

Functional specifications

Index

- Overview
- Query formulation interface (QFI)
- Browser
- Viewer
- AnnotationEditor
- GUI

Overview

This documents describes how the IVB will look and feel from the *user interface* point of view. Developers should also refer to the [technical documentation](#). Not everything described in this document is or will be implemented: user feedback will play a role in shaping the end result.

Importance of good labels, and user not exposed to uuids.

The user interface is organized through tabs (much like any modern browsers).

Query formulation interface (QFI)

The search consist of three main parts: input text (not implemented at least in the early versions of IVB), entity selection and property selection. We just need at least one of them to be specified, and each one refine the query (AND). See following illustration for common terminology.

keyword

statue

category things actors places events concepts

relationship term

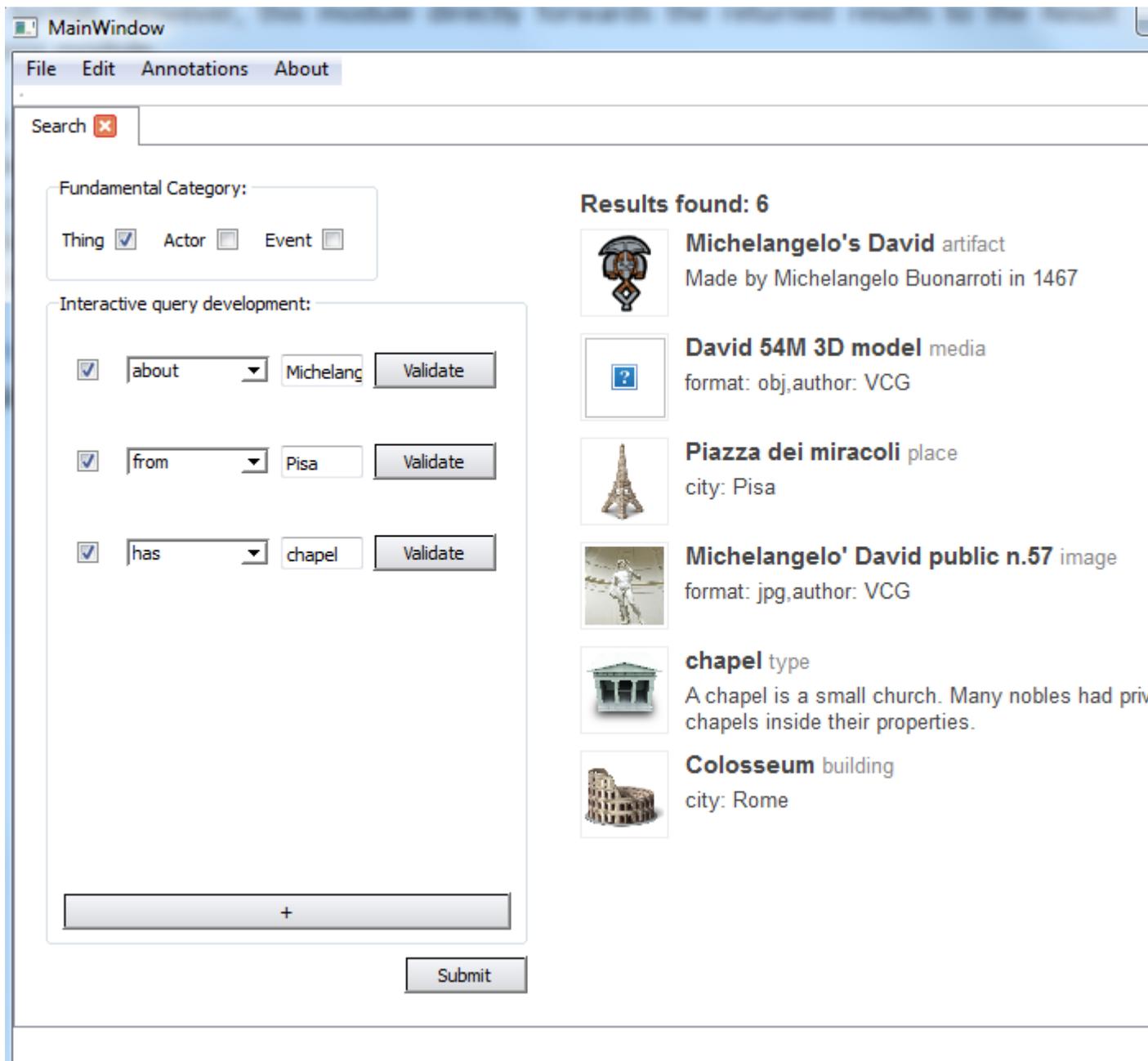
condition is or has wings

from

validated term

greece	
Greece	[Place]
Ancient Greece	[Place]
Roman Greece	[Place]
Ancient Greece	[Time]
Roman Greece	[Time]
Greece, New York	[Place]

First implementation



Text input

The input part is a simple text input. The keywords entered here are matched against uri and appellations of entities. The exact query depends on the class selected into the entity selection.

In a later stage the MSL might suggest alternatives to the user: normalizations, translations, alternatives.

In the first release of the IVB this input will not be implemented, on the ground that probably users will be able to specify their queries using the other fields in the QFI.

Category selection

Allows the user to specify what classes of entities he is looking for: people, events, places etc. It is possible to specify more than one category.

In the initial release of the IVB only fundamental categories will be used. Subcategories such as "digital object" or "3d model" might be added after user feedback.

Conditions

Allows to refine a query specifying conditions for the result such as being created in a certain period, coming from a certain place etc.

Each condition is composed by 2 parts: a combo box of relationships (is located in, has met, is from period, etc), and a term formulation aid, which allows the user select precisely an entity (a material, a period, a location, a style).

Selecting categories constrains the relationships into a subset, while selecting the relationship usually constrain the term to be of one or few classes: e.g. selecting the relationship *is located in* forces the term to be a place.

Before a relationship can be used the term must be validated: the user writes "napoleon" in the term input, the interface present the users all the possible entities which "napoleon" could refer to:

- Napoleon I (Napoleon Bonaparte; 1769–1821) was a French military leader and emperor.
- Napoleon II (1811–1832), son of Napoleon I
- Napoleon III (1808–1873), last monarch and first president of France, son of Napoleon I's brother Louis
- Napoleon Charles Bonaparte (1802–1807), prince of Holland and son of Napoleon I's brother Louis
- Napoleon Louis Bonaparte (1804–1831), King of Holland and son of Napoleon I's brother Louis
- Napoleon Joseph Charles Paul Bonaparte (1822–1891), son of Napoleon I's brother Jerome
- Napoleon Charles Grigoire Jacques Philippe Bonaparte (1839–1899), son of Napoleon I's brother Lucien

and the user select the relevant one.

For material and shapes a model dialog with a custom interface will be displayed.

The user can add an arbitrary number of additional condition and test different combinations enabling and disabling each condition using the checkbox on the left of each one.

It might be possible in a distant future to return also the number of objects matching the subquery, to give feedback to the user.

Feedback

Using an entity from a previous search in a new search. For example search for David, one of the result is Michelangelo, then searching for works by Michelangelo. Or search for 3d-models related to David with keyword Stanford. The entity is used to fill the term in the property selection.

The user can copy an entity in the browse (using the right click menu, for example) and paste it into the term input, or can directly drag and drop it.

Context menu

The term input has a context menu for copy and paste of entities.

Browser

Browser modes

The browser widgets present the results of the query to the user. Three different interfaces has been designed: a list, a matrix of thumbnails and a graph. In the current release only the first is available. As soon as good practices for thumbnails and labels will be created the matrix interface will be made available. The graph interface is expected for next year.

Upon adding additional interfaces a control switch will be placed in the browser window (upper right)

Results are paged (not in the graph display, of course), to a configurable number of items per page.



Context menu

- View...
- Co-reference annotation: select 2 or more object and check authorization, ask for confirmation. If no authorization an annotation is created instead (bring up annotation editor user interface) This needs of course to be discussed with users and FORTH
- View with... for objects where more than 1 viewing plugin is available (for example ply, could have an *open with meshlab* option (see viewer section for discussion)
- Show graph... open a browser window in graph mode, centered on this entity
- Copy
- Send to annotation (adds to the shopping card in the annotation editor)
- Comment annotation... (creates a new annotation of type comment)

Viewer

The viewer present detailed information about an entity. The actual information depends on the category of the entity: the viewer configuration needs to be flexible and adapt to the entity shown: if the entity is a viewable OR object (3d mesh, document image) there will be a main visualization area on the right, and a sidebar displaying some info about the object and annotation and areas; otherwise there will be no visualization area and the informations will take the whole page.

Behaviour for 3d-models, images, etc.

The areas and the annotation related to the entity will be shown in the left part of the tab, while the model, image or text will be displayed in a large area on the right side/>>[?

When the user hovers on an annotation on the tab, all the related areas will be highlighted in the viewer, when the user hovers on an area on the tab, the corresponding area on the viewer plugin will be highlighted.

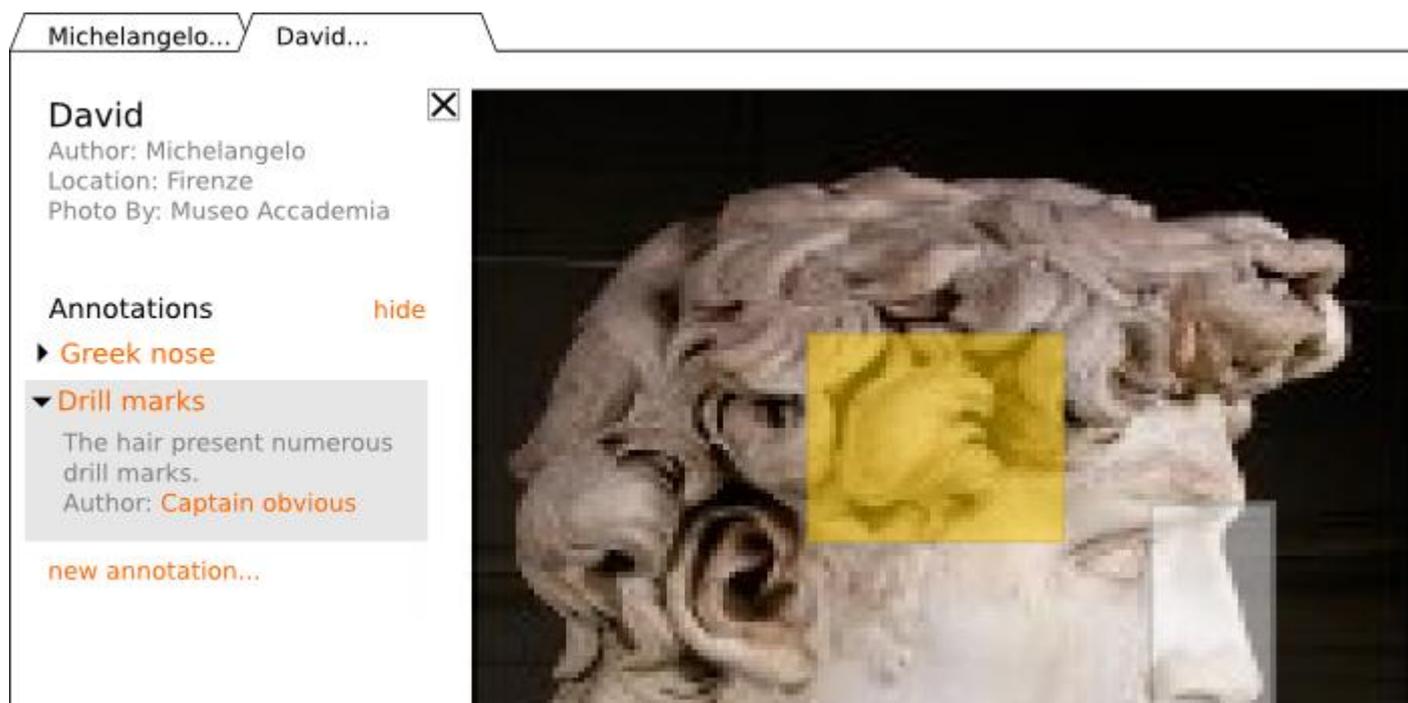
Simmetrically when the user hover on an area on the viewer plugin the annotation referring that area will be highlighted

Editing and creation

To create an additional area the tab into a modal state where the user defines the area in the viewer (or cancel the action). The new area will appear in the area tab, with style suggesting it is local and not saved to the repository

Double clicking an area does the same thing allowing the user to modify an existing area.

We provide a button to create a new annotation: the annotation will be added to the annotations tab and the annotation editor will be opened. To edit an annotation double click on an annotation



Splitting the view

Visual comparison of multiple objects is needed. The complicated part is to specify how the user will be able to select the items to be visualized.

This problem is very similar to the annotation editor need to specify additional entities, so we might use the same 'shopping cart' solution.



External viewers

The IVB allows the user to save digital objects to a file and let him open in his preferred viewer. If the user wishes to directly open the file, the file is downloaded to a temporary directory.

In the preferences a mapping specifies the default action for each digital object type (image, mesh, material, text etc.

Annotation Editor

Allows to edit an annotation or create a new one.

There are many different kind of annotations (comment, relation, and others which will be supported in the following releases) and to each one corresponds a different interface

Common to all interfaces is a 'shopping cart': a collection of entities used as a clipboard to assist the user in specifying an entity.

Annotation ✕

Shopping Cart (Drag Content from here):

Entity	
1	Michelangelo
2	David Statue
3	Leonardo da Vinci

Comment Area:

Add Comment to this Entity:

Comment:

Michelangelo di Lodovico Buonarroti Simoni, commonly known as Michelangelo, was an Italian Renaissance painter, sculptor, architect, poet, and engineer. Despite making few forays beyond the arts, his versatility in the disciplines he took up was of such a high order that he is often considered a contender for the title of the archetypal Renaissance man, along with his rival and fellow Italian, Leonardo da Vinci.]

Create

Annotation ✕

Shopping Cart (Drag Content from here):

Entity	
1	Michelangelo
2	David Statue
3	Leonardo da Vinci

Relation Area:

Add Relation with this Entity as Source:

Add Relation with this Entity as Destination:

Choose the Relation between the Entities:

GUI

Tabbed browsing

We will have a tabbed browser where search and browse will share the same tab. This way we can support easily multiple search and results. Annotations and viewers would also open as tabs

Michelangelo...

Coform^{3D}

Search: michelangelo david

- all
- media
- people
- events
- places
- images
- 3d models
- videos
- materials

Results

Navigation: 1 2 3 4 5 ... 9



White marble of carrara material
Small description or something useful usually in a couple of lines like this. Not too verbose though.
3D-COFORM Material repository



Human shape shape
Small description or something useful usually in a couple of lines like this. Not too verbose though.
3D-COFORM Shape repository



Michelangelo's David drawing study image
Michelangelo preparatory drawing, now at Museo dell'Accademia, photo by Soprintendenza alle belle arti. Created in 1552, Florence. Museo dell'Accademia, Florence



The David 3D model
Michelangelo's David digital reconstruction by Stanford
57M triangles, 0.1mm accuracy
Stanford Project



The David restoration, before - Photo 13 image
Michelangelo's David restoration photaranhic documentation.

Michelangelo...

David...

David

Author: Michelangelo
Location: Firenze
Photo By: Museo Accademia

Annotations hide

▶ **Greek nose**

▼ **Drill marks**

The hair present numerous drill marks.
Author: **Captain obvious**

[new annotation...](#)



Michelangelo... David... Annotate

Annotation editor

Relates to:



David photo 53



David photo 32

Add related entity...

Text

Some interesting remarks about the drill marks on the hair of the david and why should this relate also to the broken arm despite no apparent connection

Preferences

Similar to a web browser, preferences should manage:

- Cache (clearing it and setting its size, showing location of it)
- Mapping between kind of objects and viewers (mime mapping of web browsers)