

Interface requirements for comparison search application

Author: Alia

December 18, 2008

Note to Lynda: this document explains two things:

- requirements for the comparison search interface
 - design rationale for the components of the interface (this part has no research value but written as per your request, consist of design consideration that leads to the interface look and feel). Information about this is at the 'Interface Design' section.
 - if you have any question but its not in this document, its probably because i didn't think its worth mentioning or no research value. Let me know if you want more information on a particular design decision.
 - designing an interface is not math so there are no strict rules in what components to put where. Yes it is somewhat an art (like Lynda mentioned.) so there are no exact rules that apply here. The best interface fulfills same criteria: lets the users work efficiently, in terms of speed, cognitive load, ease of use, etc. The interpretation of this criteria and balancing between form and function is up to the designer.
- Design rules are not a focus of this research so I will not spend time on trying to rationalize the detail design decision too much.

1 Use cases

There are 3 comparison search use cases:

1. *Compare two sets* e.g. two museum collections. What are the similarities and differences between the East-Asian collection from the Rijksmuseum Volkenkunde Leiden and the East-Asian collection from the Tropenmuseum Amsterdam?
This use case is the generic use case. Other use cases are a special case of this use case.
2. *Compare two items* e.g. two artworks. What is the difference between the painting "Vase with twelve Sunflowers" by Van Gogh stored at Neue Pinakothek, Munich and the painting "Vase with twelve flowers" by the same artist on displayed at Philadelphia Museum of Art, Philadelphia?
3. *Compare one item against a set* e.g. What are the **similarities and differences** between "The Sunflower" paintings by Van Gogh and his other paintings?

2 Functional Requirements

This section provides a list of functional requirements of a comparison search interface.

1. enable object comparison and set comparison use cases.
2. the interface should enable four functions: (a) *object selection*, (b) *set refinement*, (c) *object comparison - view mode*, (d) *object comparison - exploration mode*. At any state, user should be able to go back and make changes in the

selection easily.

- (a) object selection: enable query for objects and store in a selection 'container'. A container serves as a place to temporary store all objects which are interesting to the user before he compares them. It saves time from doing repeated search.
- (b) object comparison - view mode: see all properties and values of objects. Similar¹ properties and/or values are grouped.
- (c) object comparison - exploration mode: enable to change constraints, change focus on a certain property or value and see results.

3. for set comparison, merge property information together in a visualization.
4. comparison of objects that has physical properties(location, size, color) are better with visualization (images, map)

nice to have: a save object(s) selection. and compare multiple sets (not only 2)

General design requirements:

- **GR1** Tasks which are related to each other should be presented in the same time and the same visual space.
- **GR2** Tasks should be clear and the process between task should be obvious.
- **GR3** The most important task should be placed in center of the working area.
- **GR4** Important (sub)task should be highlighted.

Constraints

1. the properties used in the research will be manually filtered
2. there will be a restricted number of properties allowed (e.g. less then 50)
3. the interfaces will only use properties common to both sets

3 Interface Design

Presentation of properties and values of different object(s) in the set(s) are the key of comparison search interface. Attention will be given to:

1. how to present (sub)properties to best support comparison search
2. how to present values and its numbers to best support comparison search.

Interface

The comparison interface is divided into three task areas: 'Search', 'Search Result', and 'Compare' panel. See Fig.1.

¹the notion of 'similar' will be defined later

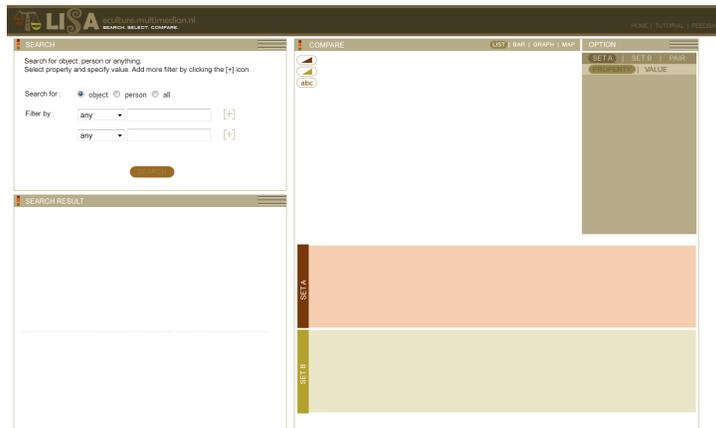


Figure 1: Initial state. 3 different panels: Search, Search Result and Compare panel.

'Search' and 'Search Result' should be in the same interface because both tasks are closely related to each other. 'Search Result' and 'Compare' should be in the same interface because users need to be able to select items from 'Search Result' and put them in the 'Selection' part of the 'Compare' panel. Because of these dependencies, all three panels are displayed in one interface (see GR1).

Searching a set from the repository

The user performs an 'advanced' search, e.g. search for all art objects from the Rijksmuseum collection that is made by Rembrandt van Rijn.

The 'advanced' search functionality should let users specify the exact set that he is looking for i.e. do not rely on any clustering mechanism of the search result.

After from the 'Search Result' panel, the user drags and drops items to the selection area, either Set A or Set B (see Fig.2). The action is repeated until both sets are defined (see Fig.3).

Visualization

In the center of the interface there is a 'Visualization' panel and an 'Option' panel. The visualization panel have 4 variations: list, bar, map, and graph.

3.1 List

A list is a basic row-to-row comparison of different property. It is the baseline visualization that is well known in any comparison search task.

3.2 Map

The map visualization shows all location values of a specific property. To think in the future: also non-direct location values e.g. culture 4. This visualization is useful for location specific values.

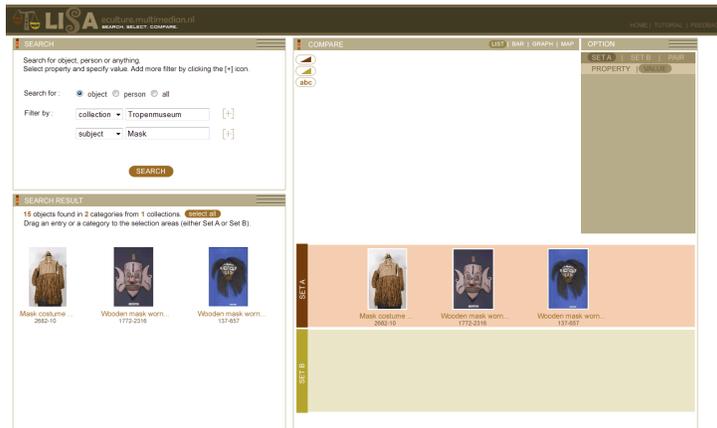


Figure 2: Searching and defining a set.

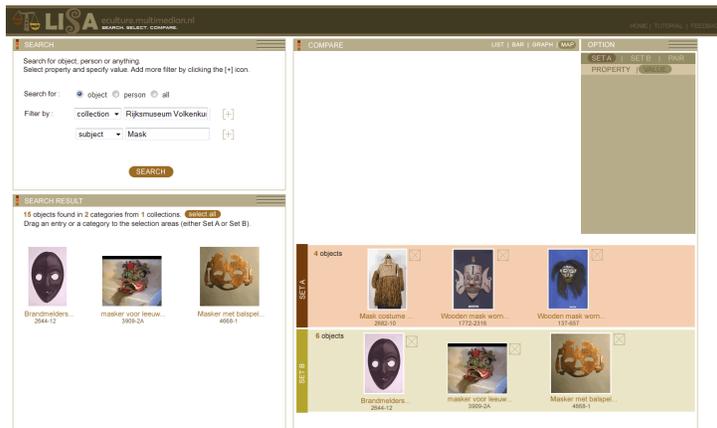


Figure 3: Set A and set B defined.

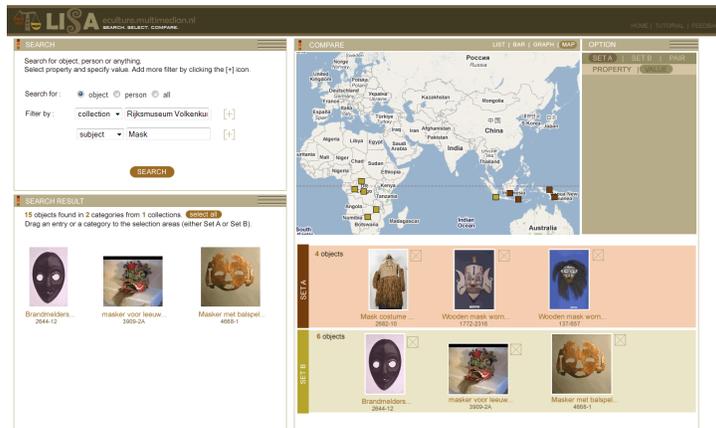


Figure 4: Map visualization.

3.3 Bar

Bar displays all values, ordered either by alphabetical or by amount (large to small). The Bar has a 'property carousel' where users can scroll and select a property. When a property is selected, a corresponding values of all properties from the two sets are display 5. This interface is very useful for detail value distribution analysis.

3.4 Graph

The graph visualization presents all values of both sets for two user-defined property (see drop down selection). X-axis and Y-axis can be dynamically configured (see fig 6). This interface is useful for comparing value distribution over two different properties.

3.5 Option: semantic manipulation

The option panel is used to semantically manipulate property and values for visualization. It consist of 3 features:

1. property manipulation: show/hide property; show/hide subproperty. see fig 7
2. value manipulation: merge/collapse values. see fig 8910
3. property pair: aligning different property for comparison. see fig 11

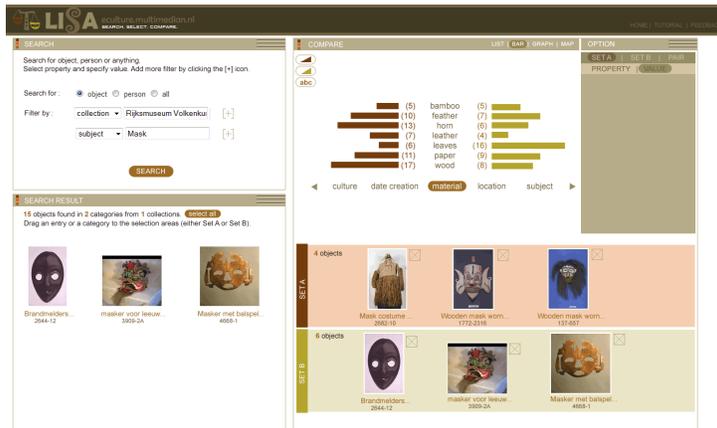


Figure 5: Bar visualization.

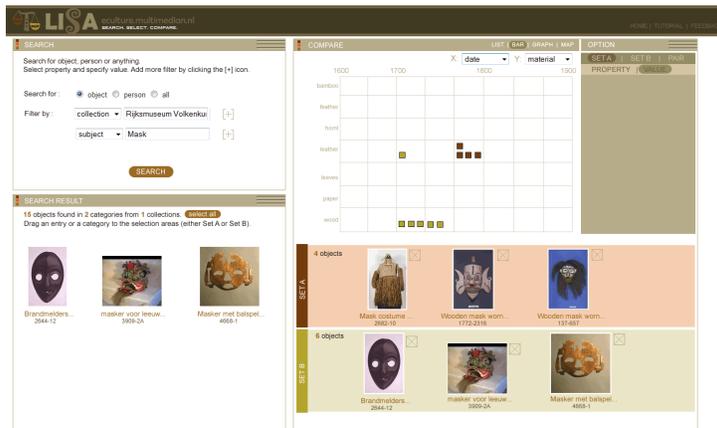


Figure 6: Graph visualization.

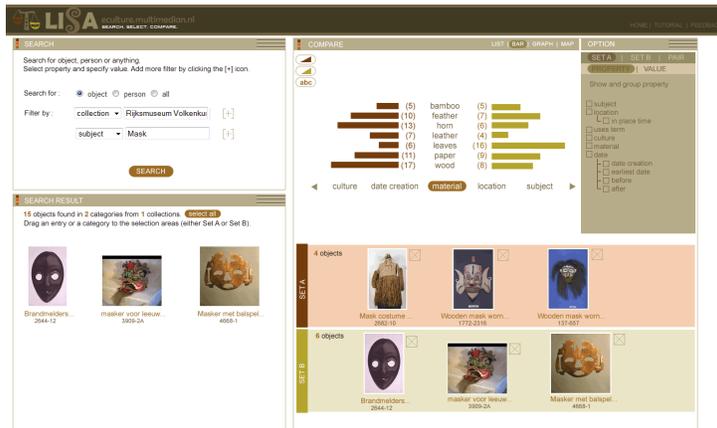


Figure 7: Manipulating properties of a set

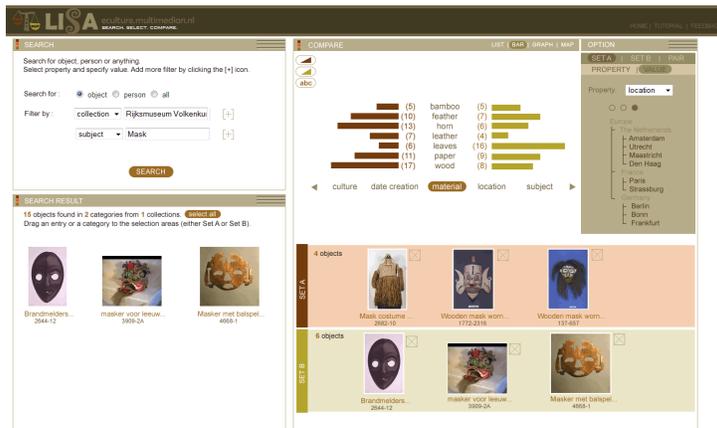


Figure 8: Merge and collapse values of a property within 1 set

The screenshot displays the LISA interface with the following components:

- SEARCH:** Search for object, person or anything. Select property and specify value. Add more filter by clicking the [+] icon. Search for: object person all. Filter by: collection [+], subject [+].
- SEARCH RESULT:** 15 objects found in 2 categories from 1 collection. Drag an entry or a category to the selection area (either Set A or Set B). Three mask images are shown: Bransmeesters (264-12), masker voor leeuw (393-2A), and Masker met balspel (488-1).
- COMPARE:** A horizontal bar chart comparing properties between two sets. Properties include culture, date creation, material, location, and subject. The 'material' property is highlighted.
- OPTION:** SET A: in place time, SET B: location. A PAIR button is visible.
- Object Selection:** SET A contains 4 objects: Mask costume (2062-10), Wooden mask worm (1775-219), and Wooden mask worm (197467). SET B contains 3 objects: Bransmeesters (264-12), masker voor leeuw (393-2A), and Masker met balspel (488-1).

Figure 11: Pair properties from 2 sets