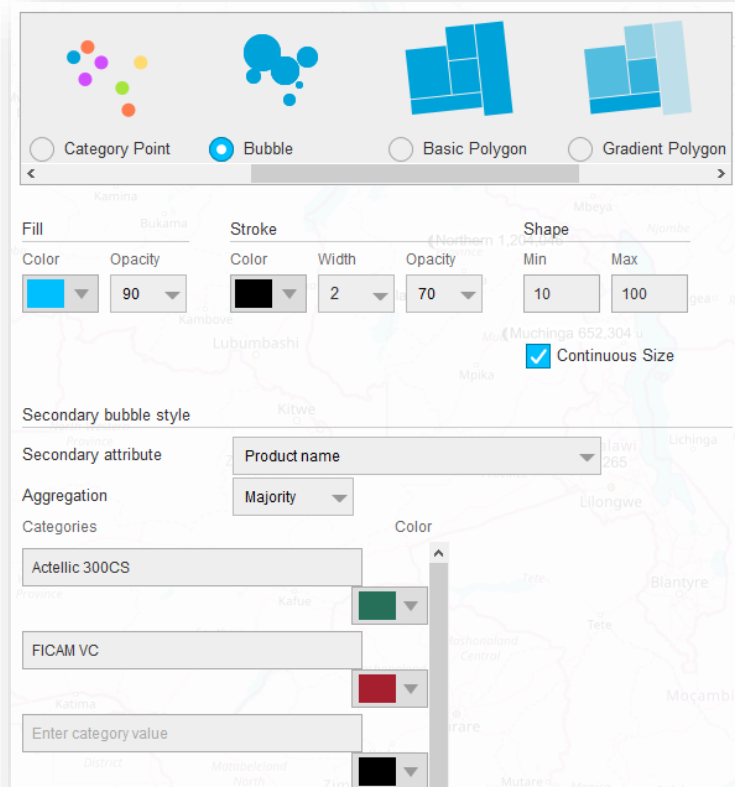
❖ You can also select *Basic Polygon*, *Gradient Polygon* and *Category Polygon*. These have the exact same attributes as their Point counterparts.

o   *Display in legend*: Each layer can have a legend. By default, a legend will be created. Sometimes a legend is not required and can even be disturbing. Examples are Reference Layers and continuous bubbles. To remove the legend for the layer, un-tick the *Display in legend* check box.
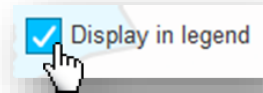
✓   Click on   Map It   . You should see the following map:
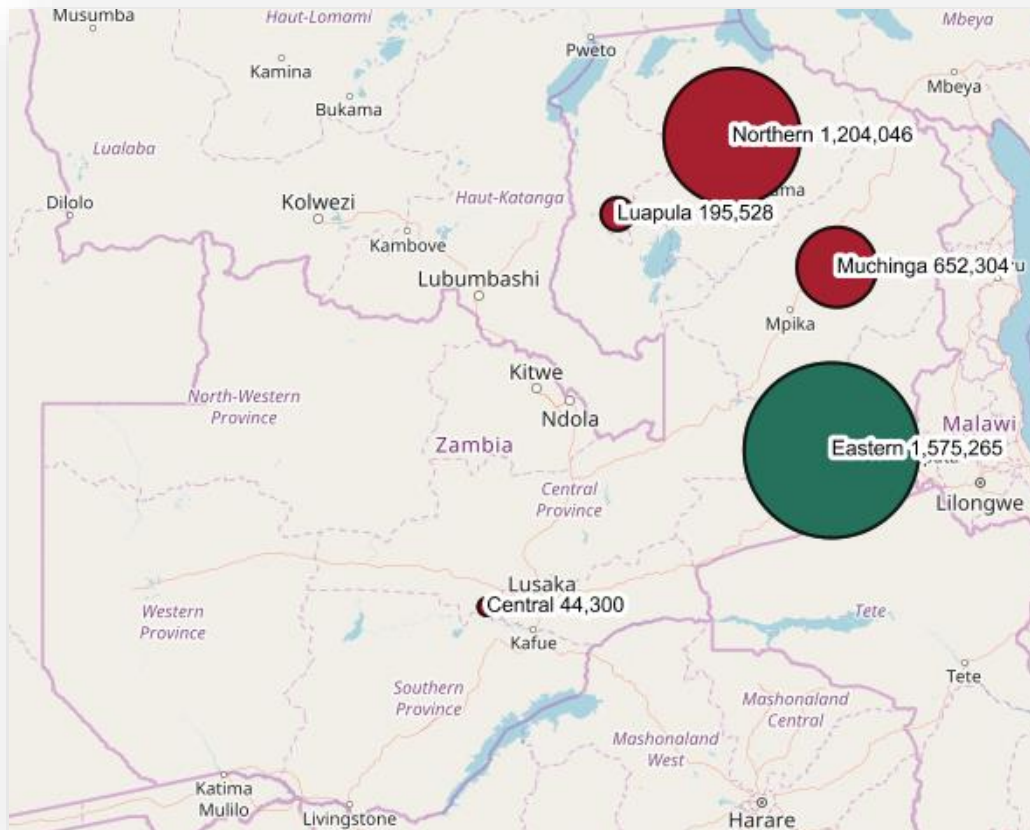
Exercise: Enhance the map

✓   Create a reference layer showing the provinces as basic polygons in grey

✓   Create a map layer *using (SUM) Total* from the *Vector count* data set as the base map. Use a gradient polygon from light blue to dark blue

✓   Use the *Bing Satellite* base map

Your map should look like this:
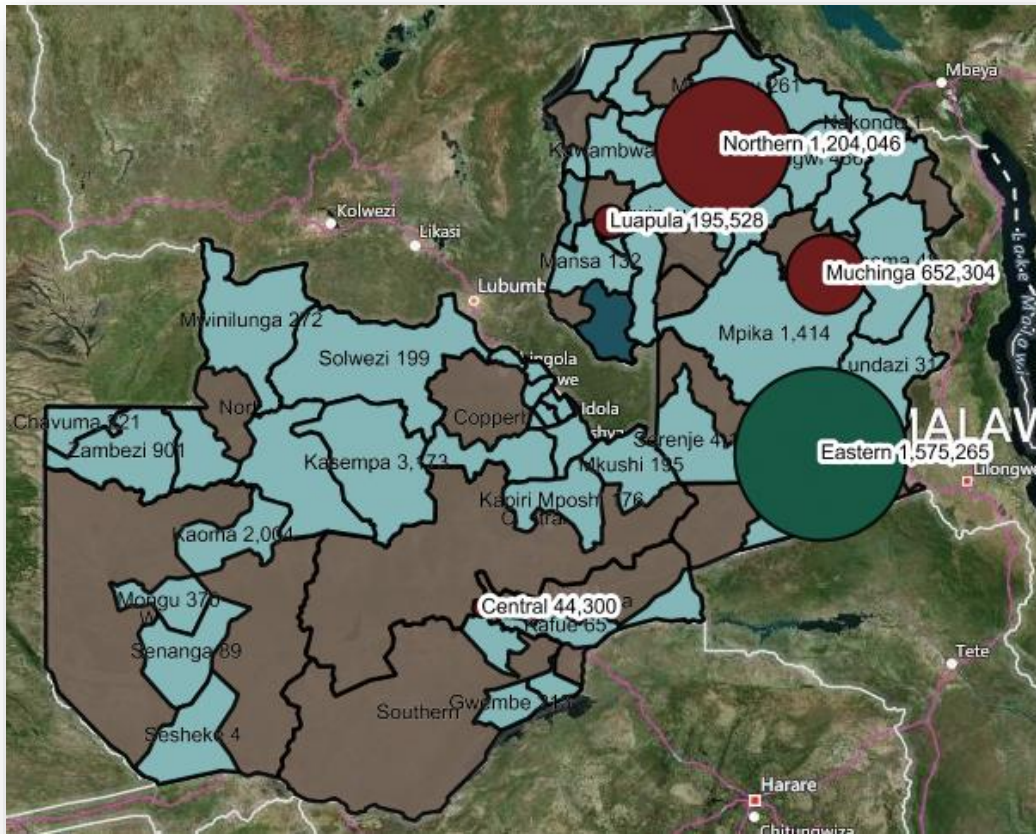
4.7.   Other Map Layer Pane Options

The map layer configuration pane can be collapsed by clicking on the double
arrows pointing left.

Each section in the pane can be expanded or collapsed by clicking on the
arrow to the left of the section headers.

Map layers can be switched on or off by ticking or un-ticking them in the
Map Layers section.

The order of the layers can be changed by dragging one above the other. The reference
layers will always be under the map layers.

The legends, which by default show in the legend section of the pane, can be detached or
re-attached by double-clicking the legend. It can be repositioned by dragging it.

Each layer can be deleted, edited or exported as an xlsx file by clicking on
the appropriate icon.

You can export the map as a PNG image file. Note that all layers are exported in one image, whether or not they are ticked in the *Map Layers* section.

You can zoom in and out by using the scroll button on the mouse. If you want to zoom back to the full extent of the map, you can click on the following icon:

You can add the map scale and north arrow by clicking on the appropriate icon. They can be repositioned by dragging them.
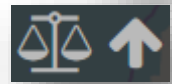
The Kaleidoscope also supports rendering individual geo entities in the map as separate images in the database, to be used by reports. The functionality is identical to the functionality that was available in the mapping module of 1.05 but it is quite a bit faster.
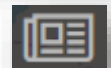
Note: To render map sections as database images, the process uses the globally saved state of the map. If you add the scale or north arrow or position a legend, you must click on the globe icon at the bottom of the data set pane to include these in the map sections.

4.8. Map Interactivity

Kaleidoscopes are interactive. You can click on a polygon or a bubble and a pop-up will provide more information about the geo entity and the data element.

For any layer type you will have the option to zoom to the feature (at the bottom of the pop-up). This simply zooms the map to the full extent of the entity you had clicked on. The name of the geo entity is in the pop-up.
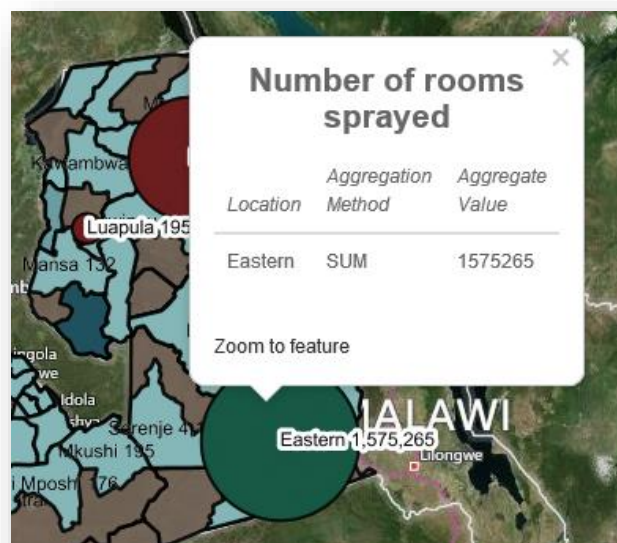
If you click on a polygon, you will also get the chance to drill down. Every layer renders at one level lower if it is possible and only shows the geo entity that was drilled-down on. If there is a layer that has no data within the drilled-down polygon, an error message will pop up and the layer will be removed.

It is not possible to drill-down in our map because the polygon layer is at district level and that is the same level as the data set, so there is no information available to drill-down.

To demonstrate drill-down, change the group by of *(SUM) Total* to *Province*.

✓ Open the *(SUM) Total* map layer for editing and change *Group b*y to *Province*. Map it.

Now you can click on *Luapula* province and drill down:
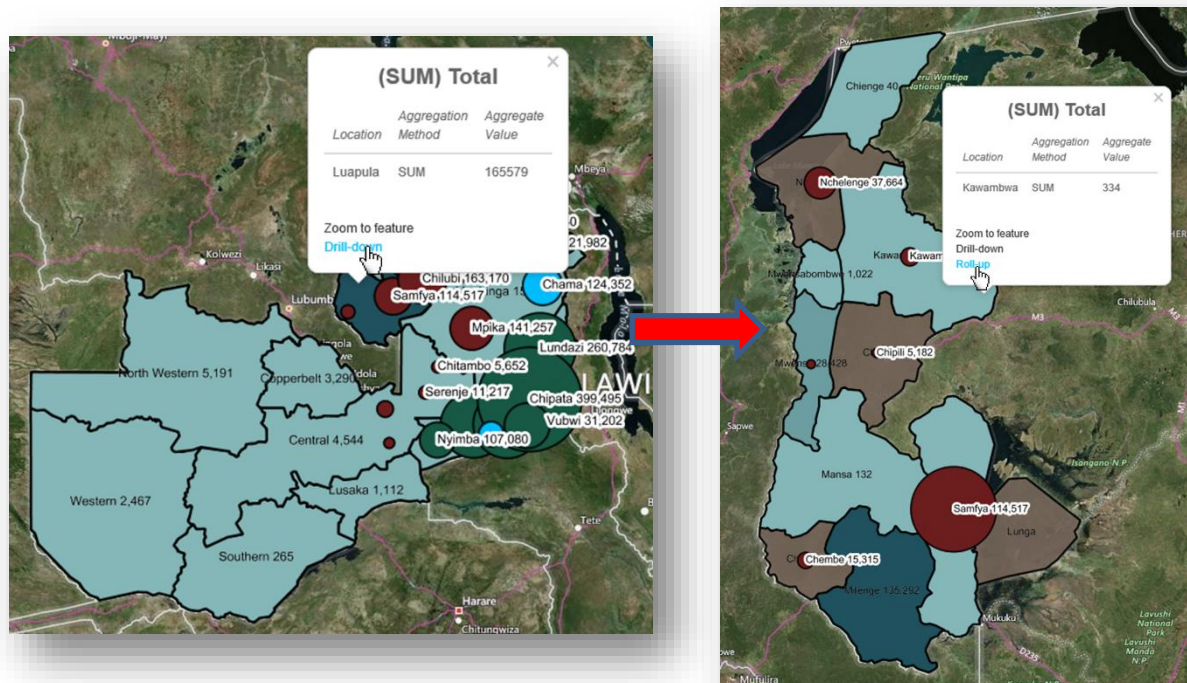
By clicking on any polygon in the drilled-down map you will now also get the option to Roll-up again, to the previous map.

Because the DDMS supports a meshed geo level configuration, it is not always predetermined which route the drill-down should take. If the universal tree branches, a question will pop-up asking which route to take. This choice will be remembered if you drill down further so the same question will not pop-up if a different layer faces the same choice. When rolling-up, the same path will be followed.

5. Creating an Eclipse Map Linked to a Kaleidoscope

It is possible to create a report that is linked to the Kaleidoscope. Such a report will be rebuilt every time the user clicks on a geo entity. The behaviour is defined in the report.

This functionality is only available when using the latest version of BIRT, which is now called Eclipse. Eclipse is installed with a 1.06 install or patch. BIRT is still included for backward compatibility reasons.
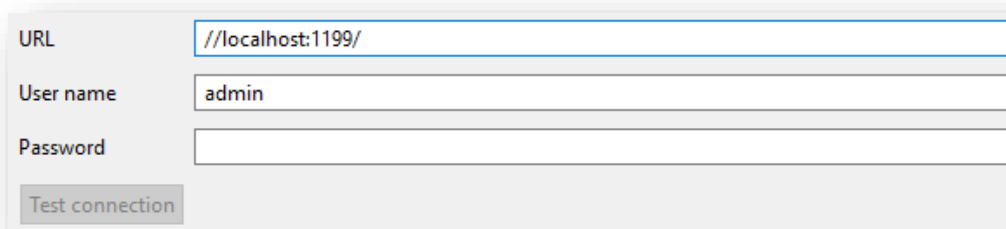
✓ Open the Eclipse report writer from the Windows menu (DDMS group)

5.1. Kaleidoscope Data Sets

For an Eclipse report it is easiest to use a template. It requires a custom data source and some parameters. We will configure the data source manually and copy and paste the parameters from an existing report.

✓ Open a new report
✓ Add a new data source
✓ Select the *DDMS Data Source*
✓ Name the data source 'Kaleidoscope'
✓ Click on *Next>*

The next step has a predefined URL, but this URL depends on your installation.

The number you see in the URL is the RMI port of the installation. This can be found in the common.properties file in your *C:\MDSS\tomcat\webapps\Zambia\WEB-INF\classes* folder.

The user name and password are your DDMS user name and password. You can use train/train.

- ✓ Complete the input fields and click on *Test connection*. You should get a success notification.

5.2. Report Parameters

There are some parameters that are required to create the connection between the report and the Kaleidoscope.

- ✓ Open the report design provided by the trainer and copy the Report Parameters. You can select them all at once by clicking on the first and shift-clicking on the last.
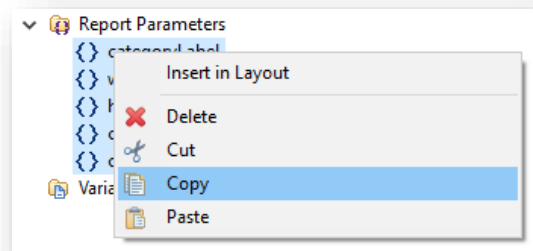
- ✓ Paste them in the *Report Parameters* of your new report

5.3. Kaleidoscope Data Sets

Next we need a data set.

- ✓ Add a new Data Set. Name it 'Vector' and proceed to *Next>*

Under *Select type* you will see all the data sets that are available in the DDMS.

- ✓ Select *Vector count*

The *Geometry aggregation* allows you to select how you want the report to react when a geo entity is clicked on in the Kaleidoscope. There are a lot of options you can experiment with but the most commonly used geometry aggregation will be:

- *Children*: the report will use the geo entities one level down from the clicked geo entity. This is useful to give you a visual breakdown of what is happening one level down, without needing to drill down in the map
- *None*: Only show data for the clicked geo entity. This can be used to give a full tabular overview of all the data elements in the dataset, whilst only showing one or two in the map. It can also be useful to give a breakdown over time, which is impossible in a single map.
- *Type*: show all data at the same geo level. This is essentially the same as *None*, except that *None* is filtered on the clicked geo entity.

When a geo entity is clicked in the Kaleidoscope, the report is re-run. The filtered data set is sent to the report, and the geo columns in the data set will contain the geo entities as determined by the *Geometry aggregation*. The data will not be aggregated. This must happen in the Eclipse report.

Example: the filtered data set has 1000 rows. 500 rows have a geo entity that is located in Muchinga province. These could be district, villages, collection sites… You have set the *Geometry aggregation* to Children. When you click on Muchinga in the map, the data set that is returned to Eclipse contains 500 rows and the geo Entity will be the district that the original geo entities were located in.

- ✓ Select *Children* and click on *Finish*

The *Output columns* names can be cryptic but the *Display Name* is more meaningful. Of special importance are the following:

- *Geo entity* and *Geo entity ID*: note that these columns are not level specific. These columns contain the name and geo ID of the geo entities that are returned by the Kaleidoscope. If you clicked on a province and have selected *Children* as *Geometry aggregation* these columns will contain the name and ID of the districts located in the clicked province.
- *Category*: This is the name of the geo entity that was clicked in the Kaleidoscope

You can preview the results in the data set but the *Category* cannot be changed. You can change the *Geometry aggregation* by selecting *Dashboard* in the data set.

- ✓ Set the *Geometry aggregation* to *Grandkids* and preview the result. Set it back to *Children*.

One last thing that must happen before you finalise the data set, is to link data set parameter *Context* to the Report Parameter *Context*.

✓ Select *Parameters* in the data set and double-click on *context*

✓ In the *Linked To Report Parameter* select list, select *context* and click on *OK*

5.4. Kaleidoscope Report

You can now create a report as before, but the data set will be updated every time you click on a geo entity in the Kaleidoscope. A report using a DDMS data source is useless if not linked to a Kaleidoscope, because its data sets will not be updated. We will link the report to a Kaleidoscope in the next chapter.

To see the results of a linked report, we will create a basic chart in our report:

✓ Drag a chart into the report and leave the chart type as *Bar*

✓ Next configure the data as shown below:
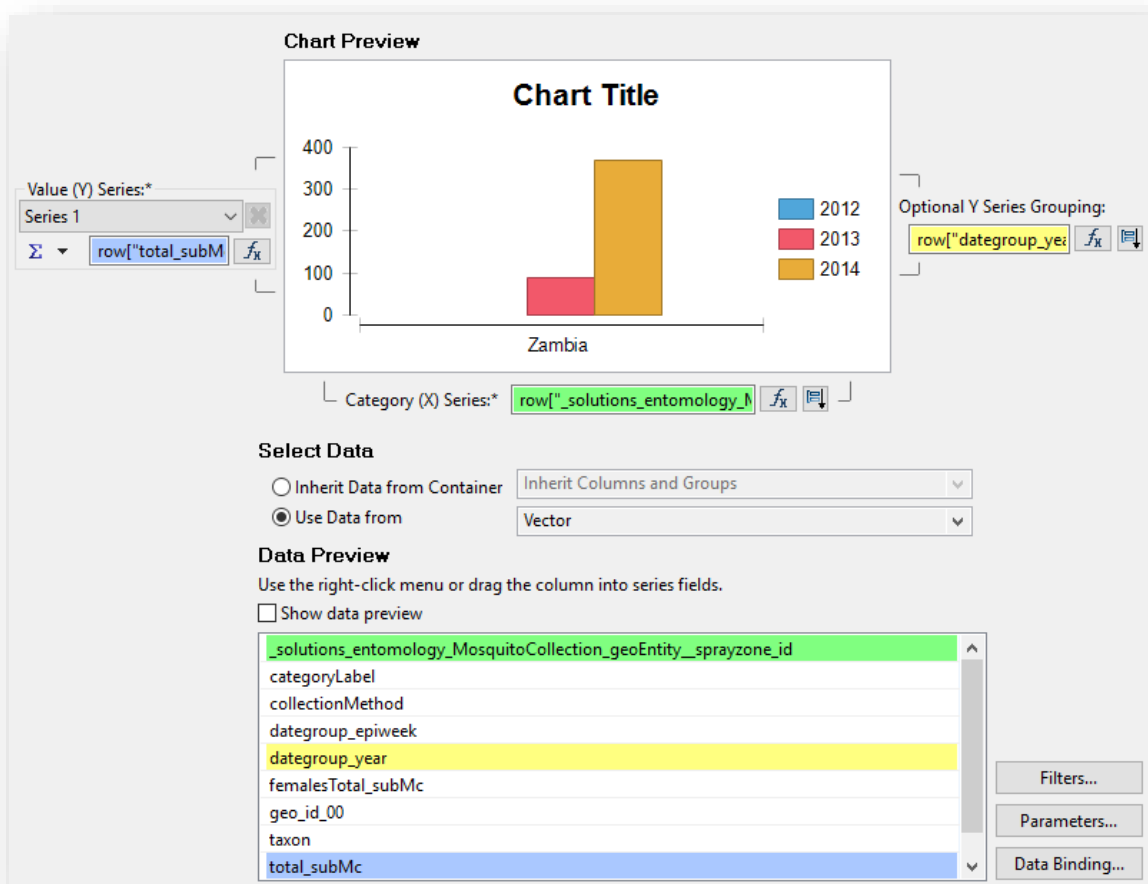
Note that by default the values are summed:
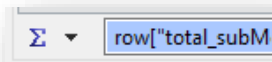
This is usually correct. Aggregation is typically required because the data set that is returned by the Kaleidoscope has substituted the geo entities without aggregating them.

✓ *Finish* the report and size it to the full width of the page

You can preview the result to see if there are any errors in the report but since the preview is governed by Eclipse and not the Kaleidoscope, you cannot see what happens when you click in the Kaleidoscope until you link this report to the Kaleidoscope in the DDMS.

✓ The report is automatically saved every time you preview it. If you have not previewed it, save it now by clicking on the *Save* icon.

5.5.   Combining a DDMS Data Source with a JDBC Data Source

It is possible to add a classic JDBC data source to a Kaleidoscope report using saved query views or map images. They can still be linked by using the Category value from the DDMS data source's data set(s) as filter on the other data. As such, you could for instance show a row of thumbnail maps in the report, showing specific data, year by year, for the clicked geo entity. This would require all the possible maps to be stored as images in the database. It would not take long to render such a report, since displaying images contained in a table is quick.

Use of saved query views could slow down the rendering of the report significantly, as the queries are run each time you click in the map.

6.   The Kaleidoscope Report Pane

6.1.   Managing a report in a Kaleidoscope

To upload a report to a Kaleidoscope, you must open the Kaleidoscope.

✓ Check that you still have the *IRS impact on vector abundance* Kaleidoscope open in a tab. If not, open it.

The Kaleidoscope report pane is collapsed by default. You can expand it by clicking on the double up-arrow at the bottom of the screen.

This will bring the Report pane up halfway the screen. Now you can right-click in the pane and you will get a context menu.

✓ Click on *Upload report* and browse for the report you have just saved. Click on *Submit*.

Nothing will happen until you refresh the page or refresh the Kaleidoscope by clicking on the *Refresh* icon at the bottom of the Data set pane.

The latter is the better option as it will not collapse the report pane.

Now you see the report, but since you have not clicked on a geo entity in the map, it simply shows Zambia.

✓ Click on *Luapula* province
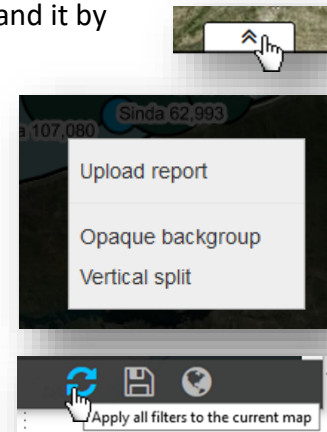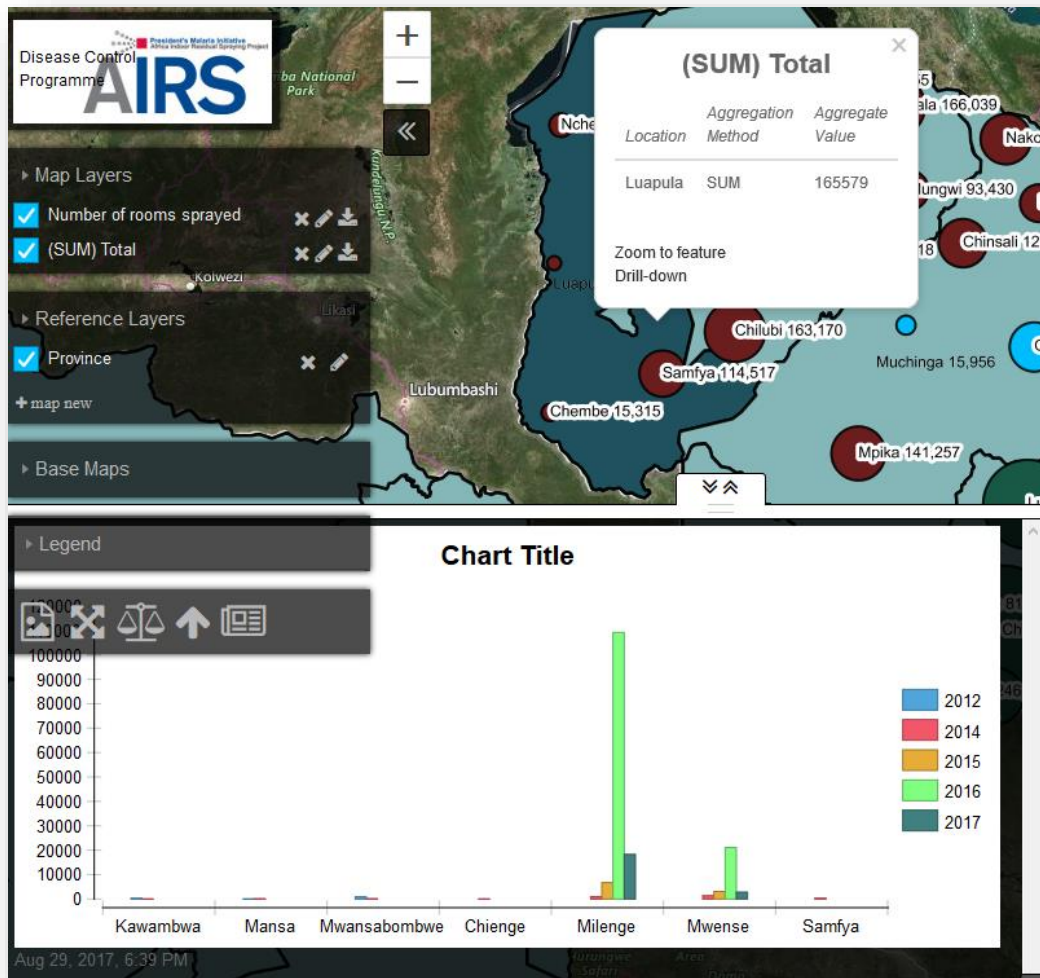
The chart now shows a breakdown of collected mosquitoes by district, showing only the Luapula districts. If you are seeing something else, it is possible that your *Geometry aggregation* in the Eclipse data set is something other than *Children*.

Fixing report layouts is easy in 1.06. Change the report design, save it, re-upload it and refresh.

: Reports can be scrolled in the report pane. Avoid pagination in your report and select Auto Layout in the report's general properties. Auto layout does not affect charts but will resize tables according to your DDMS window.   Pagination can be useful if you have different report sections that you want to access by clicking on report items.

Upload report
Remove report
Export report

Export DOCX
Export XLSX
Export PDF

Opaque backgroup
Vertical split

You can remove or export the report by using the context menu. Once you have uploaded a report, there are more options available.

The *Export report* option is very useful. It will export the Eclipse report design, so that even if you didn't have the original report, you can export and edit it. The reports are included in a DDMS backup.
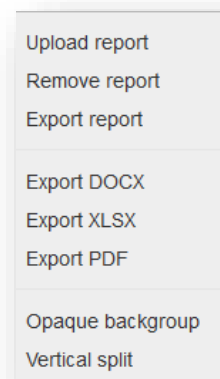
6.2.   Kaleidoscope Report Visualisation Settings

Now that a report has been uploaded, you can play with the report visualisation settings.

The double-up and double-down arrows change the size of the pane from half to full screen and back again. You can also drag the pane to any size you want.
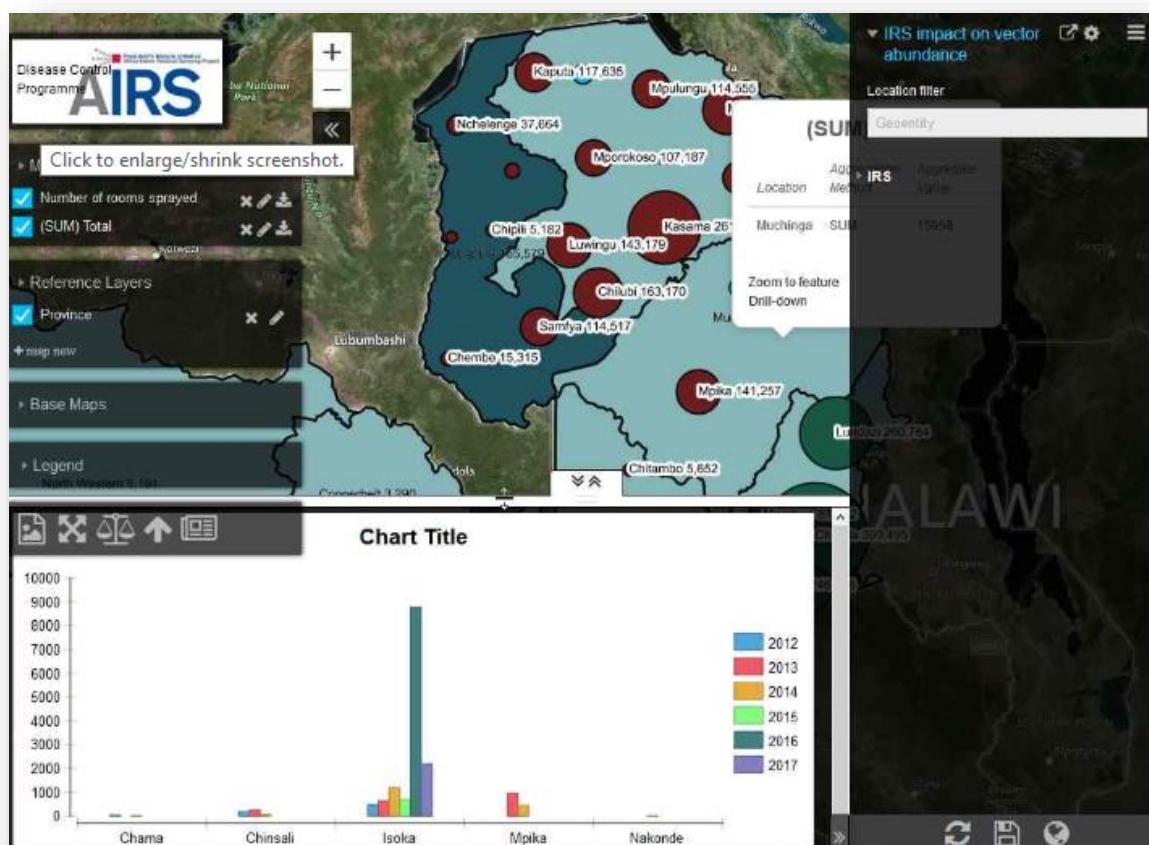
The translucent background of the report is sexy, but it can be annoying because neither the

map behind the report nor the report are clearly defined. Making the background opaque results in a white background which hides the map but makes the report more readable.

✓ Right-click in the report pane and select *Opaque background*

You can change it back by right-clicking in the report pane and selecting *Translucent background*.

Depending on the layout of the report and the shape of the country you are working with, it could be useful to have the report pane use the left part of the page instead of the bottom.

✓ Right-click in the report pane and select *Vertical split*

The pane moves to the left and is collapsed. Because there are now two collapsible panes competing on the left side, it is possible that they get in the way of each other. The report pane is behind the map layer pane and the tab to expand it is hidden. You can still drag the pane bigger or if you want to access the expansion tab with the double-right arrows, you can collapse the map layer pane.

If you want to move the report pane back to the bottom, use the context menu and select *Horizontal split*.

Note: using a context menu in the report pane has repercussions. Charts in an Eclipse report can also be interactive, but a choice had to be made whether the DDMS or the SVG chart image reacts to mouse actions. It is currently impossible to have both. The result is that Eclipse charts in a Kaleidoscope are no longer interactive. They do not respond to mouse actions.

Charts were sometimes used to drill down but now the map is used to drill down, so the functionality has simply been moved. If you do want to use report items to jump to different sections in the report, you must use a table adjacent to the chart.

7. Indicators

7.1. What are Indicators

Although you added *Operational coverage(AG)* to the persisted IRS query, we did not use it in the Kaleidoscope and for good reason.

Calculated values in the query builders typically use ratios. The number of rooms that were sprayed does not tell us much, unless we compare it to a reference value. Reference values could be targets that were defined at the onset of the spray season (planned coverage), or the number of rooms that were visited (operational coverage). Meaningful values that are 'normalised' are indicators. *Operational coverage(AG)* is an indicator defined by the number of rooms sprayed (numerator) divided by the number of rooms visited (denominator).

Indicators don't aggregate correctly. If 100 rooms were visited in Luapula and 100 sprayed, the operational coverage will be 0.8 or 80%. If in Eastern province 200 rooms were sprayed of 300 visited, the operational coverage will be 66.6%. There is no exact way to aggregate the 80% and 66.6%. If you take the average it will be 73.3% but this is not the operational coverage for both areas together. The correct number is (80 + 200) / (100 + 300), or 0.7 (70%).

The indicator value typically aggregates numerator and denominator separately and then divides the two.

Persisting an indicator in a query result is possible, but the resulting data set only contains the values, row by row, and not the calculation that allows you to aggregate these values correctly. Therefore it is recommended not to add calculated values to persisted queries. Most DDMS query builders contain the required numerator and denominator data elements. If you add these to the data set, it is possible to define indicators that will provide the correct results in Kaleidoscopes.

All forms that are created are also data sets, so you can also define indicators on forms. These indicators will automatically appear in the form's query builder so that you can add your own calculated fields to form query builders.

Unfortunately, the pre-defined functional modules such as IRS or Entomology are not data sets, so you cannot define your own calculated fields in the query builders of these modules. You can create a data set from these modules by persisting detailed queries and then define indicators on the resulting data set. The indicators can be used in Kaleidoscopes but will not appear in the original query builder.

7.2. Defining Indicators on Data sets

All the defined data sets are accessible in the Data set Manager.

✓ Open *Dataset Management* from the *Administration* menu

We are currently only interested in the top half: *Persisted results and forms.* Here you see all the queries you persisted and any forms that were created.

Clicking on the edit icon allows you to change the column labels and add information about the source and the data set itself. It does not allow you to change any other attributes. This you need to do in the persisted query or the form itself. However, it is possible to add indicators here.

✓ Open the *IRS* data set for editing and click on **+Add indicator attribute** at the bottom of the pop-up
✓ Type in 'Operational coverage indicator' in the *Label* field
✓ Select Sum and (SUM) Rooms sprayed # for the Numerator
✓ Select Sum and (SUM) Rooms # for the Denominator
✓ Tick the *Percent* check box – this multiplies the result by 100

✓ Submit the indicator

The indicators can be edited or deleted in the Data set Manager.

7.3. Using Indicators in Kaleidoscopes

Once an indicator has been defined on a data set, it is accessible in any Kaleidoscope using the data set.

- ✓ Open the *IRS impact on vector abundance* Kaleidoscope if it is not already open

- ✓ Click on the Kaleidoscope options icon ⚙

- ✓ Expand the *IRS* data set

The indicator is available in the data set but is not added automatically to the Kaleidoscope. To use it in the Kaleidoscope you must tick it.

The indicator can then be used in a thematic map layer and it will aggregate correctly when drilling-down or rolling-up.

8.   Using ETL to Import Data Sets

9.   Introduction

ETL (Extract, Transform, Load) technology is used to extract data from various types of data sources, transform it into a format that is compatible with the target system and load it into the target system.

ETL has been introduced in DDMS 1.06 to load foreign data in XLSX format into the DDMS framework.

Once a spreadsheet has been successfully loaded the first time, the process will have:

- Created a DDMS data set
- Created a DDMS form, including the data entry and query builder functionality
- Optionally created term tree entries for select lists
- Imported the data

This is the first release of ETL in the DDMS and it has some limitations:

- The spreadsheet may not contain formulas
- The error handling is less verbose in some cases
- Only the first tab in the spreadsheet is considered
- It is an *All or Nothing* import process, so any error or inconsistency in the spreadsheet will cause the whole spreadsheet to be rejected
- Consequently, there is no exception spreadsheet listing the failed rows
- The import process is slower than the 1.05 technology
- You cannot use this technology to import data into existing functional modules or manually created forms

Nevertheless, the new ETL import technology hugely simplifies the introduction of foreign data into the system. One way of optimising the advantages of both import technologies is to import a subset of the new data using ETL. This will automate the legwork of creating a form and term tree branches. With the form, the old import functionality becomes available as well. Now you can choose which technology you want to use to import subsequent data.

The listed limitations will be addressed in subsequent releases, as the need arises.

9.1.   Spreadsheet Requirements

The format requirements for ETL spreadsheets are in many ways less rigid than the old import templates since it does not require a class identifier in cell A1 and it doesn't require pre-defined column identifiers in row 2, but there are some other requirements:

- Only XLSX format is supported
- Row 1 is used for mandatory column headers, which must be unique in the spreadsheet. Any column without a unique header will cause the import to fail.
- The spreadsheet may not include formulas
- There should be no grouping/subgrouping or totals in the spreadsheet
- The data type of the data in a column must be consistent. If the import process determines that a column contains dates and any row contains text in that column, the import will fail

9.2. Importing a New Data Set

We will first use *Zambia climate data short.xlsx* to create a new data set. This will quickly create the framework. All data provided is fictional.

The spreadsheet looks like this:

gid	Form Id		District	HFCA	Month	Precipitation	Minimum temperature	Maximum temperature	Predominant wind
19	1		SZ0405	Mwense Rural Health Centre	Jul-17	124.6902	288.5421	305.0797	N
1	2		SZ0405	Zesco Musonda Rural Health Cer	Jul-17	3.754935	288.5421	309.0628	S
2	3		SZ0406	Chabilikila Rural Health Centre	Jul-17	5.146191	288.5421	306.1315	SW
3	4		SZ0406	Chisenga Rural Health Centre	Jul-17	10.90999	292.5987	309.3075	NE
4	5		SZ0406	Kabalenge Rural Health Centre	Jul-17	10.45113	286.8329	306.1329	N
5	6		SZ0406	Kafutuma Rural Health Centre	Jul-17	1.022963	286.808	303.7984	E
6	7		SZ0406	Kambwali Health Centre	Jul-17	3.141927	286.808	305.2245	N
7	8		SZ0406	Kanyembo Rural Health Centre	Jul-17	0.8316529	285.3829	306.3398	S
8	9		SZ0407	Chipako Rural Health Centre	Jul-17	1.588907	290.3475	305.459	NE
9	10		SZ0407	Chishi Rural Health Centre	Jul-17	1.063438	285.1281	305.7609	SW

The format is compatible with weather satellite data except that satellite data only contains a grid ID (gid) and not a geo entity name. This was added to simplify the training. Grid to geo entity mapping can happen in a GIS application. Grid could also be added as a universal type and then the grid can be imported into the geo tree, maintaining the original gid.

The Form Id column was also added for training purposes.

The temperatures are in Kelvin.

Adding a new ETL data set or adding data to an existing ETL data set is a drag and drop effort.

- ✓ Open *Administration > Dataset Management*

✓ Drag *Zambia climate data short.xlsx* into the drop zone
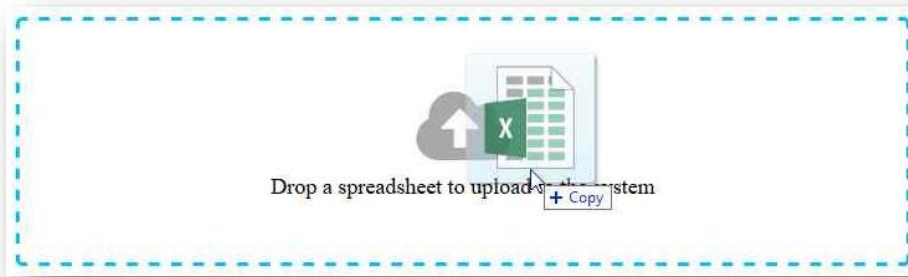
You can also click in the drop zone and browse for the file to upload.

The next page is simply a reminder of the spreadsheet requirements.

✓ Click on *Next*

In the next page you can change the data set label and add a description if you like. The default label (Climate July 2017) is taken from the tab name of the spreadsheet.

✓ Change the label to 'Satellite climate data' and click on *Next*

9.3. Column Types

The ETL process now analyses the spreadsheet and suggests the import format, also known as metadata. Most of the suggested configuration is typically correct but we will go through them systematically and discuss the possible column types:

- ✚ **Ignore**: *gid* - Although *Category* is suggested, we actually don't care about this column. Redundant columns can be ignored during the import process.

- ✓ Set the *Type* of the gid column to *Ignore*.

- ✚ **Form Id**: When a form is created, it always has a form ID field. If this is not present in the spreadsheet, the field is added anyway and random unique form IDs will be generated. If you want a structured form ID that is meaningful, make sure the

column is present and the values are meaningful. Although the import process does not support Excel formulas, you can initially use a formula to generate the ID, then copy the column and paste it back as 'values'. This will overwrite the formula with the value. Alternatively you can omit the column in the spreadsheet and after the import you can change the configuration of the form ID in the Form Generator, to be an expression. This will update all the random form IDs.

✓ Set the *Type* of the *Form Id* column to *Form Id*

🞣 **Location**: This is a column containing geo entities. It could be the name of the geo entity or the geo ID. Within the column, they must all be of the same geo type (universal). There can be several Location columns.

✓ Set the *Type* of the *District* and *HFCA* columns to *Location* and the *Location type* to the corresponding geo type.

✓ **Date**: When the ETL importer finds values of type date in a column, it will pre-populate the *Type* with *Date* and you will not be able to change it. It makes sense that if a spreadsheet contains dates, they are meant to be dates. You could force a change by changing the type in Excel and starting again. Note that if you are expecting a column to be of type *Date* but it shows as text, the column in the spreadsheet actually contains text and not dates, not matter what they look like. You will have to change the type to date in the spreadsheet and start again.

✓ **Decimal or Whole Number**: The importer typically identifies these accurately, but still allows the user a lot of flexibility because it is easy to convert a number to text. Also, the number could be a Form ID or a Geo ID. *Precipitation*, *Minimum temperature* and *Maximum temperature* are Decimal numbers.

🞣 **Category**: This type will create a term field in the data set. If the category is new, the import process will create a flat branch in the term tree under *ETL generated categories*. If you want to allocate an existing root, you can select *Select from tree* in the *Category Option group* and then find the correct root by using the search field or browsing the tree.

Label	Type	Category Options Group	Category Label
Predominant wind	Category	Select from tree	

9.4. Validation

By default there is no validation on the imported data other than the configured type. If you want to put validation on individual columns you can tick the *Validation* check box and you will get the same validation options as you do in the *Form Generator*.

Location columns cannot be user validated.

As always with validation, the stricter it is, the higher the probability that the import fails. Note again that with this importer it is everything or nothing. It is best not to put more stringent validation on the columns than already existed in the source system. On the other hand, it is difficult to add validation once you have already imported data. This is possible, but chances are there is already data in the data set that does not adhere to the validation. In this case you have introduced inconsistencies.
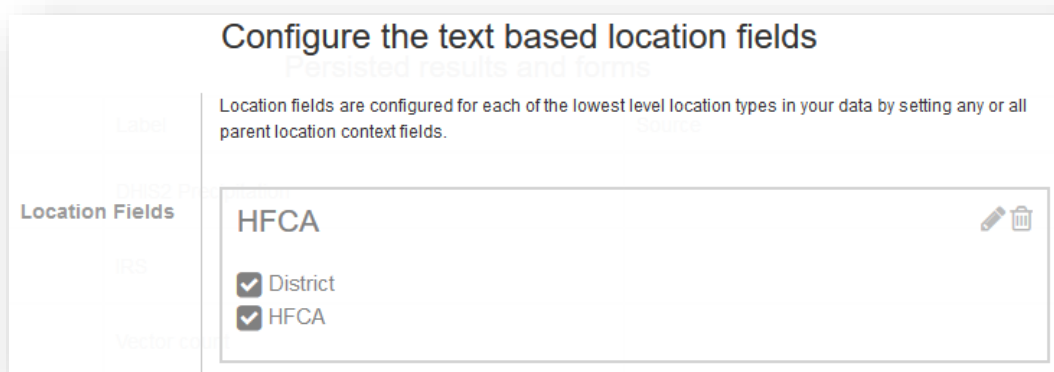
Under validation you also have a *Required* check box. By default this is not ticked. Empty or null values are also accepted. If you tick the *Required* check box null values are not accepted in that column.

- ✓ Tick *Validation* next to *Precipitation*. Set '0' for *Start Range* and tick the *Required* check box.
- ✓ Click on *Next*

9.5. Geography

An imported spreadsheet can have multiple geo columns, belonging to one or more geo entities. In case surveillance, you might have columns Province, District, and Health Facility for the health facility, and Province, District, Settlement for Residential Address.

In the previous page you identified the type of any geo columns in the spreadsheet, but if there are several, the importer needs a bit more information. This is configured in the current page.

The importer has guessed correctly. It has named the location HFCA because it is the lower of the two. This is in fact the one that will imported. The District column adds context in case of ambiguity. The importer assumes they make up a set. If you want to edit this, you can click on the edit icon . You can then rename the geo entity (e.g. Weather station), exclude columns from the Weather station context and create a new context if needed.

✓ Click on *Next*

You get a last chance to review your settings before importing.

✓ Take a look at your configuration and if you are happy, click on *Import*

9.6. Resolving Unknown Geo Names

The import failed because there was a HFCA with an unknown name.

You now get several options:

📍 : Add the unknown geo entity within its context (i.e. in our case, under the associated district)

❌ : Exclude rows with this geo entity from the import. This choice is not remembered. If you import another spreadsheet containg this same geo entity, you will get the same text location problem.

🔗 : Create a synonym for an existing geo entity. This is the one we will explore.

The importer does not suggest a best fit. This could be introduced at a later stage but Soundex algorithms haven't worked well in the past on African names. There is a search field and the geo tree to help you find a match. We will use the search field.

✓ Type 'Kam' in the search field

Because of the District context, the search field immediately finds the match. Without additional context columns the search result could contain more geo entities, but they will all be HFCAs.

✓ Click on the found HFCA to make the synonym icon active and now click on

The problem is resolved and you can import again.

✓ Click on *Import*.

The dataset is imported.

9.7. Forms Created by an ETL Import

Importing the spreadsheet has created a data set. You can now import another spreadsheet in the same format using the same interface. We will do that later.

In the process of creating the data set, a DDMS form was also created.

✓ Open *Administration > Form Administration*
✓ Open the *Satellite climate data* form

All the data set columns are there. You can change validation (but be careful with that), change the field order, add workflow and add expression fields. We will add two expression fields to generate the min and max temperature in °Celcius instead of Kelvin. We will do this as an exercise, as form building is not part of this course.

Exercise

✓ Rename the min and max temperature fields 'Minimum temperature (Kelvin)' and 'Maximum temperature (Kelvin)'

✓ Add two *Decimal Fields*: 'Minimum temperature (Celsius)' and 'Maximum temperature (Celsius)'. These are expression fields. The temperature in °Celsius is the temperature in Kelvin -173.15.

✓ Take the Required attribute off the *Precipitation* field

✓ Add an *Indicator Field*: 'Effect of precipitation on max temp' = *Avg Maximum temperature (Celsius) / Avg Precipitation.* This is a nonsense indicator.

Check your changes in *Dataset Management*.

✓ Add the form CRUD and QB to the menu. Add them to the *Administration* role as well.

9.8. Importing Spreadsheets into an Existing Data Set

If (and only if) a data set was created by importing it using the ETL interface, you can drop subsequent spreadsheets into the data set. We will use the same spreadsheet and change some entries:

- ✓ Edit the *Zambia climate data short.xlsx* spreadsheet: Change the Predominant wind in the first row to 'W' and on the third row to 'South West'. Save the file.
- ✓ Drop the file into the ETL drop zone

The interface recognises that this spreadsheet matches one that has already been imported, so it gives you the option to update an existing one, but you can also create a new data set. We will update the existing one.

- ✓ Click on *Update an existing dataset*

The interface knows only one existing data set matches the spreadsheet. You have the option to add/update the data to the spreadsheet or replace the data. Be very careful here. Replacing the data does not mean 'update existing data'. It will destroy all the data in the data set. Add/update will update the data if the form ID is in the spreadsheet. Otherwise it will add the rows to the data set and give them a new, unique form ID.

- ✓ Click on the icon.

The review page is shown again. This is for you to confirm that you have the correct data set.

- ✓ Click on *Import*

9.9. Resolving Unknown Category Options

This time there is no issue with the geography, because you created the appropriate synonym the first time you uploaded the spreadsheet. Due to the edits you did, it doesn't know what to do with *South West* and *W*.

When you created the *Satellite climate data* data set, you set *Predominant wind* to *New Category Option Group*.   This created a term tree branch which is now used for validating new imported data.

The branch was created as a flat branch because it is not possible to determine any hierarchy from an imported spreadsheet, but you can manually move and modify the branch.

If an unknown term is encountered in a new spreadsheet, you will get the same options as you would for unknown geo names.

✓ Resolve the issues by making *South West* a synonym of S*W* and by adding *W* as a valid new term.

Term synonym support is new to 1.06. It makes the DDMS an advanced data cleaning application.

✓ Import the full *Zambia climate data.xlsx* spreadsheet. We will use this later in the course. The import takes about an hour.

10. The DHIS2 Interface

10.1. Introduction

The DDMS contains detailed operational data allowing in-depth analysis. This also includes flexible (dis)aggregation using terms in the term tree. The term tree supports unlimited depth and aggregation at the nodes. This makes the system very powerful in operational decision making environments.

The Ministry of Health (MoH) often operates one or more DHIS2 instances and typically would like an aggregated form of these data in the DHIS2 data repository. Importing data into DHIS2 is not always straightforward. Even the manual configuration of new data sets can require a fair amount of effort in DHIS2.

DDMS 1.06 has the technology to simplify this process, both for native DDMS data and foreign data sets.

Using the ETL interface, any data set can be cleaned and imported in a few easy clicks, as we have seen. This process automatically creates the required DDMS metadata which can be redeployed for DHIS2.

In addition, no matter how granular the imported data, the DDMS query builders can easily strip, filter and aggregate the data so that it is in the format that the MoH would like to see in DHIS2.

Data sets are typically aggregated at the level of Health facility or District in DHIS2. Aggregation periods are typically week or month. This is not a DHIS2 limitation but a configuration typically favoured by the MoH.

DDMS 1.06 has a powerful interface that can take both the metadata and data from a data set and push this to an existing DHIS2 installation. The only effort required is to make sure the geo trees of both systems are properly mapped.

This course explains how to push a data set to DHIS2. It does not cover any DHIS2 training.

10.2. Process

The DHIS2 interface uses persisted queries. If you want to push a complete data set at the same detail level that you imported it through the ETL interface, you can create a persisted query that contains this disaggregated information.

The process is as follows:

1. Create a suitable query (the query requirements are detailed below)
2. Add any required indicators in the Dataset Manager

3. Connect to an existing DHIS2 system
4. Ensure that the geo entities in the query result have been mapped to DHIS2 org units
5. Add the persisted query to the interface
6. Export the query's data set

The interface will assess the query and create a data set in DHIS2. The data set will contain each numeric column as a data element. The data elements are also grouped (DHSI2 data set group) to be easily accessible in DHIS2 analytics.

The data elements will have the same aggregation in DHIS2 as they have in the persisted query.

The query's time grouping becomes an attribute of the DHIS2 data set.

The geo entities in the query's geo column are exported as the org units associated with the data.

Each text column generates a DHIS2 category. These categories are combined into a DHIS2 category combination which is linked to the data set. This allows disaggregation of the data element.

Any indicators in the data set will also create indicators in DHIS2 (1.06.0011).

Every time the persisted query is exported, the metadata is also exported in update mode. This allows the DHIS2 metadata to stay in sync with the DDMS metadata.

It is obvious that an interface between two systems that are not fully integrated has limitations. The interface is fairly robust, because it controls the DHIS2 unique object IDs, so even unwanted interventions in the metadata by a DHIS2 user will typically be undone by the next metadata update.

In turn, this implies that it is typically useless trying to fine-tune the imported metadata in DHIS2, as each subsequent export will undo most changes that were effected in DHIS2.

Because term fields can have a substantial amount of selectable terms, it would not be practical to create input tables in DHIS2 that contain a row for every possible category option combination. This could be a table with 10,000 columns. Because of this, the categories and category combination are configured with Data dimension type *Attribute*. The categories show as select lists and the data elements are not disaggregated in a table but only show the values associated with the selected attribute values.

It is not the intention to input data for these exported data sets manually in DHIS2. In theory it is possible but it would be tedious and there is always the chance that subsequent exports override any manual entries or edits.

10.3. DHIS2 Persisted Query Requirements

To be able to export a data set to DHIS, there are some requirements that govern the query:

- There must be a set of geo columns (geo name and geo ID)
- Only one set of geo columns is supported
- There must be time grouping. This could be either a date or a combination of *Calendar year* with either *Quarter*, *Month* or *Week*
- There must be at least one numerical column that can be aggregated
- Every numerical column must have an aggregation function
- It may not contain calculated columns, as it will not be known in DHIS2 how to aggregate them
- Indicators can be added to the data set and they will be exported (1.06.0011). These should replace any calculations that are readily available in DDMS query builders

Once a DHIS2 query has been persisted and exported, it is strongly discouraged to edit the query. In theory this is possible but there are several corner cases that may create inconsistencies in DHIS2. Therefore, a persisted query that is destined for the DHIS2 interface should be very carefully thought through and discussed with all stakeholders before exporting it for the first time. If you do need to modify the query, it would be best to remove all evidence of the data set from DHIS2. As a minimum, if you do change a persisted query that has been used in the DHSI2 interface, re-upload all the data for the geo entities and time periods that you have already uploaded. An exception is saved filters. These can be modified and persisted without impacting the interface.

Changing the time period grouping or geo grouping of a query is very strongly discouraged.

Note: The interface in not destructive. It will never delete anything (metadata or data) in DHIS2, even if these were originally created by the interface. At best it updates metadata or data. Even if you remove the data set from the interface, nothing will be deleted in DHIS2.

We will create several DHIS2 persisted queries to use in the interface:

Mosquito Collections and Abundance QB

- ✓ Create and persist the following query (call it 'DHIS2 vector count'):

Columns

District Entity name (Geo entity)
District Geo entity ID (Geo entity)
Calendar year
Epi week
Collection method
Species (AG)
- Anopheles funestus sensu lato
- Anopheles gambiae sensu lato
- Culicinae

(SUM) Number females
(SUM) Total

This query should return 1485 rows.

✓ Create and persist the following query (call it 'DHIS2 vector resistance'):

Columns

District Entity name (Geo entity)
District Geo entity ID (Geo entity)
Calendar year
Epi week
Test method
(SUM) Number exposed
(SUM) Number dead
Assay Type
- Adult diagnostic

(SUM) Corrected number dead
(SUM) Control number
(SUM) Control number dead
Species used
Active ingredient

This query should return 322 rows.

✓ Create and persist the following query (call it 'DHIS2 climate'):

```
Columns
Health facility catchment area Entity name (HFCA)
Health facility catchment area Geo entity ID (HFCA)
Epi year
Month
(AVG) Precipitation
(MIN) Minimum temperature (Celsius)
(MAX) Maximum temperature (Celsius)
Predominant wind
```

This query should return 1916 rows.

10.4. Hierarchies

Categories that are term fields can have a hierarchy in the DDMS. This technology is not supported in DHIS2 at this time. It is possible to simulate two levels by creating option groups and option group sets. In fact more levels could be simulated by creating an option group for every node under the field root. This could be done by the interface but there are several complications:

- The simulated hierarchy would not be visually apparent in DHIS2. Every node would be a dimension in DHIS2 analytics and any relationship between these dimensions would be lost. The user would have to be aware of the relationships.
- What do we do with data allocated to nodes instead of leaves (you cannot reference option groups or option group sets in DHIS2)?
- What happens if a DDMS user changes the structure of a tree or even the root of a term field?

There are possible solutions to all of these complications, but they would make the interface substantially more complicated – not only to implement but also to understand. Therefore the first release of 1.06 does not support any effort to simulate the hierarchy.

If option groups or group sets are created manually in DHIS2, the interface will not delete or modify them. This allows users to decide how they want to organise the options without the DDMS enforcing anything.

The geo hierarchy is supported in DHIS2, but not a meshed hierarchy. It should be possible to map each DDMS geo level to a DHIS2 org unit level. If need be, the persisted query can be aggregated to a geo level that is unambiguously supported by DHIS2.

On the whole, DHIS2 stores aggregated data. It is typically useful to reduce the complexity of hierarchical data by aggregating it as we did in the first query (*DHIS2 vector count*).

10.5. Name duplication

DHIS2 is sensitive to name duplication. Data elements, categories etc. must all have unique names within their class. This can cause problems for several reasons.

- The DDMS can have multiple fields with the same name in different query builders
- The DDMS can have options (terms) or categories that already exist in DHIS2 (such as Gender, Male, Female) but that the DDMS does not know about
- Other more obscure corner cases that we won't go into

The interface solves these issues in several ways. For one, it appends a space to all names. This is fairly unobtrusive but will most likely resolve conflicts with existing names in DHIS2.

Any term field that has the same root as another term field will not be created as a different category. An example is *Species* in the vector collection QB and Species used in the resistance QB. If these have the same root, they are in effect the same category. It would not be a major problem creating these as different categories, but if both fields were called Species, you would get name duplication.

Data elements that are essentially the same but occur in different exported queries, will be identified as being the same.

The only use case that has currently been identified as a potential issue, is if two different fields have the same name in the DDMS. Since they are in fact completely different, the interface will create both with the same name and DHIS2 will reject the second. The solution is to manually change the field name in the default disease locale.

10.6. Connecting to a DHIS2 installation

- ✓ Open *Administration > DHIS2 Management*

There are three steps to uploading data to a DHIS2 instance:

- Configure the DHIS2 instance
- Map the DDMS geo entities to the DHIS2 org units
- Add and export the persisted queries

You can access the different steps by selecting them on the left of the page.

The DHIS2 base URL is essentially the URL the DHIS2 user would use to get to the DHIS2 login page. If dhis2-live is running on your computer, you can use http://127.0.0.1:8082 or http://localhost:8082. The interface supports both http and https.

- ✓ Make sure DHIS2 is started on your laptop
- ✓ Enter http://127.0.0.1:8082 as the *DHIS2 base URL*

The authorisation type for DHIS2 can be Basic password authentication or OAuth 2 authentication, but in 1.06.0010 only *Basic password authentication* is supported.

The remaining fields are the DHIS2 user name and password. Note that in DHIS2 data as well as users are allocated to organisation units. Make sure the user you configure here has access to the organisation units in the data you want to export.

- ✓ Enter 'admin' and 'district' for the user name and password
- ✓ Click on *Connect*

During the connection process, the interface is also downloading the DHIS2 organisation unit hierarchy and is preparing the framework for the interface, so it takes a few seconds to complete. If there is something wrong with the instance configuration or DHIS2 is offline, you will get an error message. If no organisation units are defined in DHIS2, you will get an unspecified error.

10.7. Mapping Geo Entities to Organisation Units

This is the only tedious part of the interface, but it needs to be done since there is no way to fully automate the mapping of two independent systems.

When you connect to the DHIS2 instance, the organisation units and organisation unit levels are fetched from DHIS2. These need to be paired with geo entities in the DDMS.

✓ Click on *Geo mapping* on the left of the page
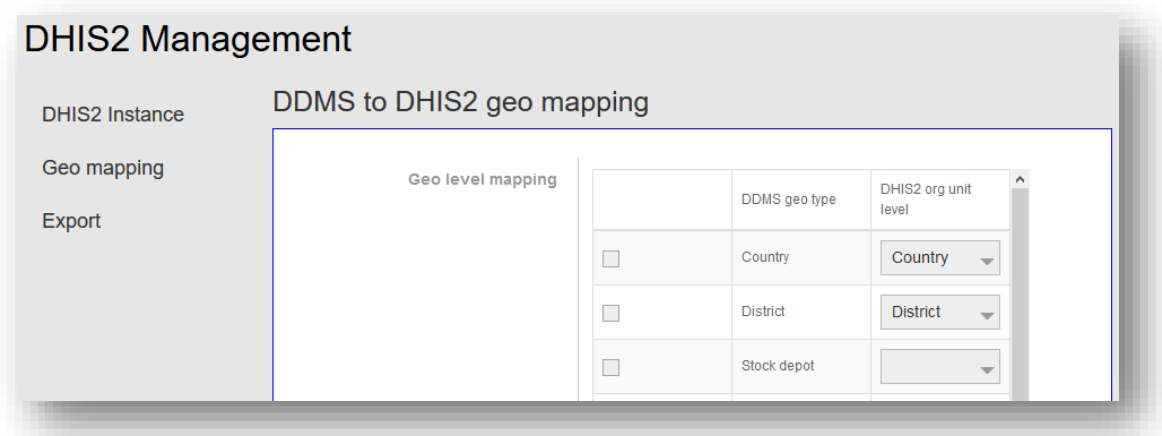
The top section is the *Geo level mapping*. As from 1.06 we are moving away from the term Universal. This originates from ontology terminology but is not intuitive to new DDMS users. Therefore henceforth universals are referred to as geo levels.

An algorithm has run that performs a best guess matching, but none of the matches are confirmed. You need to validate the matches by ticking the check box to the left of the geo level or geo entity name. If you made a mistake you can un-tick it.

✓ Confirm the matches for *Country*, *Province* and *District*.

Important note: Mapping is only necessary for the geo entities that are in the query output. If all your DHIS2 queries are aggregated at district level, it is sufficient to map the districts.

It is possible to map multiple geo levels to the same organisation unit level. This can be useful if the DDMS geo level tree branches (DHIS2 does not have org unit level branches), or if levels are missing in DHIS2, e.g. Collection site could be paired with Settlement if there is only one collection site in a settlement. Note however, that many to one mapping can result in ambiguous data from the DHIS2 perspective. If you have two distinct rows for two different collection sites but they both map to the same DHIS2 settlement, you risk a duplication error occurring.

If you cannot map geo entities that you want to export, the administrators of the DHIS2 instance will have to make sure the relevant organisation units exist. It is impossible to export data containing geo entities that are not mapped to organisation units.

The bottom section is the geo entity mapping section and is collapsed.

✓ Expand the tree down to district level.

Because the DHIS2 org units are based on the DDMS geo tree, the suggested matches are almost 100%. Unfortunately this will not always be the case.

There is no option to automatically confirm all matched geo entities, because this is quite dangerous. If there is a strong demand to provide this option, it can easily be done.

There are a few issues with the suggested mapping. Some district names have changed since the DHIS2 org units were configured, such as *Chienge* district in *Luapula* province. No match was found.

✓ Find *Chienge* district in the tree and type 'Chie' in the search box.

Chiengi district is found.

✓ Select *Chiengi (District)* and confirm the match.

In the same province you find Lunga district. It could be wrongly matched with Lunga settlement. There is a pending change request to limit match results to the corresponding geo level but currently (1.06.0010) this is not happening.

✓ Type 'Lunga' in the search box and select *Lunga (District)*. Confirm the match.
✓ Check the matched geo entities and confirm them if you are happy with the match. Pay attention to the org unit level and change the match if necessary. Complete any missing matches.

10.8. Adding a Persisted Query to the Interface

✓ Click on *Export* on the left of the page
✓ To add a query to the interface, click on the ➕
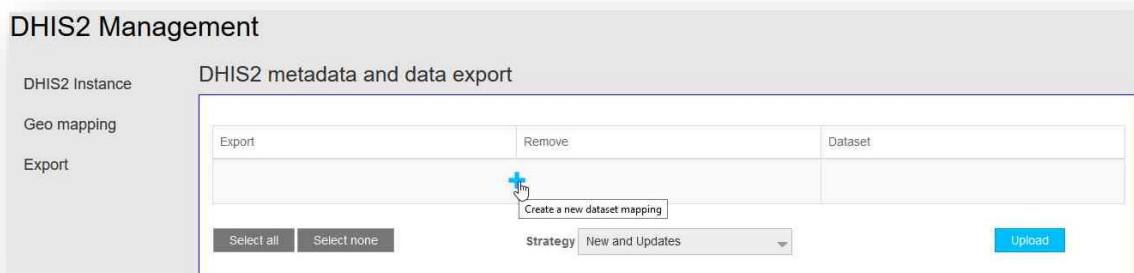
You can select any query that you persisted, even if it does not adhere to the DHIS2 interface requirements. This is because the persisted query can be changed and saved again, and it will still be in the interface definition, so the validation must happen each time you actually export the data.

✓ Add *DHIS2 vector count* and type 'DDMS vector collection' in the *DHIS2 dataset name* field
✓ Click on *Submit*

You can add multiple data sets and select which ones you want to export. You also have the option to *Select all* or *Select none*.

You can change the filter(s) on the persisted query and save it again. These will automatically apply to the interface. This way you could decide to upload the data for one or several districts or provinces. Alternatively, you could create a persisted query for each province and add them all independently. Then you can tick the ones you want to upload.

- ✓ Select *DHIS2 vector count* and click on *Upload*

The interface will now generate JSON files for the metadata and data for DHIS2, make the connection and upload both files. A copy of the files can be found in C:\MDSS\tomcat\webapps\Zambia\DHIS2. There is also an export log here. These files are in JSON format. If you want to scrutinise the files, it would be best if you install a JSON viewer plug-in in your browser.

Once the upload is completed, you can take a look at the results in DHIS2.