

**7th Framework Programme
ENV.2010.4.1.2-2**

**Integrating New Data Visualisation Approaches of Earth
Systems into GEOSS Development**



Project Nr: 265178

**QUAlity aware VIualisation for the Global Earth Observation system
of systems**

Heuristic Evaluation of Greenland

Date prepared: December 17th, 2013

Aston University

Introduction

This report documents the performance and results of a heuristic evaluation (HE) of the Greenland web application (version used was <http://giv-uw.uni-muenster.de/vis/v2/>), and provides preliminary guidance for improvement based on the problems identified within the application¹. Considering the scope and delivery medium of the application and its highly interactive nature, we opted to use a state-of-the-art list of heuristics designed specifically for evaluation of interactive web-based systems [1]. The following section describes the way in which the heuristic evaluation was carried out. Thereafter, an overview of the results is provided. This is followed by a detailed description of each identified problem. This report closes with a general discussion of the results.

Method

The heuristic evaluation was conducted based on completion of a set of representative tasks (designed to explore the full extent of the system as far as possible – see Table 3). During completion of the tasks, the user interface was evaluated based on its conformance with a set of heuristics [1] (see Table 1), and where contraventions of the heuristics were identified – that is, where usability problems were considered to be found – each problem was allocated a severity ranking (see Table 2) to support prioritisation of problem fixing. The heuristics used were suggested by Petrie *et al.* [1] and are divided into 4 categories, as shown in Table 1.

Table 1: List of Heuristics (taken from [1]).

Category	#	Heuristic	Description
Physical Presentation	1	Make text and interactive elements large and clear enough	Default and typically rendered sizes of text and interactive elements should be large enough to be easy to read and manipulate.
	2	Make page layout clear	Make sure that the layout of information on the page is clear, easy to read and reflects the organization of the material.
	3	Avoid short time-outs and display times	Provide time-outs that are long enough for users to complete the task comfortably, and if information is displayed for a limited time, make sure it is long enough for users to read comfortably.
	4	Make key content and elements and changes to them salient	Make sure the key content and interactive elements are clearly visible on the page and that changes to the page are clearly indicated.
Content	5	Provide relevant and appropriate content	Ensure that content is relevant to users' task and that it is appropriately and respectfully worded.
	6	Provide sufficient but not excessive content	Provide sufficient content (including Help) so that user can complete their task but not excessive amounts of content that they are overwhelmed.
	7	Provide clear terms, abbreviations, avoid jargon	Define all complex terms, jargon and explain abbreviations.
Information Architecture	8	Provide clear, well-organized information structures	Provide clear information structures that organize the content on the page and help users complete their task.
Interactivity	9	How and why	Provide users with clear explanations of how the interactivity works and why things are happening.
	10	Clear labels and instructions	Provide clear labels and instructions for all interactive elements.

¹ It should be noted that expert heuristic evaluations are best conducted by up to 5 independent usability experts; in this case, due to resource limitations, the evaluation was performed by one expert and so we would advise that, although we believe that we have captured a majority of usability issues, further iterations of evaluation by additional experts would likely increase the percentage of found usability problems.

		Follow web conventions for labels and instructions (e.g. use of asterisk for mandatory elements).
11	Avoid duplication/ excessive effort by users	Do not ask users to provide the same information more than once and do not ask for excessive effort when this could be achieved more efficiently by the system.
12	Make input formats clear and easy	Make clear in advance what format of information is required from users. Use input formats that are easy for users, such as words for months rather than numbers.
13	Provide feedback on user actions and system progress	Provide feedback to users on their actions and if a system process will take time, on its progress.
14	Make the sequence of interaction logical	Make the sequence of interaction logical for users (e.g. users who are native speakers of European languages typically work down a page from top left to bottom right, so provide the Next button at the bottom right).
15	Provide a logical and complete set of options	Ensure that any set of options includes all the options users might need and that the set of options will be logical to users.
16	Follow conventions for interaction	Unless there is a very particular reason not to, follow web and logical conventions in the interaction (e.g. follow a logical tab order between interactive elements).
17	Provide the interactive functionality users will need and expect	Provide all the interactive functionality that users will need to complete their task and that they would expect in the situation (e.g. is a search needed or provided?).
18	Indicate if links go to an external site or to another webpage	If a link goes to another website or opens a different type of resource (e.g. PDF document) indicate this in advance.
19	Interactive and non-interactive elements should be clearly distinguished	Elements which are interactive should be clearly indicated as such, and element which are not interactive should not look interactive.
20	Group interactive elements clearly and logically	Group interactive elements and the labels and text associated with them in ways that make their functions clear.
21	Provide informative error messages and error recovery	Provide error messages that explain the problem in the users' language and ways to recover from errors.

We chose to use this set of heuristics due to their focus on the increasing interactivity of web applications which is in stark contrast to the more traditional web sites at which previous heuristics, such as Nielsen's seminal heuristics [2], were targeted.

During the evaluation, the severity of identified usability problems was classified using Nielsen's Severity Ranking² (see Table 2).

Table 2: Nielsen's Severity Rankings

Severity #	Description
0	I don't agree that this is a usability problem at all
1	Cosmetic problem only: need not be fixed unless extra time is available on project
2	Minor usability problem: fixing this should be given low priority
3	Major usability problem: important to fix, so should be given high priority
4	Usability catastrophe: imperative to fix this before product can be released

Evaluation of the Greenland application was based on the assumption that the application is targeted at exclusive use by domain experts; it was therefore assumed that domain knowledge or an initial training stage will guide the first steps of users. Consequently, we have not evaluated the interface on the basis of being initially self-explanatory for a user without requisite domain knowledge; we have assumed that a user would at least have consulted the guidance on how to use the Greenland application as provided to us, including guidance on concepts related to visualisation of uncertainty.

² <http://www.nngroup.com/articles/how-to-rate-the-severity-of-usability-problems/>

We conducted an initial analysis of the user interface; thereafter, a selected tutorial³ as provided by the Greenland developers was used to guide deeper, task-driven exploration of the user interface, providing context in terms of where usability problems were found (and hopefully easing the task of reproducing and fixing the identified problems). Hence, as listed in Table 3, Task 1 was initially performed without any guidance and, thereafter, Tasks 2-6 were selected from the exercises in the tutorial (e.g., Task 2 corresponds to Exercise 1 in the tutorial, and so on). The evaluation was conducted using a laptop running Windows 7 and Google Chrome version 31.

Table 3: Set of Tasks Performed

Task #	Description
1	Initial Analysis
2	Learning how to use the tool
3	Multidimensional interaction
4	Categorical data
5	Combined visualisations
6	Data preparation

During execution of the selected tasks, each identified problem was recorded with the following details: a) context in which the problem was identified; b) description of the problem; c) heuristic(s) breached; d) usability problem severity; and e) suggested fix. We report the results as follows: i) an initial summary table identifying all usability problems found (see Table 4); ii) overview graphs showing the distribution of heuristics breaches across categories (see Figure 1) and their collective severity (see Figure 2) to establish a general sense of the usability of the Greenland application; iii) a series of graphs demonstrating the extent to which specific heuristics within each of the different categories have been breached (see Figure 3 to Figure 6) to help identify where the majority of usability issues have occurred; and iv) detailed descriptions of each identified problem to support reproduction and fix.

Summary Results

A summary list of the usability problems identified is provided in Table 4, with each problem allocated a unique identifier:

Table 4: Summary List of Identified Usability Problems

ID	Description	Heuristics Breached	Severity
1	Inconsistent time slider	#4, #7, #9, #10, #12	1
2	Mimic real world objects (time controls)	#7, #8, #9, #10, #12, #14, #16, #17, #20	2
3	Provide consistent icons	#2, #10, #12, #16	2
4	Logically identify interactive elements in the feature selection	#8, #9, #12, #16, #17, #19, #20	2
5	Make relevant information clearly visible	#4, #9, #16, #17	2
6	Consistent Map Navigation	#9, #10, #12, #16, #17, #20	3
7	Inconsistent pointer actions	#7, #9, #10, #12, #16, #17, #20	3
8	Inconsistent Swipe Icon	#8, #9, #16, #17	2
9	Graticule representation inconsistent	#9, #13	2
10	Swipe representation inconsistent	#9, #14	2
11	Resource name not descriptive	#9, #10, #12	2

³ <http://giv-wikis.uni-muenster.de/agp/bin/view/Main/UncertaintyVisualisationWorkshop>

12	Add a resource	#9, #12, #14, #17	3
13	Add a resource to the map on the right	#2, #4, #9, #10, #14	4
14	Provide feedback while loading a resource	#4, #9, #13, #17	3
15	Provide feedback when a resource not loaded/unavailable	#21	3
16	Error messages are inconsistent	#5, #9, #10, #13, #21	3
17	Improve visualization icons	#9, #10, #16, #17	2
18	Clearly state the purpose of interactive elements	#5, #9, #10, #13, #19	2
19	Standard conflict/Unclear purpose of interactive element	#9, #10, #16, #17	2
20	Unidentified units	#5, #7, #21	3
21	Functions are not logically grouped	#2, #9, #10, #14, #20	3
22	Inform the user of new interaction techniques	#2, #9, #10, #13, #15, #17	3
23	Logically group controls to handle confidence interval	#8, #9, #10, #12	2
24	Misspelled resource type	#5	1
25	Update time information when a time related resource is loaded	#4, #10, #13	3
26	Inconsistent threshold input	#12, #14	2
27	Exceedance intervals are not available	#5	4
28	Legend description is not clear	#5	2
29	Feature selection does not work properly	#9, #14, #17	3
30	Not able to load NetCDF data	#21	4
31	Unable to load O&M resource	#21	4
32	Unable to load a raster resource	#21	4

Figure 1, below, summarises the number of times heuristics were breached according to the 4 principal categories.

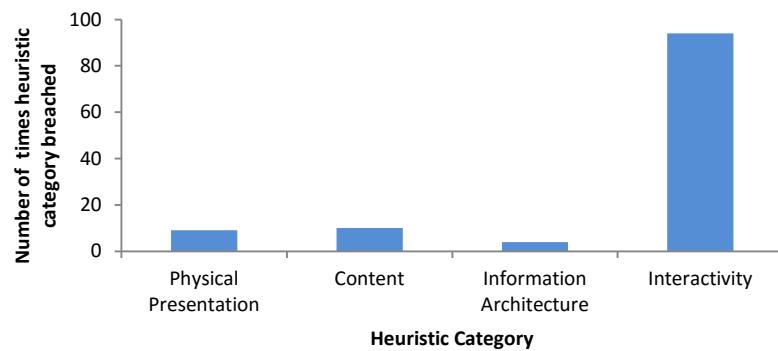


Figure 1: Tally of Heuristics Breaches Grouped According to the 4 Principal Categories

This highlights that, from a usability perspective, Greenland suffers most from issues associated with interactivity; these are issues that have the potential to impact its use and acceptance by target end users. Specifically, interactivity heuristics were breached a total of 94 times across 29 of the identified usability problems; as Figure 1 shows, in most instances, a single usability problem was the result of more than one of the heuristics in the interactivity category being breached.

Figure 2 highlights the severity of the 32 identified usability problems according to Nielsen's Severity Rankings:

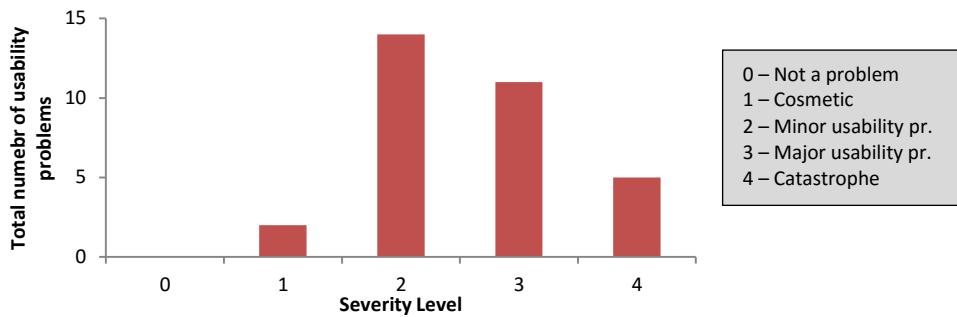


Figure 2: Number of Usability Problems per Severity Level

As Figure 2 illustrates, although almost 50% (14) of the identified problems were minor (level 2) and represent a lower priority for fixing, 5 usability problems were considered catastrophic (level 4); such problems should be considered high priority in terms of fixes because they can significantly impact use and even block users from using the application. The relatively high number of major usability problems (level 3) is also cause for concern, and should be considered as important to fix.

Figure 3 shows the number of times heuristics in the *Physical Presentation* category were breached across the 32 identified usability problems. In fact, these breaches were isolated to 8 different usability problem instances (see Table 4)

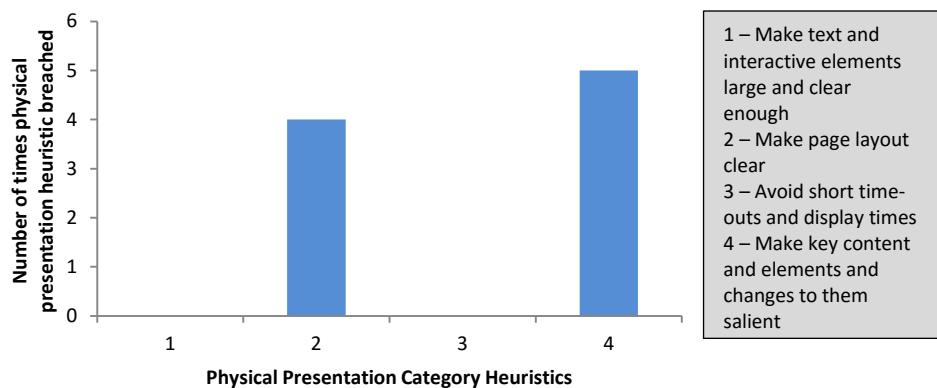


Figure 3: Number of Heuristics Breaches Within Physical Presentation Category

This indicates that the majority of associated problems relate to key interactive elements not communicating changes in the user interface. Moreover, there are also problems with the layout of the application where actions are not as logically grouped as they could be to ease user comprehension and interaction.

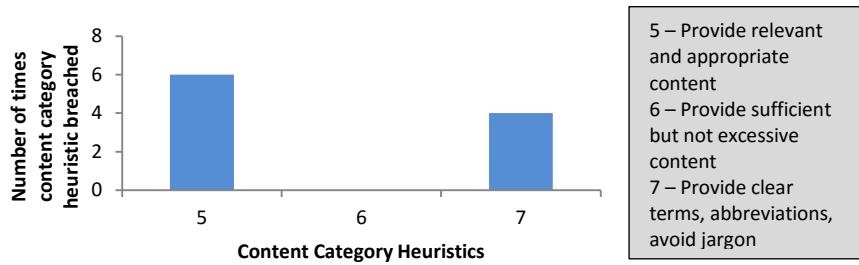


Figure 4: Number of Heuristics Breaches Within Content Category

Figure 4 reflects the number of times content-related heuristics were breached across the 32 identified usability problems. Related problems concern use of terms that may not clearly reflect the function provided by the application or even use of names in legends that does not clearly state meaning – in some instance, for example, abbreviations have been used in legends which fail to adequately inform the user about the meaning of the content.

The number of times the (single) Information Architecture category heuristic was breached is shown in Figure 5. In reality, only 4 usability problems were identified in relation to unclear information structure. Specifically, they relate to unclear information layout (e.g., unclear structure of information on controls or the need for additional controls to easily map actions) or a requirement to better structure feedback (e.g., to improve perception of actions or help users to recover from errors).

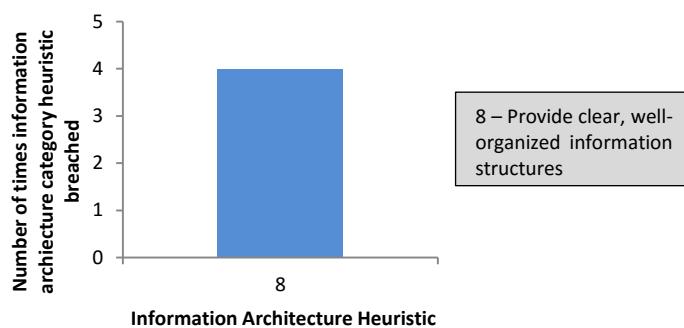


Figure 5: Number of Heuristics Breaches Within Information Architecture Category

Figure 6 highlights the number of times interactivity-related heuristics were breached across the 32 identified usability problems. All bar 3 of the usability problems identified concerned breaches of interactivity-related heuristics. As can be seen, most of the usability problems associated with interactivity issues relate to the failure of the design to clearly state how and why things are happening. Some identified actions are not clear and may lead to confusion; use of labels is sometimes unclear and interactive functionalities are sometimes inconsistent in terms of expected standards and behaviours as provided by other applications.

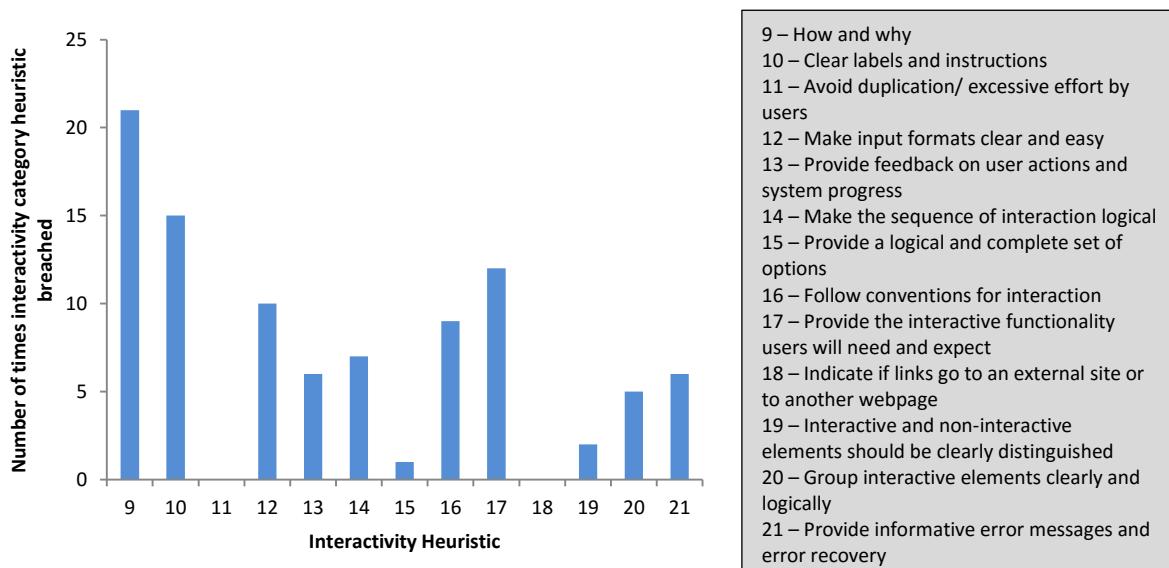


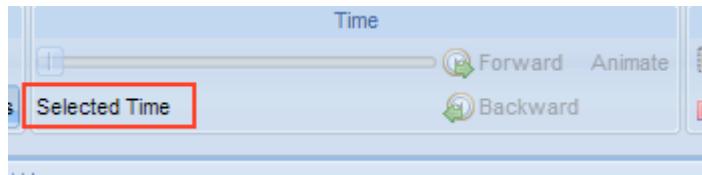
Figure 6: Number of Heuristics Breaches Within Interactivity Category

Detailed results

This section outlines the nature, severity, and fix suggestions for the 32 identified usability problems in turn.

1. Inconsistent time slider

Context: Task 1 – No interaction was performed with the application. Identification of the problem was based on an initial observation of the user interface.



Description: Initially, the time slider cannot be used yet it is not perceived as disabled and the label message ("Selected time") is unclear or at least, not informative. In addition, the message may suggest an interaction which is not possible before loading a time-related resource.

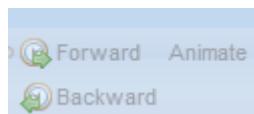
Severity: 1

Breached Heuristics: #4, #7, #9, #10, #12

Suggestions to fix: Always show the option on the toolbar and disable or hide all elements of the time group if a time-related resource is not loaded. The "Selected Time" label can be hidden until such time as it has contextually relevant meaning.

2. Mimic real world objects (time controls)

Context: Task 1 – No interaction was performed with the application. Identification of the problem was based on initial observation of the user interface.



Description: The icons used to control a time-based animation do not resemble common controls for animations, meaning that their representation may not be intuitive.

Severity: 2

Breached Heuristics: #7, #8, #9, #10, #12, #14, #16, #17, #20

Suggestions to fix: Mimic the media player metaphor, which is typically familiar to users. *Forward* and *backward* buttons could be represented with the related step forward and backward icons (i.e., ▶ and ◀) and *animate* could be represented with a play icon (i.e., ▶). Finally, the layout of the toolbar group "Time" could be rearranged with controls (i.e., play, backward and forward) on the left and the slider on the right as in common media players.

3. Provide consistent icons

Context: Task 1 – No interaction was performed with the application. Identification of the problem was performed based on initial observation of the user interface.



Description: Icons in the toolbar at the top should be consistent. Interactive items should have an equal representation (i.e., an icon *and* a label or at least an icon with tool tips to describe the action on mouse over). Inconsistent representations and inconsistent dimensions should be avoided.

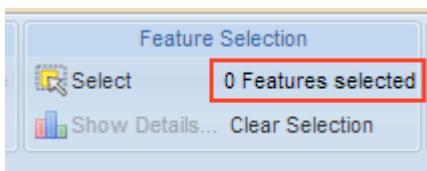
Severity: 2

Breached Heuristics: #2, #10, #12, #16

Suggestions to fix: Define a common size for all the buttons in the toolbar so that users can understand, and easily and consistently identify each available action.

4. Logically identify interactive elements in the feature selection

Context: Task 1 – No interaction was performed with the application. Identification of the problem was performed based on initial observation of the user interface.



Description: In the feature selection toolbar group, the number of selected features is shown to the user. That element is purely informative and, not being interactive, would be better located outside of the toolbar.

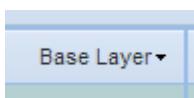
Severity: 2

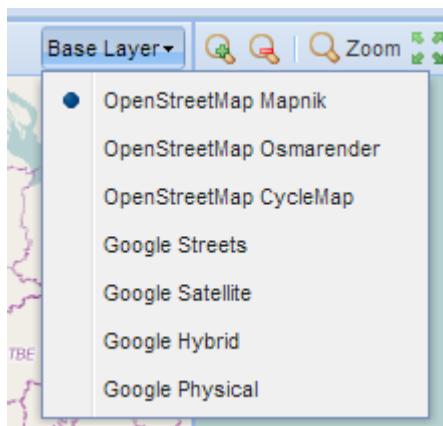
Breached Heuristics: #8, #9, #12, #16, #17, #19, #20

Suggestions to fix: Consider adding the number in concert with the select icon, or create a logically defined information group to provide such data to the user. Alternatively, a status bar at the bottom of the window could be used to provide contextual information.

5. Make relevant information clearly visible

Context: Task 1 – No interaction was performed with the application. Identification of the problem was performed based on initial observation of the user interface.





Description: On the toolbar over a map, a combo box is used to identify the base layer. Although the user can see a label describing what the component relates to, the information related to the value set within the combo box is hidden.

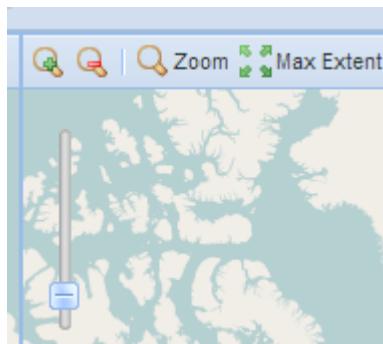
Severity: 2

Breached Heuristics: #4, #9, #16, #17

Suggestions to fix: The combo box could show the layer/value selected “OpenStreetMap Mapnik” instead of “Base Layer” and the action to select the “Base Layer” could be better represented as a label on the left of the combo box.

6. Consistent Map Navigation

Context: Task 1 – No interaction was performed with the application. Identification of the problem was performed based on initial observation of the user interface.



Description: Map navigation operations do not clearly follow common metaphors used by related tools (e.g., Google Maps). At the top, left-hand corner of the map, the slider does not clearly state the action or relative value since the meaning of its position is not clear (i.e., which direction increases or decreases its value). In addition, the Max Extent icon does not provide a best fit of the map in the panel. When the map is at its minimum zoom level, the user still has to pan the map to visualise hidden areas of the map.

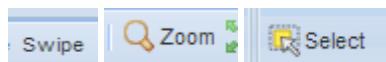
Severity: 3

Breached Heuristics: #9, #10, #12, #16, #17, #20

Suggestions to fix: Using control metaphors similar to those used in Google Maps would enhance skills transfer from other mapping application. Adding a plus or minus sign at the limits of the slider would make its actions clearer. When the “Max Extent” icon is pressed, it would be better to increase zoom out to best fit the view of the map into the available space, wherever possible, centring the map.

7. Inconsistent pointer actions

Context: Task 1 – No interaction was performed with the application. Identification of the problem was performed based on initial observation of the user interface.



Description: In each map, the pointer actions are mutually exclusive (i.e., swipe, zoom, select) but this is not made explicit via the user interface. A user can activate the actions select, swipe and zoom simultaneously although only one is executed. In addition, the actions are not logically organised.

Severity: 3

Breached Heuristics: #7, #9, #10, #12, #16, #17, #20

Suggestions to fix: Group actions (i.e., swipe, zoom, select) over the map toolbar and remove the select action from the feature selection group. In this way, a toolbox with select, swipe and zoom could be used to exclusively allow only one of the operations to be used.

8. Inconsistent Swipe Icon

Context: Task 1 – No interaction was performed with the application. Identification of the problem was performed based on initial observation of the user interface.



Description: The swipe function on the toolbar over the map does not include an icon. This makes its representation inconsistent with other functions and does not clearly visually distinguish it from the “Graticule” function.

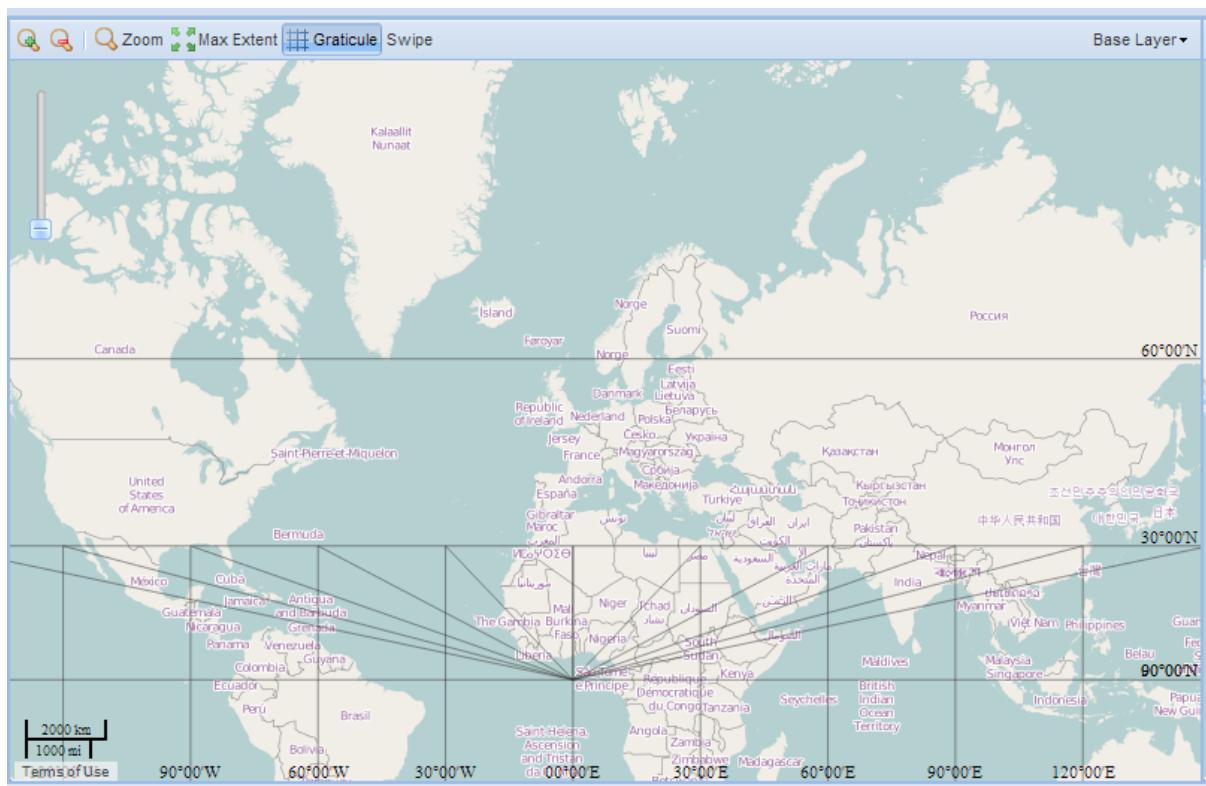
Severity: 2

Breached Heuristics: #8, #9, #16, #17

Suggestions to fix: Add an icon to the function representation. The name of the function should also be considered, possibly via a process of card sorting with target end users to ensure that functional labelling and associated grouping in the UI is optimised.

9. Graticule representation inconsistent

Context: Task 1 – The map controls were used to zoom out. Identification of the problem was performed based on initial observation of the user interface.



Description: A grid is the expected representation after invoking the graticule function. If, however, the user selects the furthest zoom out boundary, the representation is not a grid. Instead, the top part of the map only has horizontal guidelines and the bottom part has scope to further confuse the user because a third dimension seems to be represented.

Severity: 2

Breached Heuristics: #9, #13

Suggestions to fix: Provide a consistent grid representation across the different zoom levels.

10. Swipe representation inconsistent

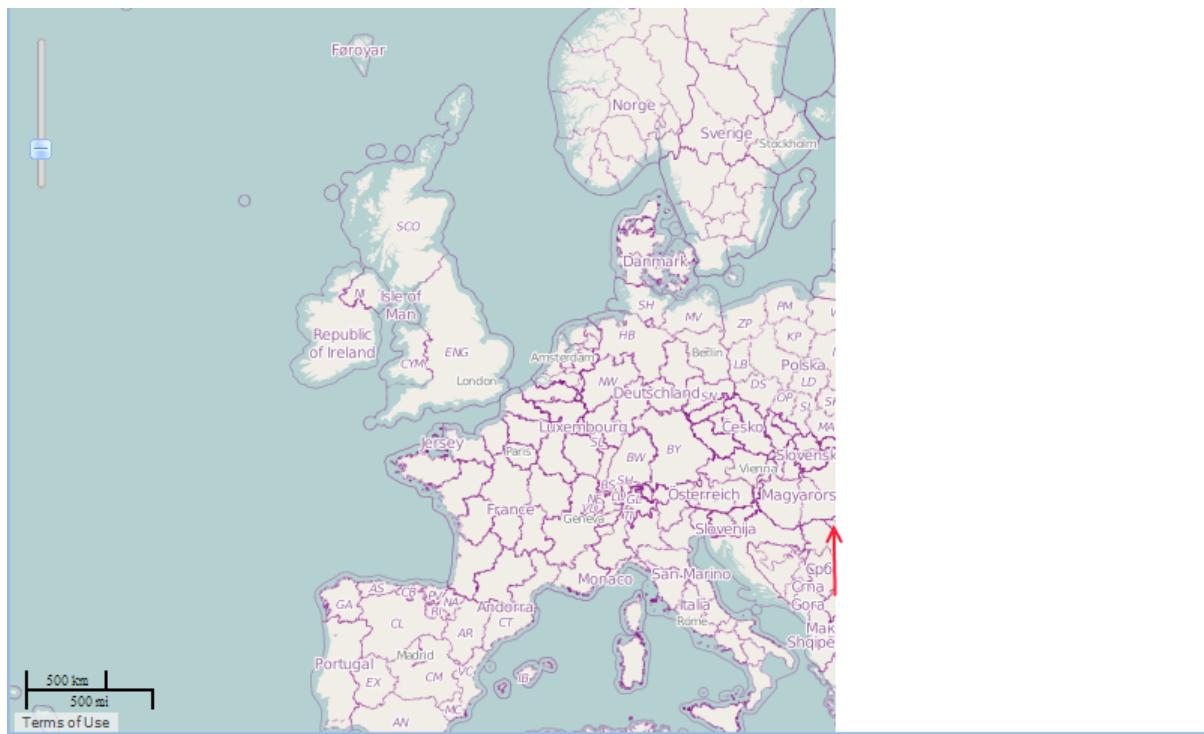
Context: Task 1 – Operation performed with and without adding a layer resource and clicking near a map border.

Description: When the swipe operation is selected, the layered resource or the map (when no layer is loaded) is cleared from the nearest point in the border up to the mouse position. Nevertheless, that model is not consistent as illustrated in the screenshot below. The red arrow shows the mouse click and the movement performed. There are positions where the swipe operation does not start from the nearest map boundary.

Severity: 2

Breached Heuristics: #9, #14

Suggestions to fix: Adjust the calculations to force the swipe operation to start from the nearest map boundary as expected.



11. Resource name not descriptive

Context: Task 2.a) – Add resources and visualise them using adjacent maps.

Add Resource

Resources

- <http://giv-uw.uni-muenster.de/vis/v2/data/raster/bi...>
- [data/json/austalObs.json](#)
- [data/json/gaussian.json](#)
- [http://giv-uw.uni-muenster.de/data/om/cropallocati...](#)
- [http://giv-uw.uni-muenster.de/data/om/exposure_F...](#)
- http://giv-uw.uni-muenster.de/data/netcdf/aqms_20...
- [http://giv-uw.uni-muenster.de/data/om/UBA_2010-...](#)
- <http://giv-uw.uni-muenster.de/vis/v2/data/raster/pr...>

Details

Add Resource

URL:

Type:

[Advanced...](#) [Add](#) [Add to Map](#)

Description: When a user tries to select a resource, its name is not completely visible meaning that relevant parts may be hidden and cause either more effort on the users' part or potential for error if the user mistakenly selects the first of a series of files with a similar prefix.

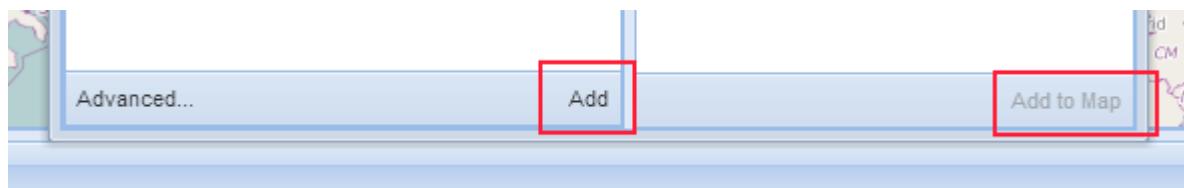
Severity: 2

Breached Heuristics: #9, #10, #12

Suggestions to fix: Several solutions are possible, including a) increasing the width of the panel to fit the longest resource name, and/or b) hiding the common part of the resource name (i.e., protocol and server name) and, instead, showing the last, unique, part of the resource name. As an example, a resource named “<http://giv-uw.uni-muenster.de/vis/v2/data/json/gaussian.json>” could be visualised as “[...]/vis/v2/data/json/gaussian.json”. A click on the “[...]” could allow the user to expand into the full name if necessary.

12. Add a resource

Context: Task 2.a) – Add resources and visualise them using adjacent maps.



Description: A left and right panel are presented to the user in the dialog to add a resource into the map, at the bottom of which are two similarly named buttons (i.e., Add and Add to Map). It may not be clear which button to use to add a new resource.

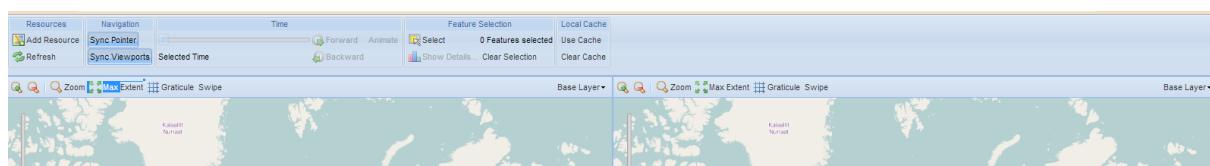
Severity: 3

Breached Heuristics: #9, #12, #14, #17

Suggestions to fix: The option to add a new resource to the list of available resources could be hidden and displayed on request (progressive disclosure). The option could even be displayed in a new dialog to avoid confusion between adding a resource to the map and a new resource to the list of available resources.

13. Add a resource to the map on the right

Context: Task 2.a) – Add resources and visualise them using adjacent maps.



Description: Adding a resource to the left map is intuitive but adding a resource to the map on the right is not clear. The evaluator was not able to perform the task.

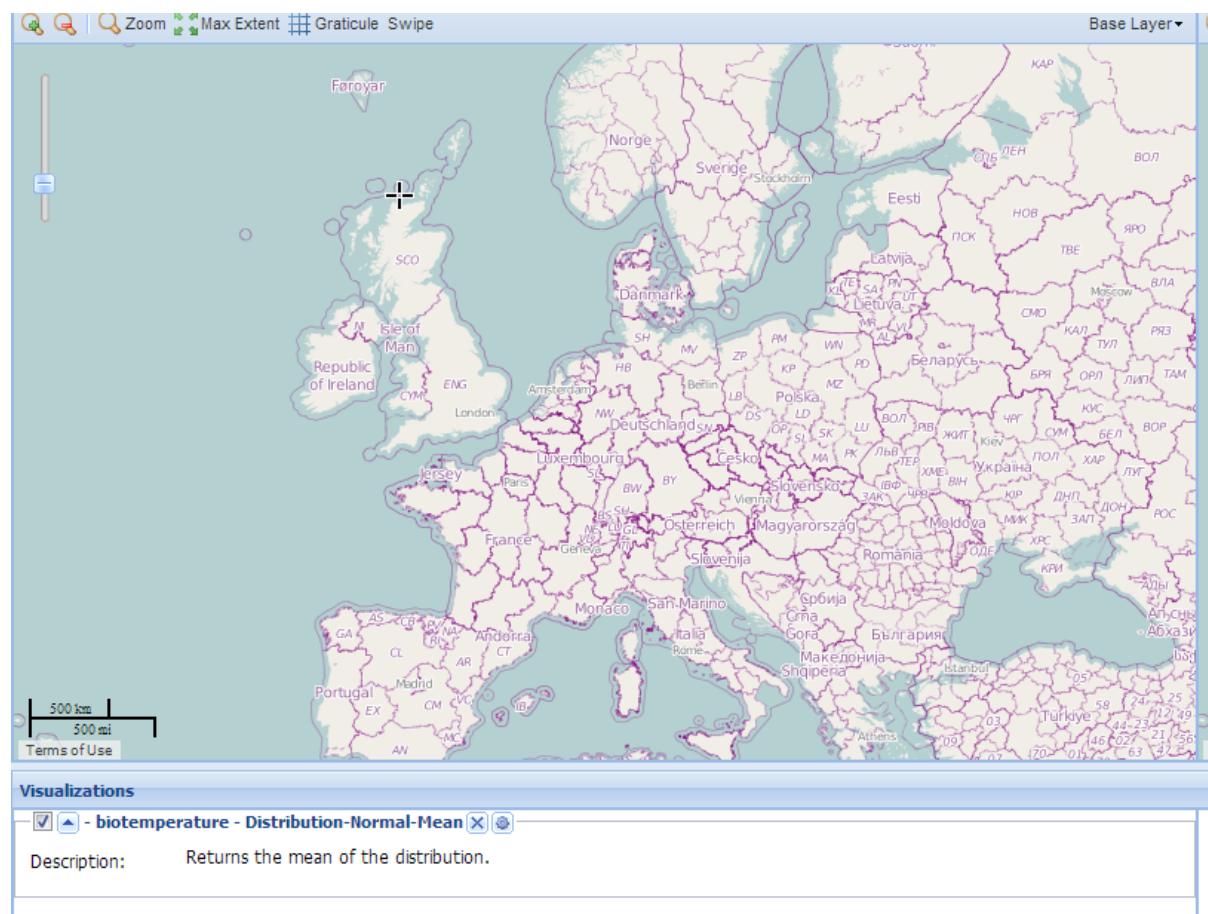
Severity: 4

Breached Heuristics: #2, #4, #9, #10, #14

Suggestions to fix: The top toolbar could be extended to also include an add resource group to the right map. The ability to add resources to the right map should be clear as it is, potentially, one of the most common operations.

14. Provide feedback while loading a resource

Context: Task 2.a) – Add resources and visualise them using adjacent maps.



Description: While a resource is being loaded, no feedback is provided to the user. The image shows a resource that was selected and is loading but no feedback is provided to the user whilst this is happening to allow the user to perceive the status of the system and obtain feedback on his actions.

Severity: 3

Breached Heuristics: #4, #9, #13, #17

Suggestions to fix: Add feedback within the map space as the resource is being loaded – for instance, a loading animation could be added to the map and visualisation panel.

15. Provide feedback when a resource is not loaded/unavailable

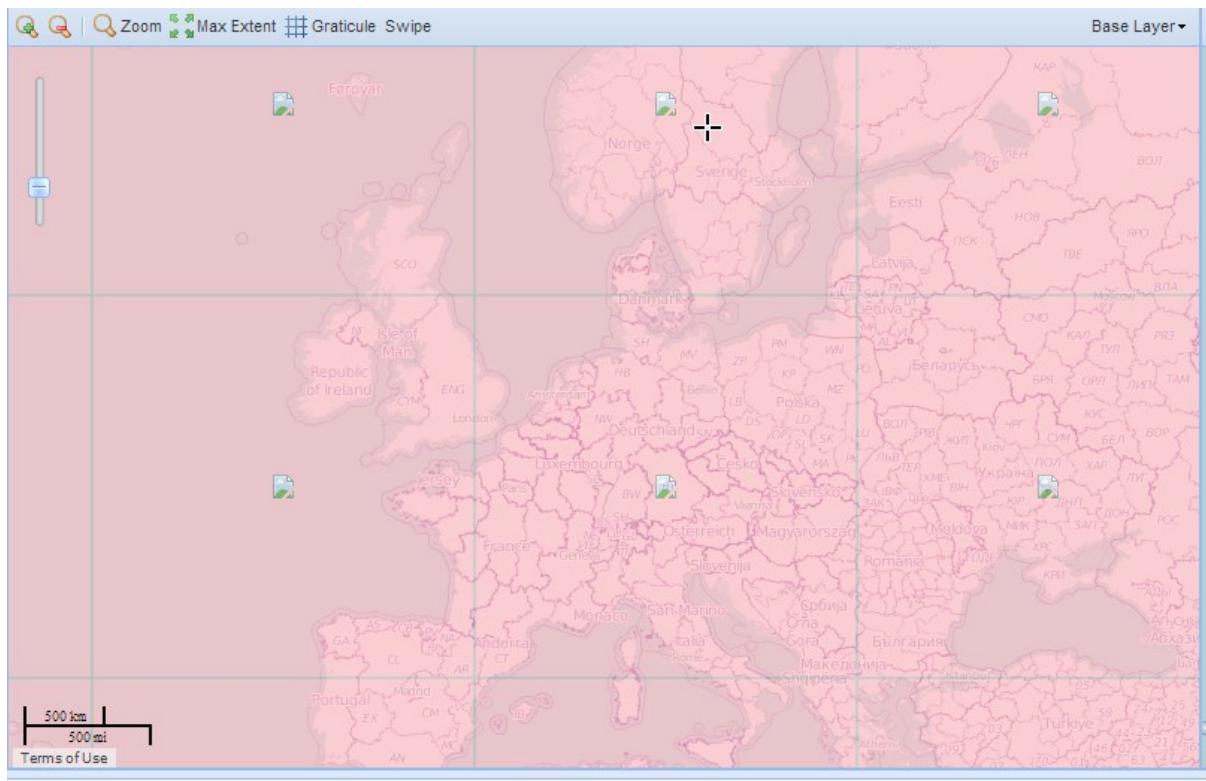
Context: Task 2.a) – Add resources and visualise them using adjacent maps.

Description: Due to a slow network, communication error or unavailability of a resource, problems can occur upon resource loading which results in failure to load; appropriate feedback should be provided to the user.

Severity: 3

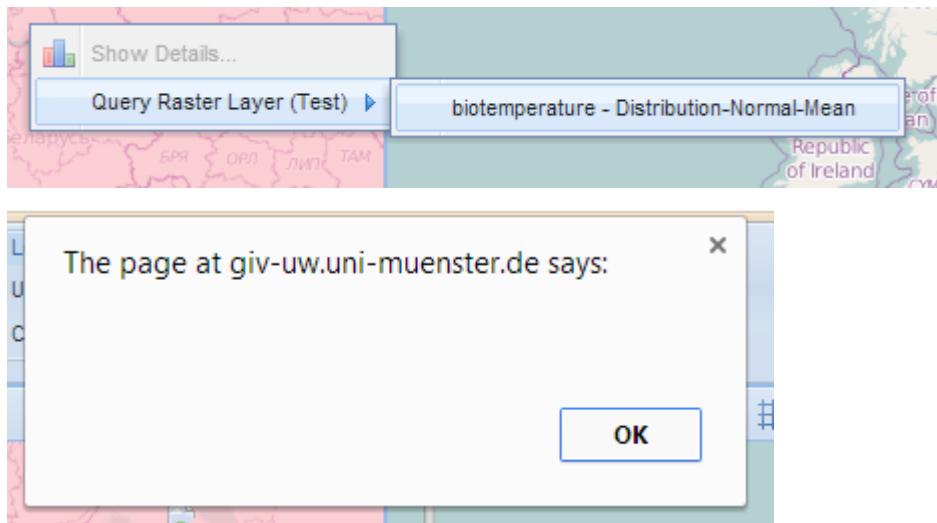
Breached Heuristics: #21

Suggestions to fix: Use status messages to provide feedback to the user. If a resource cannot be loaded, the user should be provided with adequate and informative feedback.



16. Error messages are inconsistent

Context: Task 2.a) – Add resources and visualise them using adjacent maps.



Description: When a user clicks the right mouse button on top of the map with a loaded resource (raster), an error dialogue box is displayed but contains not actual error message. The first figure shows the operation performed on top of the map and the second figure shows the error dialogue displayed.

Severity: 3

Breached Heuristics: #5, #9, #10, #13, #21

Suggestions to fix: Error messages should be descriptive and should provide hints for the user to recover from the error. Ideally, wherever possible, prevent the user from executing the action that leads to the error in the first place.

17. Improve visualization icons

Context: Task 2.a) – Add resources and visualise them using adjacent maps.



Description: Although the icon represents a gear/cog (associated with settings), when visualised on the application at the scale at which it is provided, it can be perceived as a circle meaning that the actual functionality of the icon is not clearly communicated.

Severity: 2

Breached Heuristics: #9, #10, #16, #17

Suggestions to fix: Enhance representation of icons, paying particular attention to the size at which each will be displayed, making sure that all icons work at reduced resolution and size. Use of shadows to create a 3D effect for small icons may make it hard for the user to perceive what they represent and therefore their meaning.

18. Clearly state the purpose of interactive elements

Context: Task 2.b) – Visualise data with temporal dimension.



Description: When the user adds a resource (e.g., austalObs.json), a clickable parameter is generated but clicking this UI element does not trigger any operation or enable any interactivity and so the purpose of the component/action is not clear.

Severity: 2

Breached Heuristics: #5, #9, #10, #13, #19

Suggestions to fix: Clearly state the purpose of interactive elements. For instance, hints for the purpose of the interactions could be added or the parameter hidden from the user if no interaction is possible.

19. Standard conflict/Unclear purpose of interactive element

Context: Task 2.b) – Visualise data with temporal dimension.



Description: When a temporal resource is loaded, a “best-fit” icon appears in the visualisation panel related to the resource. This icon allows the application to adjust the zoom level to include all the collected data in the window. The icon representation is, however, similar to that of the window maximize operation and so has the potential to confuse the user.

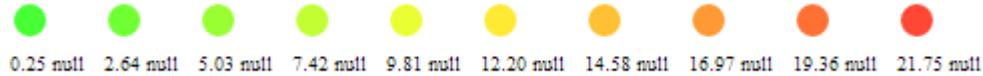
Severity: 2

Breached Heuristics: #9, #10, #16, #17

Suggestions to fix: The icon should clearly and distinguishably convey a best fit operation. Consider, for example, an approach similar to the one used on office applications to adjust/fit content within a window or page.

20. Unidentified units

Context: Task 2.b) – Visualise data with temporal dimension.



Description: When a temporal resource has loaded, units may be required to represent the values on legends but such units may be missing in the UI.

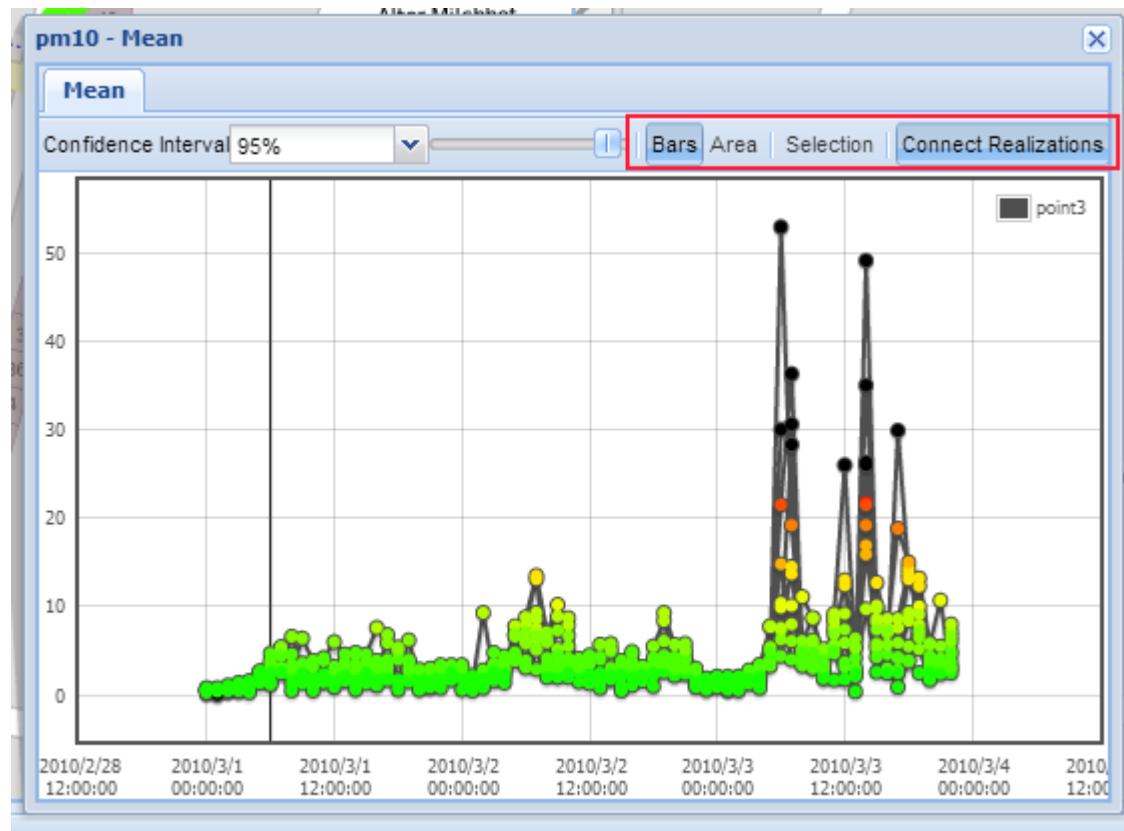
Severity: 3

Breached Heuristics: #5, #7, #21

Suggestions to fix: Units provide meaning for the information displayed. Revise the original data sources and/or provide a status message describing the error where no such units are available for display.

21. Functions are not logically grouped

Context: Task 2.b) – Visualise data with temporal dimension.



Description: When a user clicks a feature on the map and visualises its related time-series graph, the available functions lack a logical visual grouping which would help the user to understand their purpose. It is not clear that the Bars and Area buttons can change the representation of the graph's

boundaries. Furthermore, in the example of the graph used for this task, the information is not used and so the buttons can be disabled to better reflect status.

Severity: 3

Breached Heuristics: #2, #9, #10, #14, #20

Suggestions to fix: In this graph, the Bars and Area button can be disabled since they are not available functions. Furthermore, the selection actions could be better visually and more logically grouped.

22. Inform the user of new interaction techniques

Context: Task 2.b) – Visualise data with temporal dimension.

Description: When a user clicks a feature and visualises its related graph, available interaction techniques (i.e., pan, zoom or selection) are not clearly visible. It is also unclear that clicking the select button can change the set of available interaction techniques. Without the select button, the user can pan the graph and perform zoom operations with the mouse wheel, but there is no feedback to the user on the status of such operations.

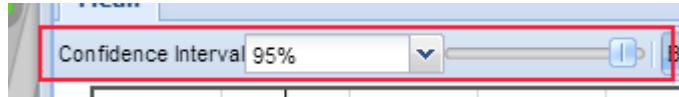
Severity: 3

Breached Heuristics: #2, #9, #10, #13, #15, #17

Suggestions to fix: Hints should be provided to the user on how to use the UI, possibly using the same metaphor as the one to navigate on a map. This would enable navigation to be immediately visible as well as zooming.

23. Logically group controls to handle confidence interval

Context: Task 2.b) – Visualise data with temporal dimension.



Description: When a user clicks a feature and visualises its graph, the controls to define the confidence interval are not logically ordered. A label, a combo box and a slider are used to define the confidence interval. In a standard approach, a slider is typically used between a label and a text box which is not the visual presentation sequence used here.

Severity: 2

Breached Heuristics: #8, #9, #10, #12

Suggestions to fix: Use only a combo box, or use a slider with a text box for free text entry (with appropriate entry format controls). When using a slider, graded steps could be included for the most commonly used values. Finally, the controls should be logically ordered (i.e., label, slider and, finally, text box).

24. Misspelled resource type

Context: Task 2.b) – Visualise data with temporal dimension.

URL:

Type: ▼

Advanced Add

Description: When a JSON resource is added to the list of resources the type of the resource is misspelled.

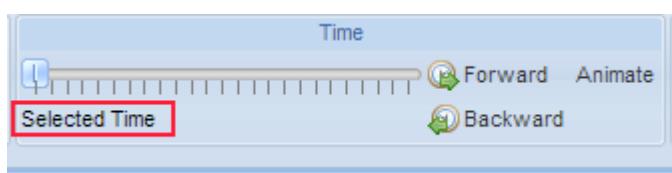
Severity: 1

Breached Heuristics: #5

Suggestions to fix: Change the JSOM label to JSON.

25. Update time information when a time related resource is loaded

Context: Task 2.b) – Visualise data with temporal dimension.



Description: When a time-related resource is loaded, the slider bar is updated with the relevant time periods intervals but the description of the selected time is not equivalently updated.

Severity: 3

Breached Heuristics: #4, #10, #13

Suggestions to fix: Update the label when a time-related resource is loaded.

26. Inconsistent threshold input

Context: Task 3 – Multidimensional interaction.

Threshold for

Description: While adding the resource gaussian.json (Exceedance Probability, threshold 10), a parameter needs to be entered (i.e., threshold). For that purpose, a text box and a button are provided. The “OK” button is not, however, used and no common (and therefore indicative) values for the threshold are provided.

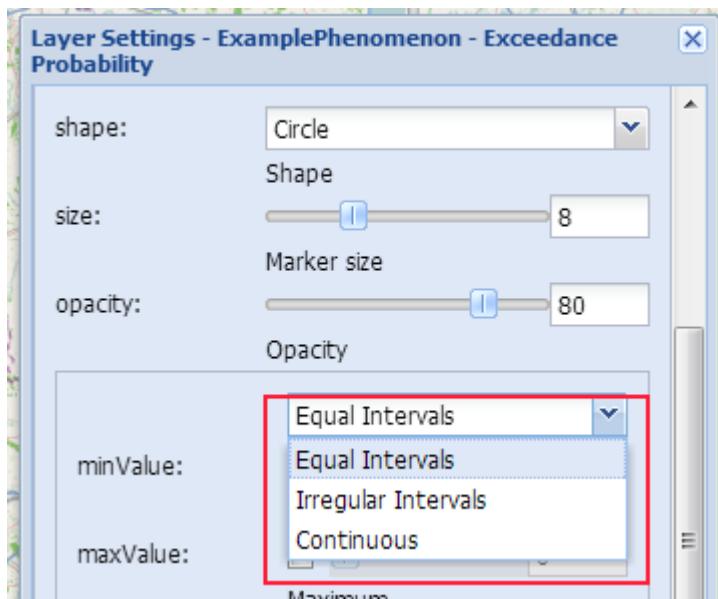
Severity: 2

Breached Heuristics: #12, #14

Suggestions to fix: As in usability problem 23, a more standard input arrangement could be used – e.g., consider using a slider and a text box, or a combo box, if consistent with other approaches.

27. Exceedance intervals are not available

Context: Task 3 – Multidimensional interaction.



Description: When changing the exceedance intervals through the layer settings, the option that should be selected is not available.

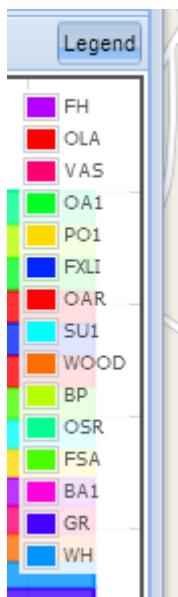
Severity: 4

Breached Heuristics: #5

Suggestions to fix: Unknown. It may be a software bug or a change in the remote information used.

28. Legend description is not clear

Context: Task 4 – Categorical Data



Description: During visualization of a time-series graph, the user should be supported in terms of interpreting and understanding what is being presented. In this graph, the legend does not help the user to understand the types of data that are available.

Severity: 2

Breached Heuristics: #5

Suggestions to fix: Provide more descriptive text about the information that the user is seeing (e.g., FH does not clearly describe what is represented).

29. Feature selection does not work properly

Context: Task 4 – Categorical Data



Description: When the user clicks on the “select” button (in the toolbar, feature selection group), and starts to select a group of features, the selection procedure is interrupted if the user moves the pointer over a feature on the map. Consequently, information about the feature is displayed and the selection operation cancelled. The example above shows a cancelled selection operation where information about a feature was presented, once the pointer was on top of the feature.

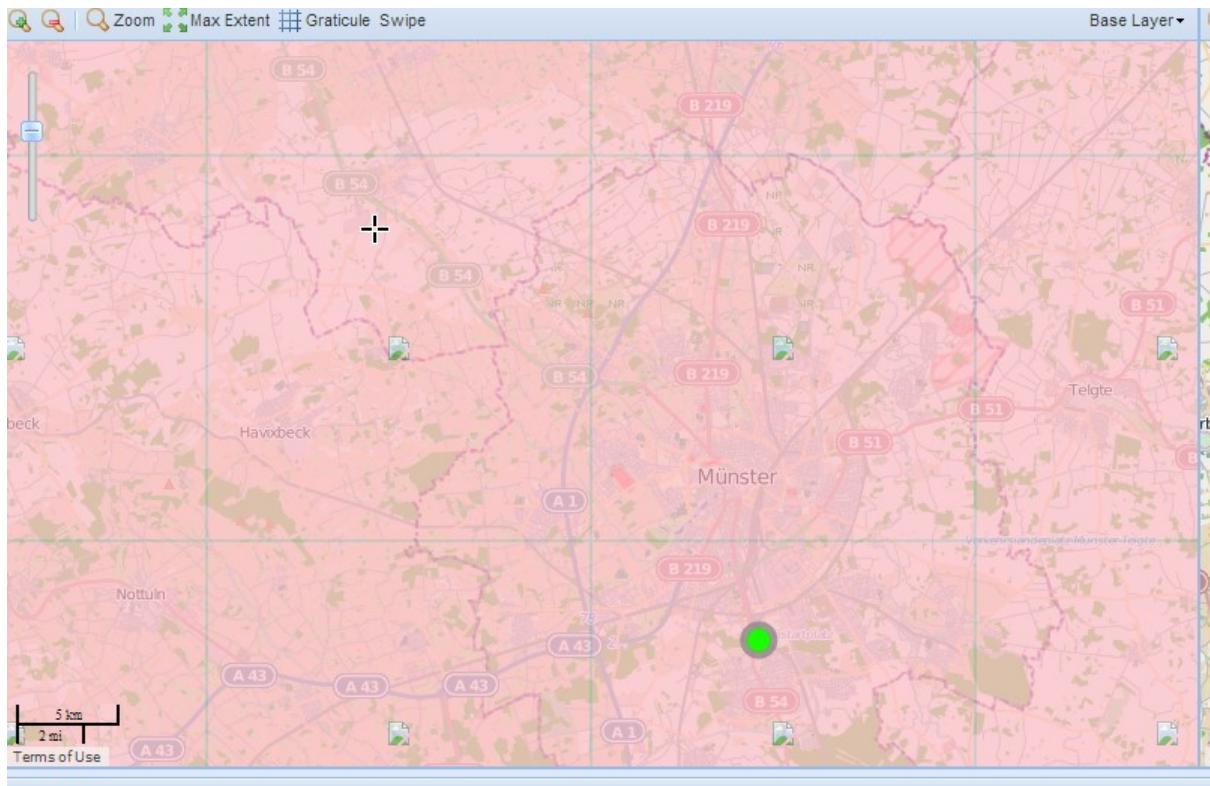
Severity: 3

Breached Heuristics: #9, #14, #17

Suggestions to fix: When a selection operation is in progress, other operations should not distract the user or cancel the current operation. The user should be able to complete the selection operation unimpeded given that the user has explicitly prioritised that action.

30. Not able to load NetCDF data

Context: Task 5 - Combined visualisations



Description: The task requires adding an external resource⁴. After adding the resource, a sample numbered 9 is also required to be selected and the resource added to the map but no information is rendered. The figure shows the result of unsuccessfully attempting to load a NetCDF resource.

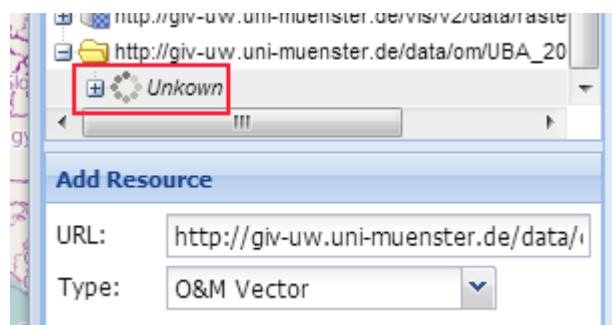
Severity: 4

Breached Heuristics: #21

Suggestions to fix: Whether the resource was corrupted or unavailable, the solution includes fixing the service or the application.

31. Unable to load O&M resource

Context: Task 6 – Data preparation



Description: The task requires adding an external resource⁵. After adding the resource, no information was rendered but the JSON conversion of the same file was loaded. The figure above shows the result of unsuccessfully loading an O&M resource.

Severity: 4

Breached Heuristics: #21

Suggestions to fix: Whether the resource was corrupted or unavailable, the solution includes fixing the service or the application.

32. Unable to load a raster resource

Context: Task 6 – Data preparation

Description: The task requires adding an external resource⁶. After adding the resource, no information was rendered.

Severity: 4

Breached Heuristics: #21

Suggestions to fix: Whether the resource was corrupted or unavailable, the solution includes fixing the service or the application.

⁴ http://giv-uw.uni-muenster.de/data/netcdf/aqms_2010-03-30.nc

⁵ http://giv-uw.uni-muenster.de/data/om/UBA_2010-01.xml

⁶ <http://giv-wikis.uni-muenster.de/agp/pub/Main/UncertaintyVisualisationWorkshop/jrcuprec.tif>

Discussion

As mentioned previously, the majority of the usability problems discovered in the application are related to interactivity issues. Mostly, the application fails to make it clear how interaction should be performed and labels or icons are inconsistent or not clearly presented (e.g., interactive elements with different sizes, with or without labels, or lack of clear navigation controls). The user interface could benefit from inclusion of metaphors and interaction techniques with which users are accustomed (e.g., Google's map navigation controls), to profit from skills transfer into the Greenland application. Indeed, many of the base layers already available are from Google, so it would be reasonable to assume that users will already be familiar with such metaphors and interaction techniques.

Another aspect requiring consideration relates to the lack of visibility of available actions/functions. For example, when the user is visualising a time-series graph, it is not clear that zooming and panning functions are available (see usability problem 22). At all times, it should be clear what the user is able to do and how to get further information about the elements (i.e., about features) being visualised. With regard to the latter, for instance, it is not clear that further details can be viewed when the user clicks a feature. The time-series graph also has identified problems regarding visible command buttons that do not convey a logical order – for example, “Area”, “Selection” and “Connect Realizations” buttons are side-by-side but lack presentation of associated meaning.

In terms of the severity of identified usability problems, a number of *catastrophic* problems were identified that prevented a more extensive evaluation of the application, including of:

- adjacent maps;
- temporal graphs;
- adding resources; and
- comprehension of visualisations.

Adjacent maps were not evaluated due to the unavailability (or unclear method) to load/add resources on the right-hand map. Temporal graphs were evaluated but catastrophic usability problems were identified while trying to load required resources (e.g., units were not loaded and some resource types were not loaded). Inability to load raster information and other data types compromised the ability to provide a comprehensive evaluation of visualisations.

References

- [1] Petrie, H., Power, C. (2012) *What do users really care about? A comparison of usability problems found by users and experts on highly interactive websites*. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12). ACM, New York, NY, USA, 2107-2116.
- [2] Molich, R., and Nielsen, J. (1990). *Improving a human-computer dialogue*, Communications of the ACM 33 , 3 (March), 338-348.