

Linguistic Model

Narrative

Symbolism

Expert

The specifications in the **Linguistic Model** must be mirrored by implementation of domain imagery in the **Graphic Model**

Portal Publisher : specifies the graphics specification which is automatically made available in studio as a **design strategy**

Studio Publisher : satisfies the graphics specification which is to utilize the **design strategy** as its final definitive reference



Graphic Model

Symbology

Context

Designer

Specification



Implementation



SVG (Scalable Vector Graphic) **scales ideally** compared to **Raster** (Gif, JPG, Png) suffering **pixilation in resolution** increase on **device displays** so it is sought after. In the graphics domain of the IT industry the conventional means to **publish**, **contain** and **distribute** extensible **SVG** graphic content is a laborious process encumbered by the requirements of extensive expertise, additional ad hoc processes and several software utilities to perform all the **tasks** that achieve those objectives.

Our alternative **solutions** using an effective combination of technology and methodology remedies inadequacies of conventional means. Our platform of consolidated utilities require far less expertise to use and fewer processes due to comprehensive integration. The accompanying methodology of best practice already implemented by default or explicitly indicated to users accommodate logically sequenced processes following structured work flows which offers convenience to optimize efficiency and productivity. These utilities are explained through illustrating fragmented screen shots with labels that indicate the sequence of process and functional purpose of its use.

In addition to the previously mentioned complications, the single greatest limitation hampering usage of **SVG** to its fullest potential is that it must be manifested in an xml file which is static data presenting numerous processing limitations which cannot be addressed by the conventional methodology. Our methodology of defining those **SVG** documents **virtually** and storing it in a database, repositories overcome the limitations with **dynamic behavior** previously impossible. That is argued in detail in our document about the platform outside the scope of this document. We recommend that you consult it to understand its advantage solving those real world problems

The publishing utilities here exceed exclusively providing aesthetics as other **SVG** editors do. But focuses on provisioning for raw data to be dynamically wrapped in descriptors of virtual **SVG** documents. This offers tremendous flexibility for reuse and re-combination allowing for interactive **SVG** imagery that can be navigated

Specification to Implementation

We observed that there are typical requirements for defining and presenting  concepts and  constructs within the  domain using  SVG which are also common to them regardless of their respective industries nature as there is a typical process it follows for domain experts to describe and designers to visualize.

So we proceeded work on a solution with the realization that when conversing to relate  domain specific concepts and constructs, that  words (definition through rationalization) convey explanation generally but illustrating  mental imagery (definition through visualization) provides precise  comprehension

The solution thus effectively needs to accommodate a methodology by which to precisely define graphic content specification and also that by which to fulfill that graphic content specification which further reveal logical and creative aspects individually dealing with distinct  constructs of their own namely that of :

1.  **Rationalizing** that deals with the  construct of a  linguistic model
2.  **Visualizing** that deals with the  construct of a  graphic model

Once a  domain is  comprehended through a  Linguistic Model, its transposed into a tangible medium of vivid imagery depicted in a  Graphic Model.

These  constructs essentially pre-determine components of the  publishing mechanism, as it needs to structure and navigate explanation and also structure and navigate illustration of that explanation. This requires special  utilities that can provide the required functionality on the  platform namely that of the :



The  Portal Publisher that has all the  designers and  browser to facilitate maintenance and browsing of all content in the  linguistic model
The  Linguistic Model defines concepts and constructs in the  domain in terms of words rationalized in the  platform by the  constructs of :

1.  **Narrative** (scoping and structuring of the  subject matter) requiring a hierarchical mechanism that relates, directs and navigates explanation
2.  **Symbolism** (describing and illustrating  subject matter) requiring a diagrammatical mechanism which visualizes illustrations of explanation

Thus the  **Linguistic Model** is maintained through the :

1.  **Narrative Designer** : facilitating the narration through a hierarchical tree menu representing grouping and directional flow of the explanation
2.  **Symbolism Designer** : facilitating the symbolism through a diagrammatic sketch labeled by visual elements showing the required illustration



The  Studio Publisher which has all the  designers and a  browser to facilitate maintenance and browsing of all content in the  graphic model
The  Graphic Model depicts concepts and constructs in the  domain in terms of imagery illustrated within the  platform by the  constructs of :

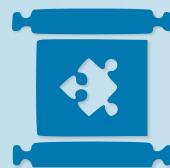
1.  **Symbology** (reuse-able visual of  subject matter) requiring a graphic mechanism that can render a composition with specific visual elements
2.  **Context** (navigation of  subject matter) requiring a linking mechanism which can temporarily couple compositions through visual elements

Thus the  **Graphic Model** is maintained through the :

1.  **Symbology Designer** : facilitates composition of graphics through a canvas and menu of visual elements with effects that can be painted on it
2.  **Context Designer** : facilitates inter linking by a search and listing that can drag compositions onto a canvas and link elements to compositions
3.  **Animation Designer** : facilitates dynamic binding of filter effects to visual elements along with animating visual elements in the compositions

Linguistic Model

To deal with the  abstract aspects within the  platform solution, for convenience of a common language dealing with the system we use specific terminology to define the constructs (definitions persisted) within the framework of those  concepts. To compartmentalize responsibilities to fall within the specific role and segregate concerns addressed by specific functionality, the  publishing mechanism supported by this  platform consists of the tool suite such as the linguistic and graphic designers provided specifically to manipulate the  constructs of the  linguistic model as well as the  graphic model persisted in  repositories



Linguistic Model by Domain Expert

The **Linguistic Model** contains graphic specifications which are to be used as design strategies for creating compositions in the **Graphic Model**. Defines the domain concepts in terms of wording, which in this platform are rationalized through the usage of both the narrative and symbolism. It is maintained through the **Linguistic Model**  designer in the web .

	Domain	A specific area of interest defined by specialized concepts and constructs that are represented in the imagery contained in a  graphic repository
	Vocabulary	Technical terminology of the domain language (jargon) is to be put into the plain language of laymen terms for easier understanding
	Semantics	Phrasing the description of intended technical imagery of the  domain into more simple wording of precise and concise semantics
	Article	Substantial body of text that assists as a treatise elaborating on the intended meaning of technical illustrations in the domain imagery
	Narrative	Is a  hierarchical containment mechanism that scopes and groups symbolism (specs) under a specific topic which is explored through imagery
		 Scope  Group  Identify  Brevity
	Subject	Indicates the focal point of the subject matter the symbolism concentrates on in both writing (articles) and illustration (symbology).
	Reference	Structured Reference translates a  symbolisms position in a  hierarchy into a  directory path notation for easy referencing
	Symbolism	Is a preliminary graphics specification of the intended imagery, that is to be followed as a definitive design strategy for a  graphic composition
	Link	Visual element appearing in a graphic composition that is temporarily configured in a context to link to another context on clicking
	Imagery	A description of intended imagery with elaborating descriptions for  annotated visual elements or a single  structured reference
	Behavior	Interactive animated behavior can be assigned to specific visual elements featuring in the  graphic composition put into a context
	Sketch	A preliminary sketch of the intended imagery's visual elements and layout usable as the visual guide for creating the  composition
	Element	Visual elements occurring in 3 specific categories namely :  shapes,  symbol and  text is specified within the  composition
	Tagging	Tag sketch with  visual elements (name, type & position) that can be  annotated with elaborated descriptions in the  imagery



Graphical Model by Graphic Designer

The **Graphic Model** contains all the  graphics compositions which implement their  design strategies specified in the **Linguistic Model**. Defines domain concepts in terms of imagery, which is illustrated through the usage of symbology and context which is graphically navigable. It is **maintained** through the **Graphic Model**  designer in the  Studio Publisher.

	Symbology	Virtual SVG composition composed of various visual elements clustered in independent reusable units that are configured in different contexts
	Element	Visual elements falling within 3 specific categories  shape  symbol  text that is rendered in the canvas of  compositions
	Shape	Archetypal, compound or path shape are used to create forms by means of shaping in the  symbol and  symbology designer
	Symbol	 shaped or  imported symbols conveniently drawn from a freely available resource of  symbol catalogs in the  platform
	Text	Native undecorated plain text, font decorated text or shaped text (follows form perimeters) can be put in  graphic compositions
	Context	Configured instance of an SVG composition facilitating  navigation of  domain imagery by temporarily wrapping visual elements in links
	Link	Visual element appearing in a graphic composition that is temporarily configured in a context to link to another context per click
	Background	Multiple symbology (compositions) can be imported, scaled and positioned as backgrounds for symbology (composition) to reuse
	Article	Multiple bodies of text can be attached to a symbology (composition) which can be stylized to complement the imagery esthetic
	Behavior	Visual elements can be decorated with combined filter effects along with assigning event driven animated behavior for dynamics

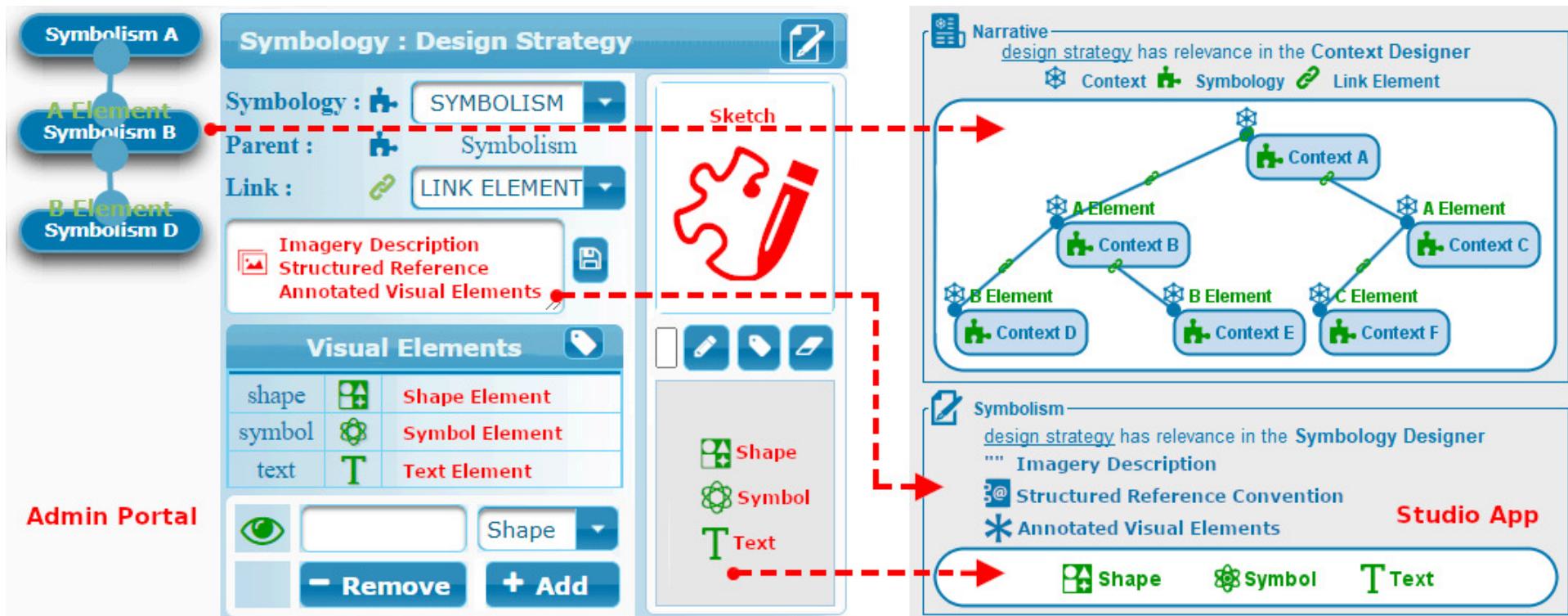
Design Strategy

After the creation of the graphics specification in the  **Portal Publisher**, it has to be conveniently and effectively communicated for accurate implementation within the  **Studio Publisher** for further designing. The  construct representing this specification in the terminology of this  platform is referred to as the  **Design Strategy**. It is automatically formulated during the process of authoring the  narrative and  symbolism that is a combination of both their specifications put into a visual format. This format is more easily understood by the  designers not acquainted with the  subject matter of the  domain, which the  expert should always keep in  mind.

 The  **Design Strategy** is to be consulted by the  designer during their  design session as the absolute definitive reference that defines the requirement and must be followed precisely to mirror the  linguistic model. It is located in every  designer and  browser where relevant by clicking on their  help button.

- The  **Narratives** hierarchical structure of Symbolism has immediate relevance in the  **Context Designer** whilst linking the compositions in  **Studio**
- The  **Symbolism** clustering composition of Elements has immediate relevance in the  **Symbology Designer** when creating compositions in  **Studio**

Below illustrated is both the  symbolism designer view in the  **Portal Publisher** with its corresponding  design strategy view in the  **Studio Publisher**. It is considered  best practice to utilize the functionality of a  design strategy consistently to ensure the  **Graphic Model** mirrors the  **Linguistic Model**.



 **Portal Publisher** is a linguistic utility maintaining the  **Linguistic Model** with designers and browser facilitating publishing of  **SVG** specs to the  repository An overview of each  designer and  browser are briefly illustrated for a general idea of functionality in the linguistic  **publishing mechanism** on the  platform

 The primary **design mechanisms** of the  **Portal Publisher** are the :

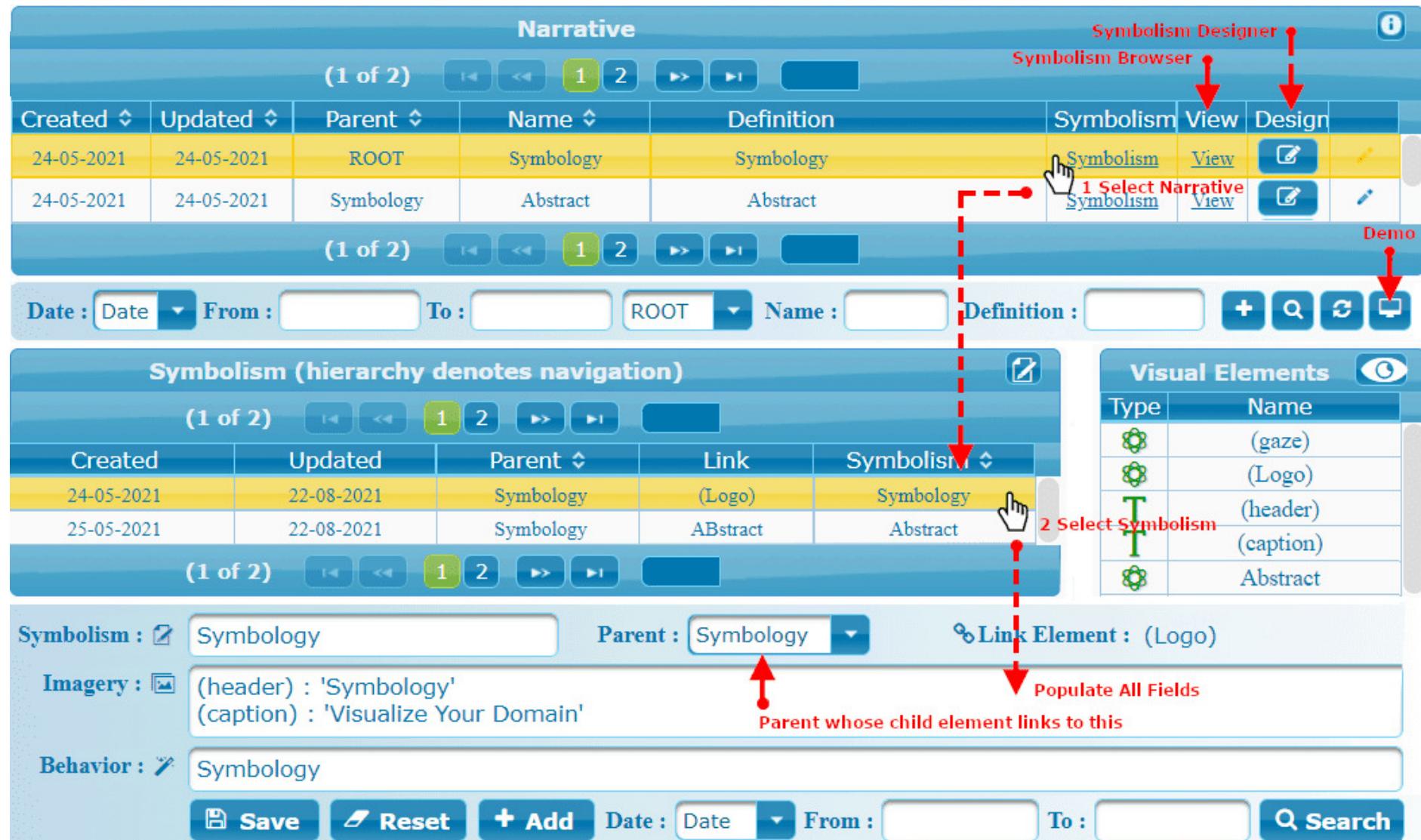
- ①  **Narrative Designer** : Mechanism where all by the  subject matter of  *concepts* and  *constructs* residing in the  domain of a  repository can be conveniently fragmented and compartmentalize into more manageable containers of information according to the strict principles of
 -  **Scope** determining the inclusion of certain subject matter based on whether or not it is immediately relevant to the specific topic
 -  **Group** determine the hierarchy of order and progression the subject matter will be explored in the chain of relevance in the topic

- ②  **Symbolism Designer** : Mechanism where all by the  subject matter of  *concepts* and  *constructs* residing in the  domain of a  repository can be explained through the use of a menu driven utility that facilitates the operation of diagrammatic and text capturing fields that aim to
 -  **Identify** the intended graphic compositions content through a basic sketch which is also labeled with specialized visual elements
 -  **Link** : select single element from drop menu of the parent symbolism's elements
 -  **Sketch** : mouse driven drawing canvas whose individual lines can also be removed
 -  **Label** : tags that are pinned onto sketch components indicating the visual element
 -  **Elements** : tabulated menu for adding or removing -  **Shape** -  **Symbol** -  **Text**
 -  **Brevity** Inside the  imagery field : use  structured reference convention to indicate forwarding to a specification
Inside the  imagery field : use  annotated visual elements to provide more lengthy descriptions for labels

 The primary **browsing mechanisms** of the  **Portal Publisher** are the :

- ①  **Narrative Browser** : Mechanism where by all the  subject matter of *concepts* and *constructs* residing in the  domain of a  repository is navigated
- ②  **Symbolism Browser** : Mechanism where by the intended  imagery of *concepts* and *constructs* residing in the  domain of a  repository is navigated

Narrative Designer  **Publishing mechanism** maintaining  narratives which  scope and  group related  symbolism dealing with common  subject matter which is to be consulted as the  design strategy for referencing by the  graphic designer in the  context designer featuring within the 



Narrative

(1 of 2)					Created	Updated	Parent	Name	Definition	Symbolism	View	Design
24-05-2021	24-05-2021	ROOT	Symbology	Symbology	24-05-2021	24-05-2021	Symbology	Abstract	Abstract	 Symbolism	 View	 
24-05-2021	24-05-2021	Symbology	Abstract	Abstract						 Symbolism	 View	 

Date : Date From : To : ROOT Name : Definition :    

Symbolism (hierarchy denotes navigation)

(1 of 2)					Created	Updated	Parent	Link	Symbolism
24-05-2021	22-08-2021	Symbology	(Logo)	Symbology	25-05-2021	22-08-2021	Symbology	ABstract	Abstract

Symbolism :  Symbology Parent :  Symbology  Link Element : (Logo)

Imagery :  (header) : 'Symbology'
(caption) : 'Visualize Your Domain'

Behavior :  Symbology

 Save  Reset  Add Date : Date From : To :  Search

Symbolism Designer

Symbolism Browser

Symbolism

View

Design

Visual Elements

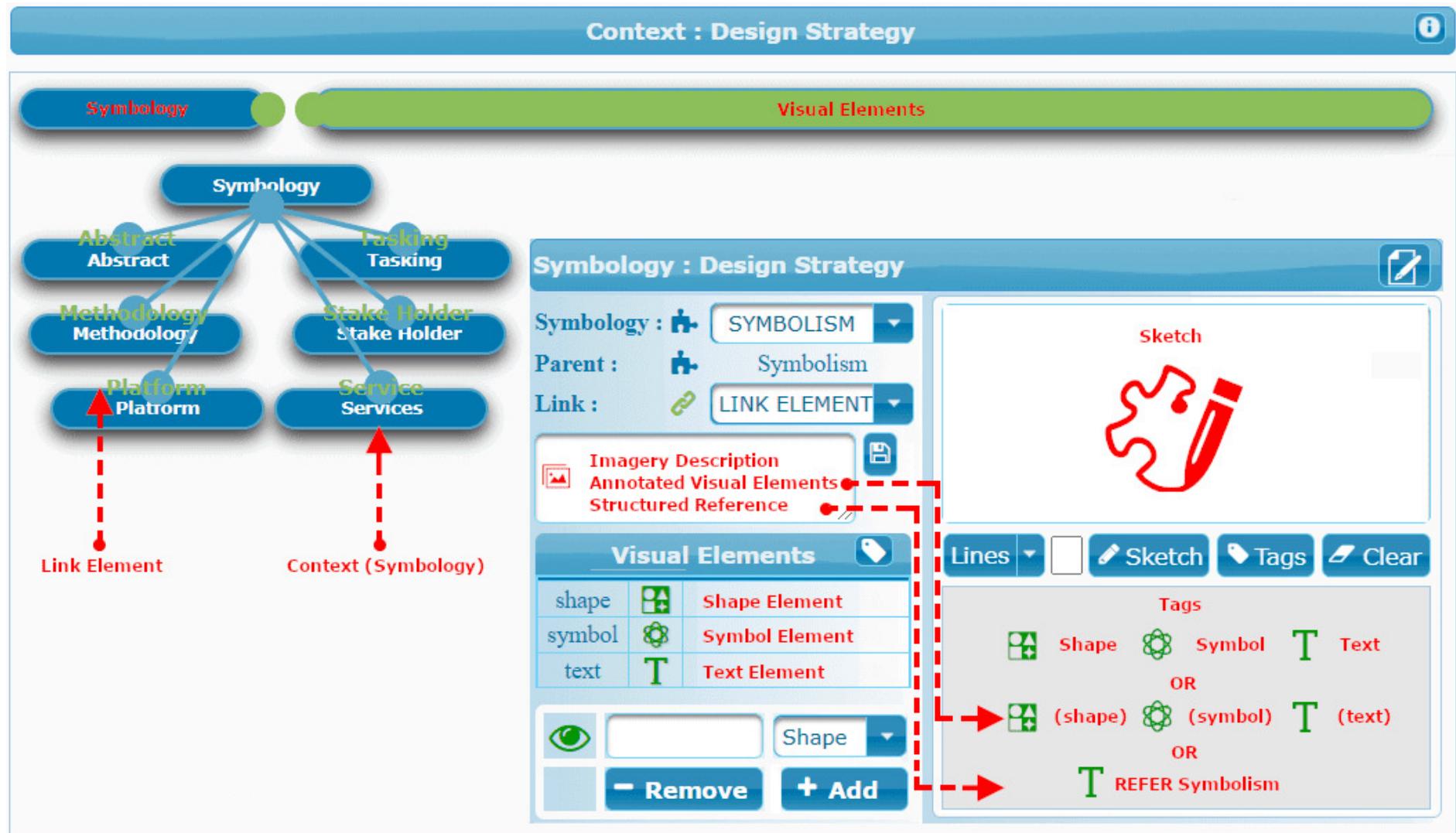
Type	Name
	(gaze)
	(Logo)
	(header)
	(caption)
	Abstract

1 Select Narrative View
2 Select Symbolism

Parent whose child element links to this
Populate All Fields

Symbolism Designer  Publishing mechanism maintaining the  symbolism specifying instructions and a preview of  visual elements in re-usable graphic specifications to be used as the  design strategy which the  graphic designer is to translate into a  symbology (graphic composition) during their  design session

Context : Design Strategy



Symbology 

Visual Elements

Symbolology : Design Strategy 

Symbology :  SYMBOLISM 

Parent :  Symbolism

Link :  LINK ELEMENT 

Imagery Description  Annotated Visual Elements  Structured Reference 

Visual Elements 

shape		Shape Element
symbol		Symbol Element
text		Text Element

 Shape  Remove  Add

Sketch



Lines  Sketch  Tags 

Tags

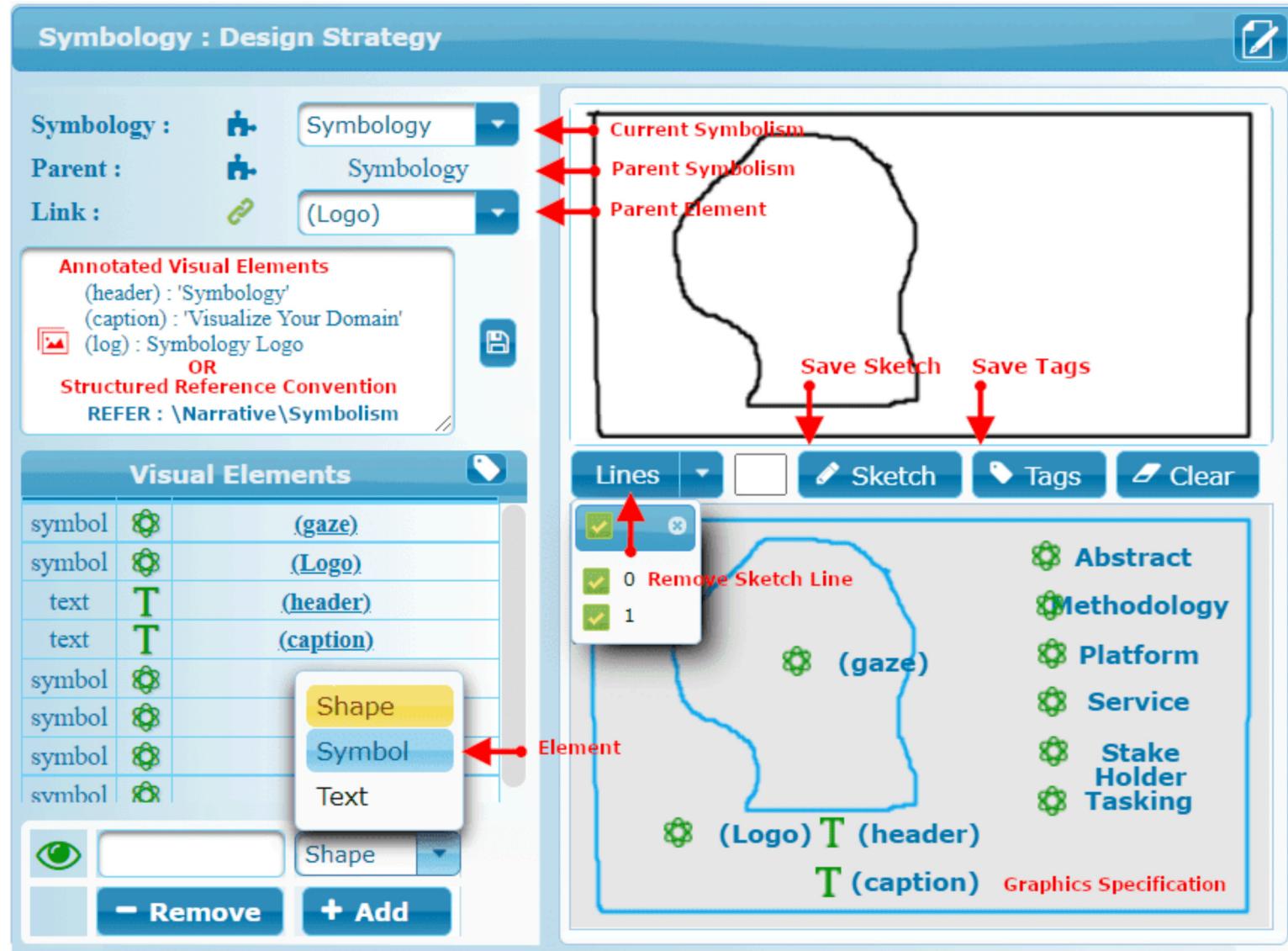
 Shape  Symbol  Text
OR
 (shape)  (symbol)  (text)
OR
 REFER Symbolism

**Symbolism
Design
Strategy**

All the relevant specifications provided here appear as the symbology  design strategy to be used within the  specifying the following :

-  **Elements** : listing all of the visual elements
-  **Imagery** : note definition of the imagery
-  **Sketch** : make a rough sketch of imagery
-  **Labels** : peg labels of visual elements

Symbolism : Design Strategy



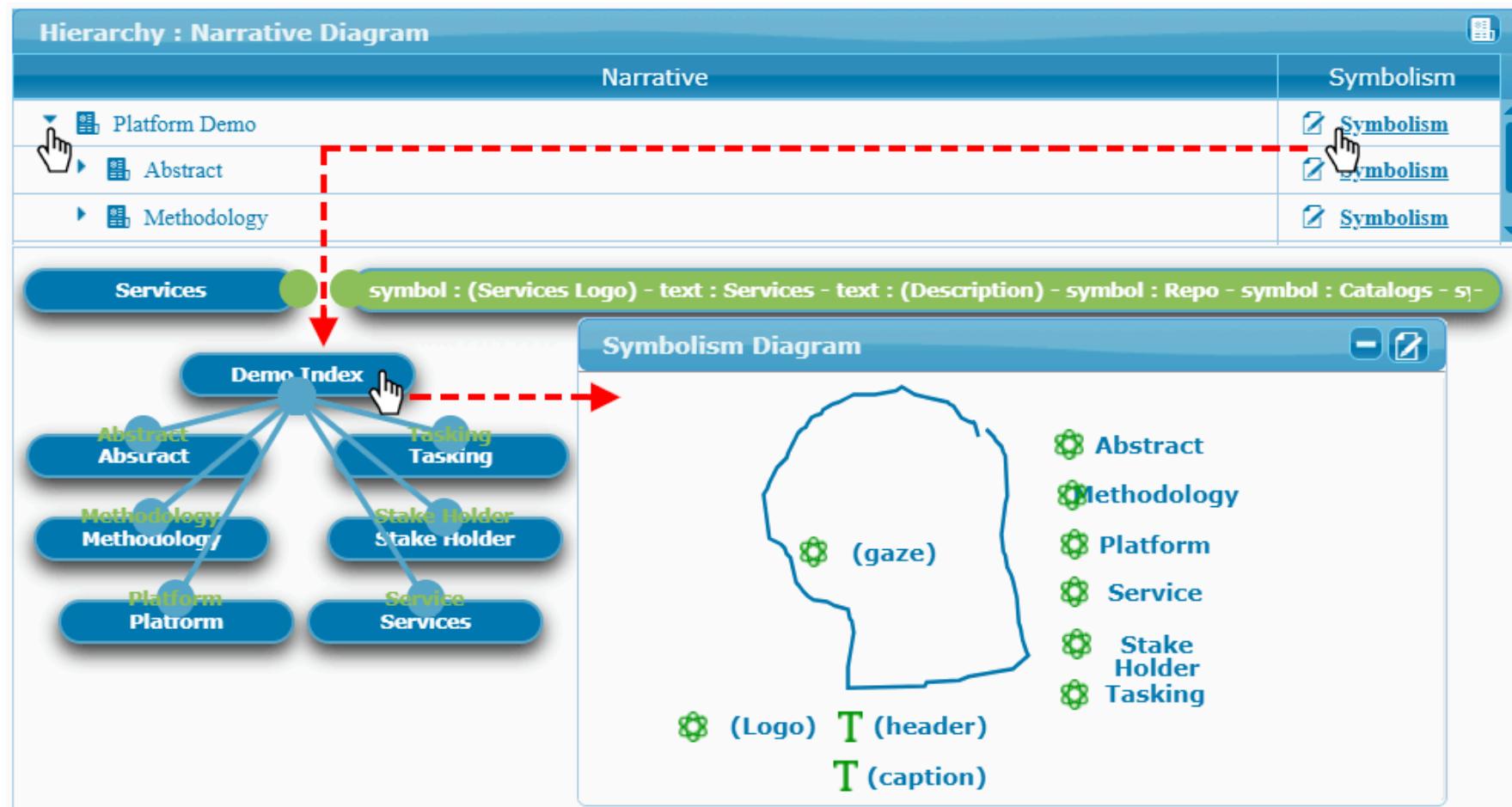
The screenshot illustrates the 'Symbolism : Design Strategy' interface. On the left, there's a sidebar with dropdown menus for 'Symbolology', 'Parent', and 'Link'. Below these are sections for 'Annotated Visual Elements' (header, caption, log) and 'Structured Reference Convention' (REFER: \Narrative\Symbolism). The main area features a 'Visual Elements' table:

	symbol	(gaze)
symbol		(Logo)
text		(header)
text		(caption)
symbol		
symbol		
symbol		Shape
symbol		Symbol
svmbol		Text

At the bottom of the table are buttons for 'Remove' and 'Add'. To the right of the table is a 'Lines' dropdown menu with options 0 and 1, and a 'Remove Sketch Line' button. The main workspace shows a sketch of a head profile with labels: 'Current Symbolism' (highlighted by red arrows), 'Parent Symbolism', 'Parent Element', 'Save Sketch', and 'Save Tags'. Below the sketch is a 'Tags' section with a list of categories: Abstract, Methodology, Platform, Service, Stake Holder Tasking, (gaze), (Logo), (header), (caption), and Graphics Specification.

Narrative & Symbolism Browser

Facilitates browsing through the **Narratives** (grouped symbolism) in a hierarchical tree menu of expandable records which on expansion fetch more **Narratives** which in turn can also be expanded. To their right **Symbolism** are browsed by clicking its link so populating the bubble tree diagram. Facilitates browsing through the **Symbolism** (graphics specification) through a bubble tree diagram by clicking on a bubble which will populate the illustration **symbolism diagram** depicting the specifications **sketch** with its **visual elements** and **labels** depicting the intended **imagery**.



Reference Convention

To ensure a **Symbolism** (graphic specification) is not duplicated but declared once and referred to by a unique identifier in the **Linguistic Model** a structured convention is needed. A mechanism facilitating this is the **structured reference convention** translating hierarchical parent child nodes into a flat directory path specified in the **imagery** attribute of a **Symbolism**. An illustrated example specified below features to the right the **Narrative : "Best Practice"** contains 2 references to the **Symbolism : "Design Strategy"** at left residing at the flat specified directory path :

REFER : \symbology\Abstract\Linguistic Model\Design Strategy.

Narrative Hierarchy		
Narrative	Definition	Symbolism
Platform Demo	Platform Demo	Symbolism
Abstract	Abstract	Symbolism
Linguistic Model	Linguistic Model	Symbolism

Parent Child Node (hierarchical)

Symbology : Design Strategy

Symbology : [SYMBOLISM](#)

Parent : [LINK ELEMENT](#)

Link : [LINK ELEMENT](#)

REF\Platform Demo\Abstract\Linguistic Model\Design Strategy

Directory Path Notation (flat)

Visual Elements

text [T](#) REFER Design Strategy

[Remove](#) [Add](#)

Indicate Reference

REFER Design Strategy

Symbolism in the Symbolism Designer



Interfaces Editor



◎ **Portals** are embedded in websites or apps by inserting an ◎ **instances** url in an iframe and detects the domain configuring its ◎ **interface**.
The portal automatically detects a websites domain and applies esthetics of the configured instance but blocks unregistered websites or apps.



❖ **Channels** are embedded in websites or apps by inserting an ❖ **instances** url into an iframe whose parameters then configure its ❖ **interface**.
The editor facilitates browsing imagery to select an initial composition and configuring the style complementing the esthetics of a hosts brand.



◎ **Portlets** are embedded in websites or apps by inserting an ◎ **instances** url into an iframe whose parameters then configure its ◎ **interface**.
The editor facilitates browsing imagery to select an initial composition a portlet is initialized with but redirect if not embedded inside an iframe.

Demo Portal : <http://> Channel : <http://> Portlet : <http://> Width : 700 Height : 500 Icon : 0.8 Track : Theme : Nav : Glass :

Theme Color
#0178af

Generated Colors
#188fc6
#34abe2
#4ec5fc
#5ad1ff

Nav Color
#b5d7e8

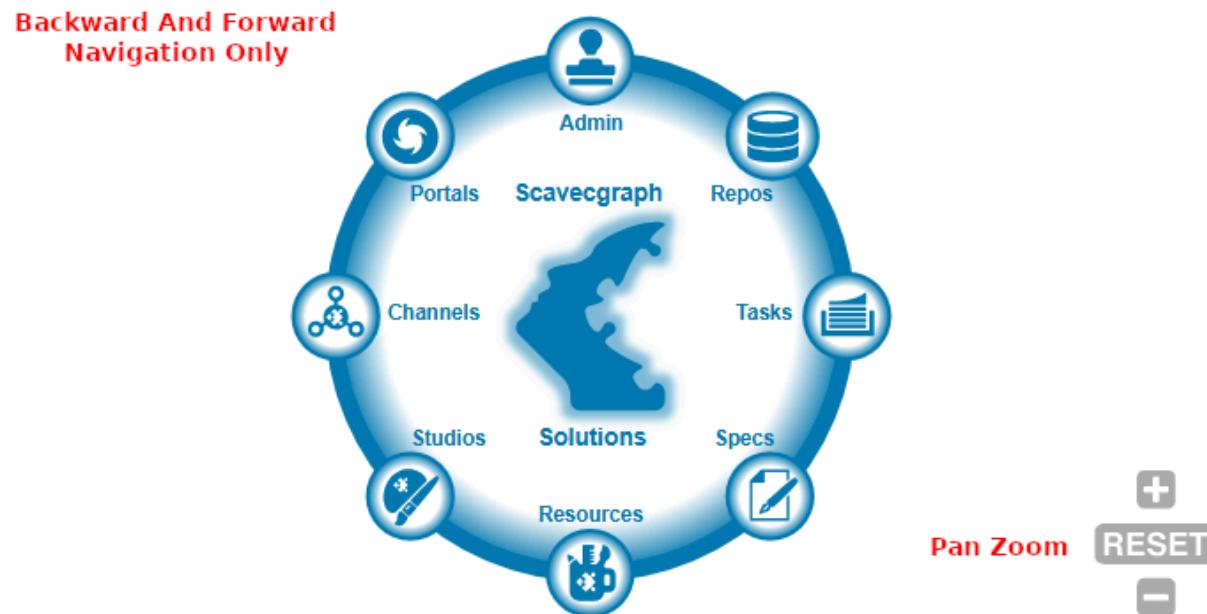
Cached History

Portals Admin Scaevgraph Repos Tasks Specs Solutions Resources Studios Channels

Embedded Portlet Pan Zoom

**Portlet
Interface**

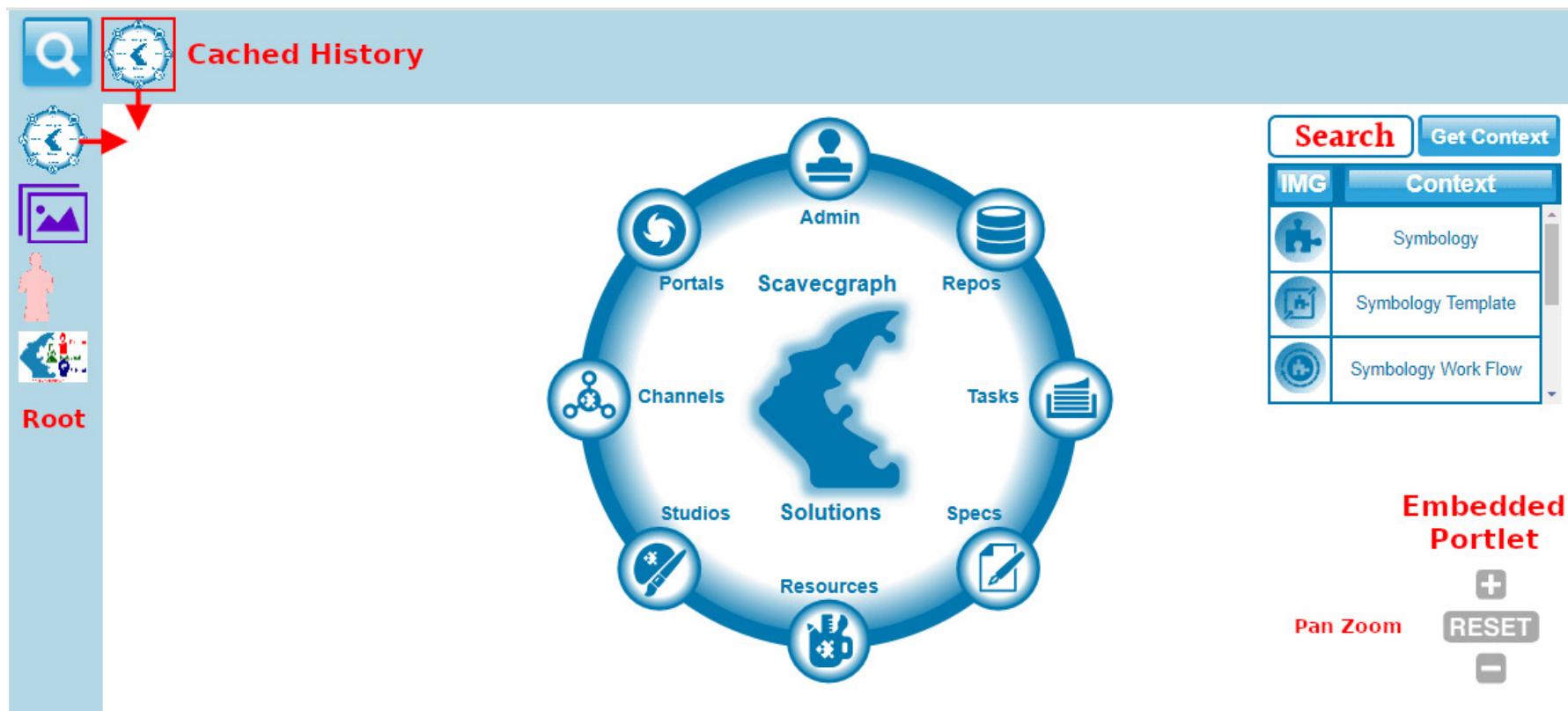
Only registered ✿ websites and 🛡 apps can embed a Ⓜ Portlet within an iframe in the hosting container or else it will be redirected and so blocked. A Ⓜ Portlet facilitates embedding a configurable container through which ✿ SVG content is trafficked with only backward and forward navigation. A Ⓜ Portlet has pan and zoom functionality to adjust viewing of the ✿ SVG content by being able to move the imagery and offer position zooming.



Channel Interface

Only registered ❖ websites and ◄ apps can embed a ❖ Channel within an iframe in the hosting container or else it will be redirected and so blocked. A ❖ Channel facilitates embedding a **configurable container** that has a content caching mechanism, fits a view ports dimensions and automatically scales the ❖ **SVG content** trafficked through its embedded ◎ Portlet providing pan and zoom functionality to adjust viewing of the ❖ **SVG content**. A ❖ Channel facilitates a container whose styling **Theme & Color** is configured to complement the **brand** (graphic identity) of its hosting container. For optimal flexibility configuration for dimensions and esthetics are defined by URL parameters that are assigned default values if not being specified.

Config URL : channel.html?showTracker=true&trackerScale=0.8&fitWidth=500&fitHeight=500&color=0178af&navColor=b5d7e8&glass=true



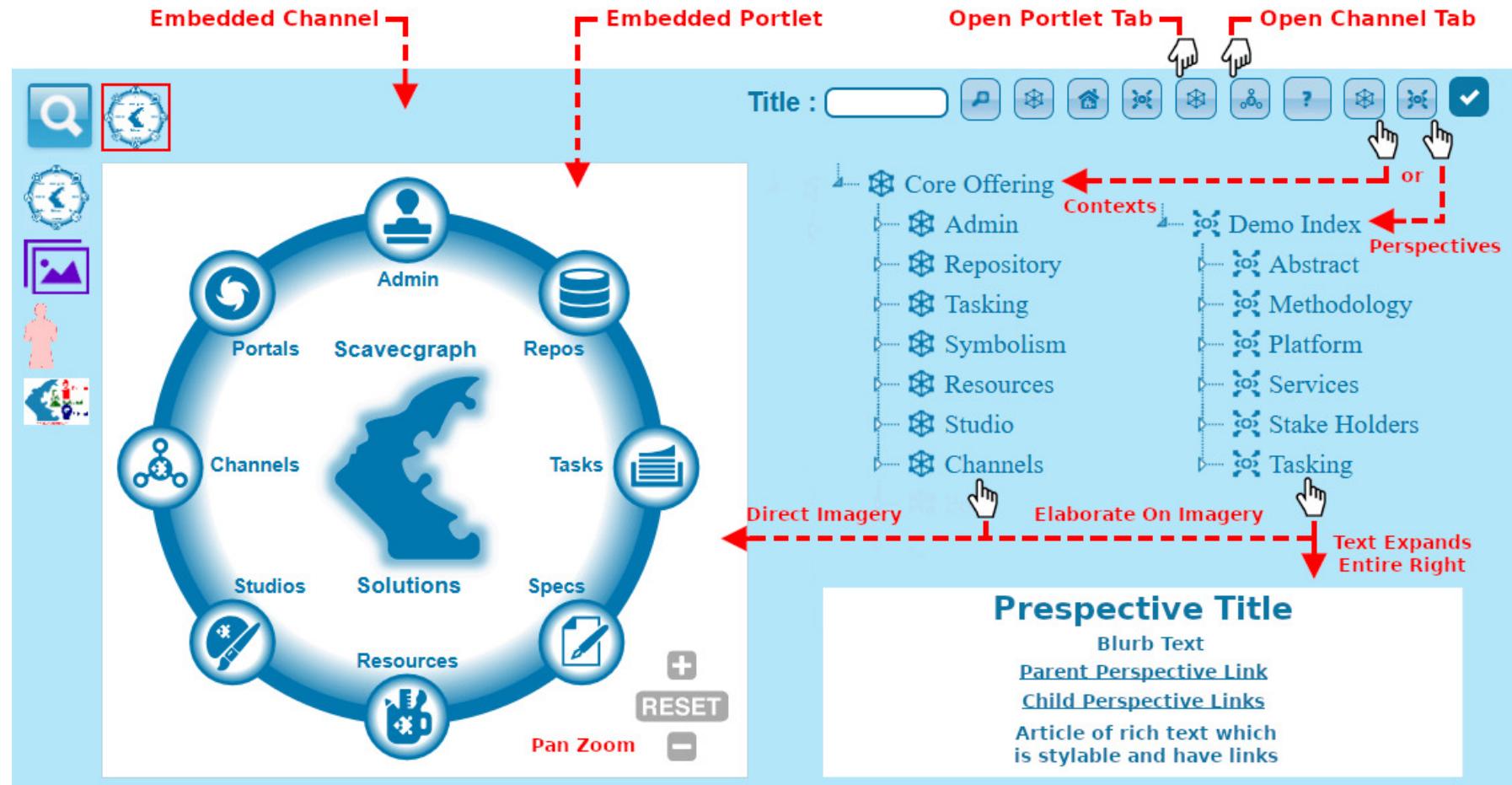
Portal Interface

Only registered * websites and * apps can embed a ○ portal within an iframe in the hosting container or else it will be redirected and thus so blocked.

When embedded a ○ portal automatically detects a * websites domain and applies the configured esthetics mapped to that specific registered website.

○ Portals provide an interface to * imagery in a * SVG Repository through an embedded * Channel whose imagery is navigated by an embedded : hierarchical * Context (configured composition instance) tree that navigates the imagery facilitating an aerial navigation of the imagery for convenience.

hierarchical * Perspective (rich text article) tree that navigates the imagery and also renders a rich text article elaborating further on the specific imagery.



**Portal
Admin
Console**

The ○ **portal admin console** facilitates the registration of ⊗ **websites** and a visual style editor that configures the appropriate esthetics of the ○ **portal** to integrate seamlessly to complement the **brand** (graphic identity) of the ⊗ **website** it is to be embedded in. It also facilitates maintenance of users as authors. A ⊕ publishing mechanism also facilitates creation and maintenance of hierarchical ⊗ **perspectives** (rich text articles elaborating on the imagery) and link specific imagery to a ⊗ **perspective** by browsing and selecting it in the left ⊗ **channel**. Thus written content can so direct and elaborate on graphic content.

The screenshot shows the **SVG Portal Admin Console** interface. On the left, there's a circular navigation menu with various icons and labels: Portals, Scavengraph, Repos, Tasks, Solutions, Specs, Resources, Studios, Channels, and Portfolios. Above this menu is a search bar with a magnifying glass icon and a globe icon. To the right of the menu is a large panel titled "Register Websites (with configured esthetics) And Perspective Publishers". This panel has tabs for Admin, Users, Websites, and Perspectives. The "Websites" tab is selected, showing a table of registered websites:

Created	Updated	Def	Org	Host	CLR
2025-01-07 14:55	2025-09-30 18:11	true	Scavengraph	www.scavengraph	
2025-01-07 16:55	2025-09-30 18:11	false	Scavengraph	10.0.0.115	
2025-01-17 23:53	2025-03-14 20:28	false	Partner 1	www.website1.com	
2025-01-31 16:29	2025-03-14 20:29	false	Partner 2	www.website2.com	
2025-01-31 16:34	2025-03-14 20:29	false	Partner 3	www.website3.com	

Below the table are input fields for Org, Hostname, Tel, E-mail, and a "Add Website" button. At the top of the right panel, there are tabs for Admin, Users, Websites, and Perspectives, with the Websites tab highlighted. Red arrows point from the "Channel" label and the "Websites" tab to their respective counterparts in the interface.

Context Console

The ❖ **Context Web Console** is accessible from ❖ **Context Designer** and ❖ **Animation Designer** and is for the purpose of quick reference to see what the ❖ **Contexts** look like in a browser. The navigation ❖ **history is cached** for ideal performance over the network not re-fetching compositions fetched before.



Symbology Console The ✿ **Symbology Web Console** is accessible from the ⚒ **Symbology Designer** for a quick reference to see what the ⚒ **Symbology** look like in a browser.

The screenshot shows the Scavecgraph Symbology Web Console. At the top, there are tabs for "Scavecgraph", "View", "Context Console", and "Symbol Console". A dropdown menu "Symbol Libs" is open, showing "Actual" and other options. On the left, there's a large blue puzzle piece graphic. To its right, a vertical list of categories with icons: "Abstract" (puzzle piece), "Methodology" (map), "Platform" (military rank), "Service" (gear), "Stake Holder" (two people), and "Tasking" (document). On the far right, a sidebar titled "Symbol Libs" lists categories with icons: "Demo Index" (map), "Abstract" (puzzle piece), "Concepts" (puzzle piece), "Domain" (globe with gear), "Vocabulary" (book), and "Semantics" (pen).

Scavecgraph
Visualize Your Domain

Symbol Libs

Actual

Demo Index

Abstract

Concepts

Domain

Vocabulary

Semantics

Abstract

Methodology

Platform

Service

Stake Holder

Tasking

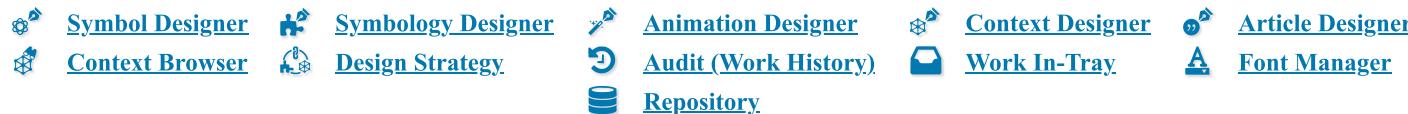
Symbol Console

The ✿ Symbol Web Console is accessible from the ✿ Symbol Designer and is for quick reference to see what the ✿ Symbols look like in a browser.

The screenshot shows the Symbol Web Console interface. At the top, there is a navigation bar with links to "Scaevgraph", "Context Console", "Symbology Console", and a "Refresh" button. To the right of the navigation bar is a "Symbol Libs" dropdown menu with three options: "Demo", "Shaped", and "Imported", with "Imported" currently selected. Below the navigation bar is a large, abstract black silhouette of a figure, possibly a person or a stylized animal, centered on the page. To the right of this central image is a vertical list of six symbols, each accompanied by a small icon and a label:

- symbology-abstract
- Linguistic Model
- Domain
- Article
- Vocabulary
- Semantics

The  **Studio Publisher** is a SVG graphics utility maintaining the  **Graphic Model** through its extensive tool suite which facilitates publishing of  **SVG content** to a  repository. The [info site](#) features a  slide show paging through the  designer and  browser screens give a preview of the functionality in the  platforms graphic  publishing mechanism. The tasking, specs and work history functionality is also comprehensively integrated for optimal work efficiency and convenience.



Symbol Designer : Help



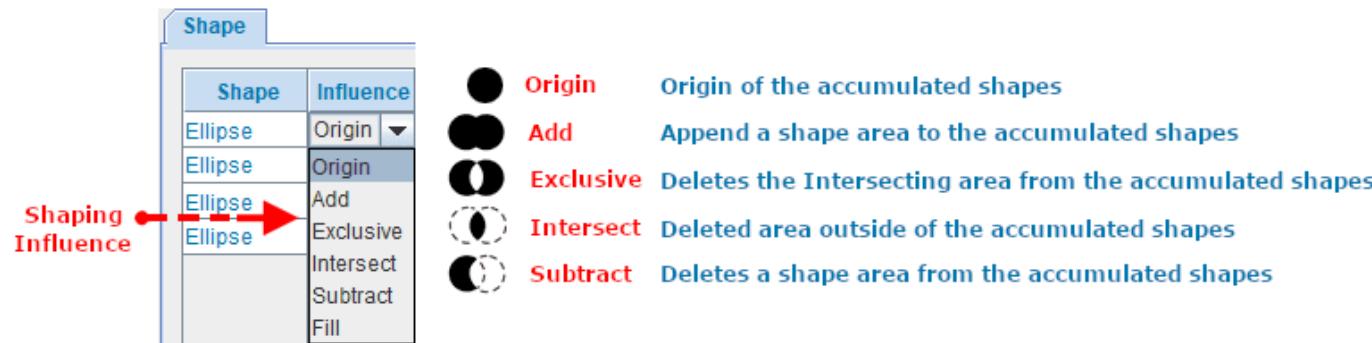
The  **Symbol Designer** has all the necessary  utilities required to perform the commonly needed manipulation of  **SVG**. In order to understand how to make use of its full functionality acquaint yourself with all the instructional use in this  help tutorial and try your hand out at some experimentation by using a  designing template

Shapes All  shaped symbols are comprised of a collection of shapes having influence over one another, to generate what is commonly referred to in  studio as a  principal form which is the single shape produced and use-able in the  symbology designer when importing  symbols into compositions. To create the  shape and start sculpting a  symbol you must select an  archetypal shape from the menu and dimension it on the canvas with a single swiping motion.



Shape Influence

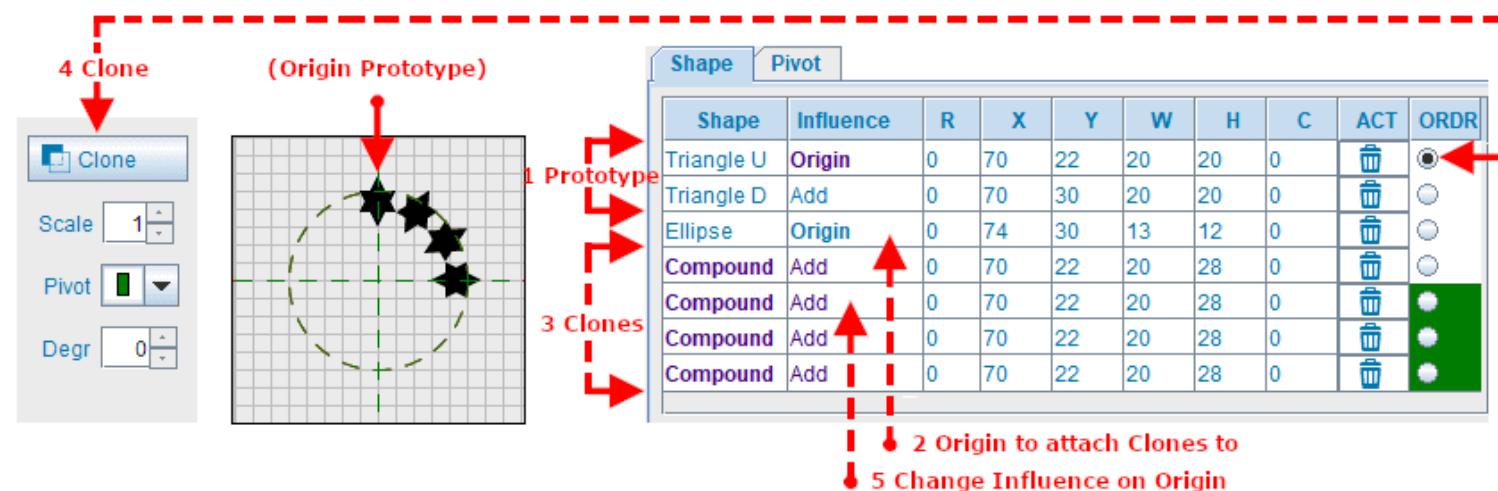
Shaping is the core function of a graphic editor and has certain concepts that need to be understood properly before using this  utility effectively. When  archetypal shapes are combined with various influences over one another they result in a compound shape. Each shape can also be manipulated further by scaling, positioning or rotating. If a compound shape in a composition conforms to three rules it becomes the  principal form use-able as a  symbol



Prototype Form

A compound  shape can be duplicated and thus manipulated as one portable entity through out a composition. The ability to repeatedly use it through out a composition is achieved through the process of  cloning. Once cloned it logically becomes a prototype of the clones. This offers 3 distinct advantages of :

1. Any change made to the prototype automatically cascades through to the clones
2. Saving the duplication of effort to recreate it for every required instance there of
3. Byte economy as it does not unnecessarily duplicate memory storage in repository

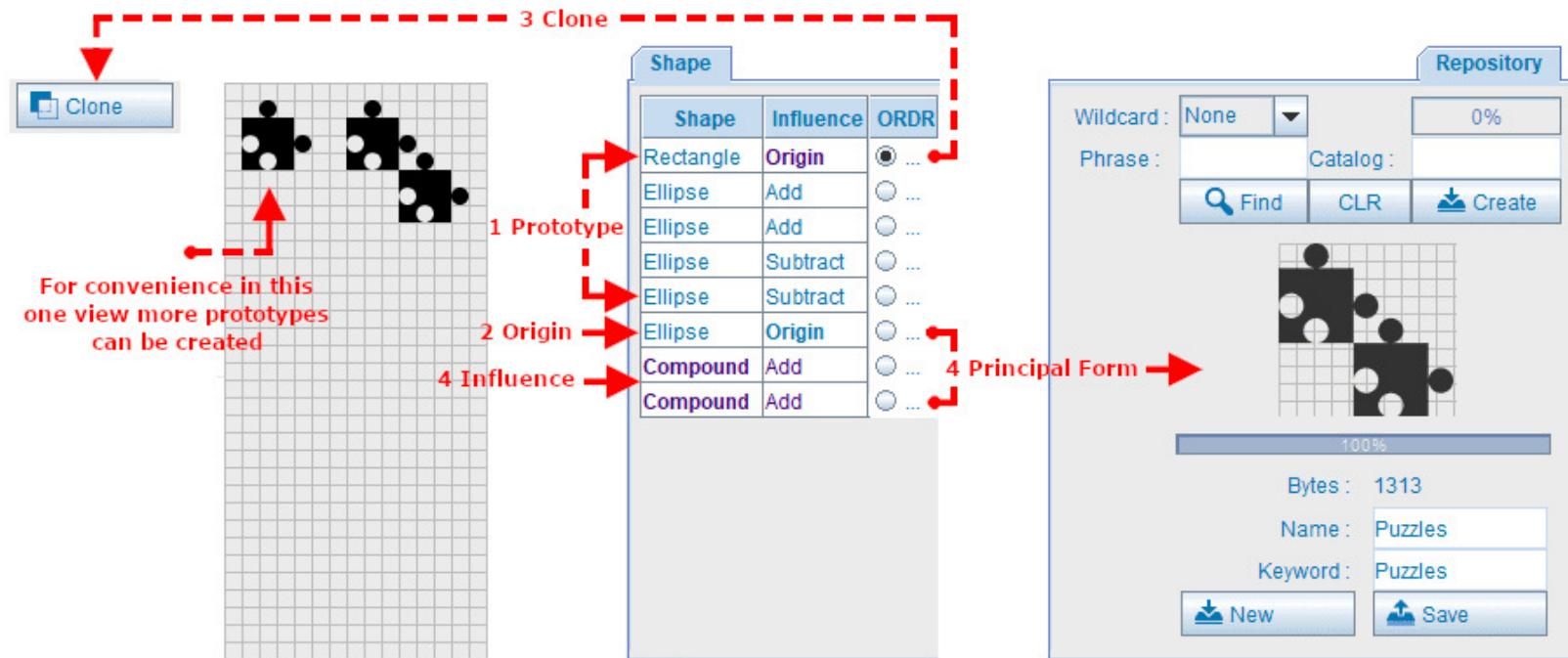


Principal Form

A **█** principal form is the final objective of a **⊗** symbol being shaped. The system automatically identifies it when a compound **⊕** shape conforms to rules :

1. Is a single Shape or Compound Shape with an origin
2. Not referenced by a Clone (thus not being a prototype)
3. Can have numerous Clones of other prototypes influencing it

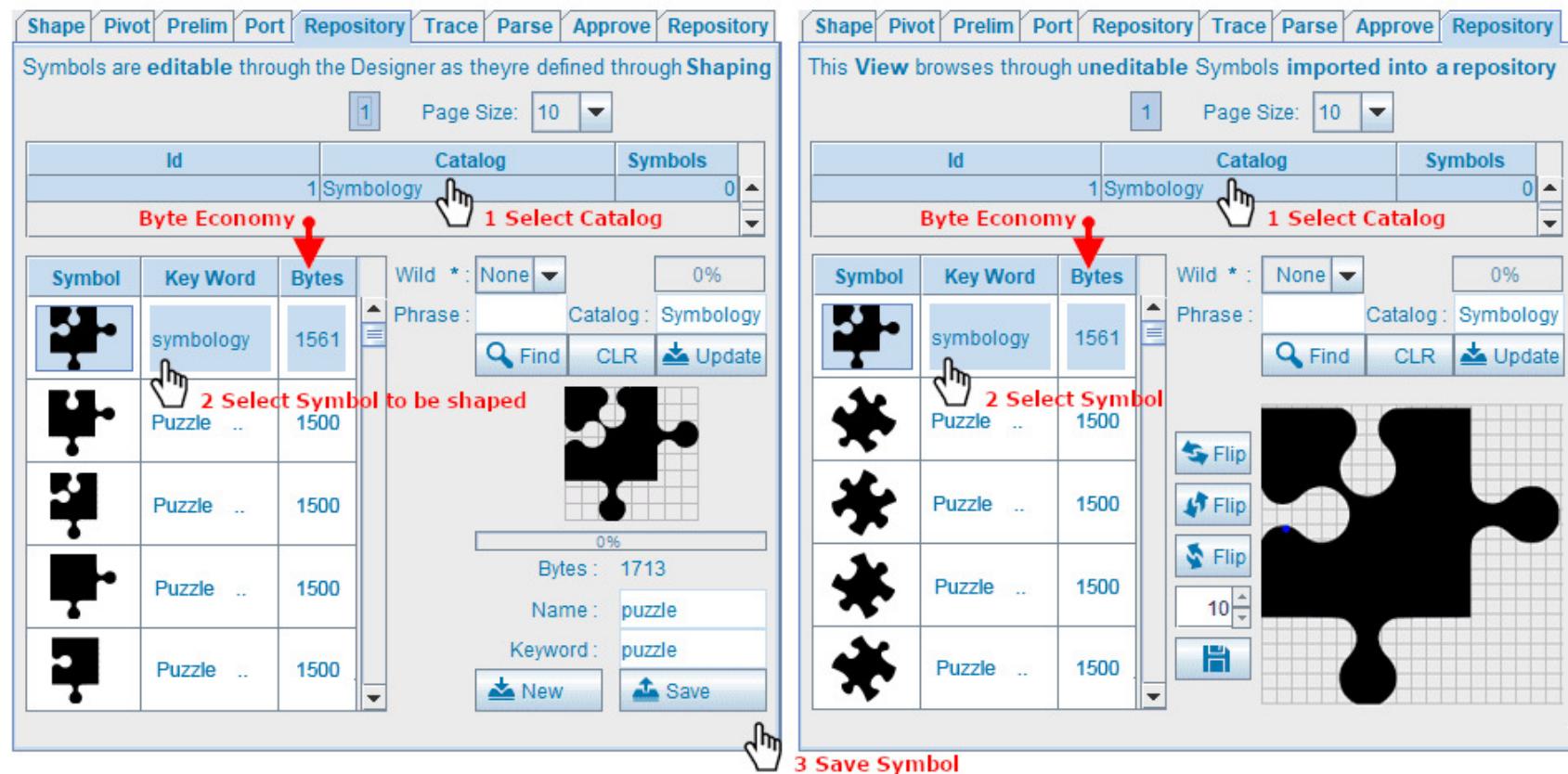
Thus many compound **⊕** shape and **█** clones can be created and their prototypes are excluded from the final composition to isolate the **█** principal form to become the resulting **⊗** symbol. Thus many prototype forms can conveniently feature through out single view while creating a complicated **█** principal form



Symbol Types

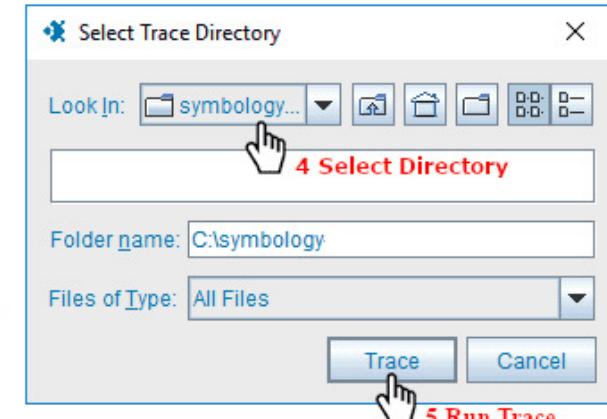
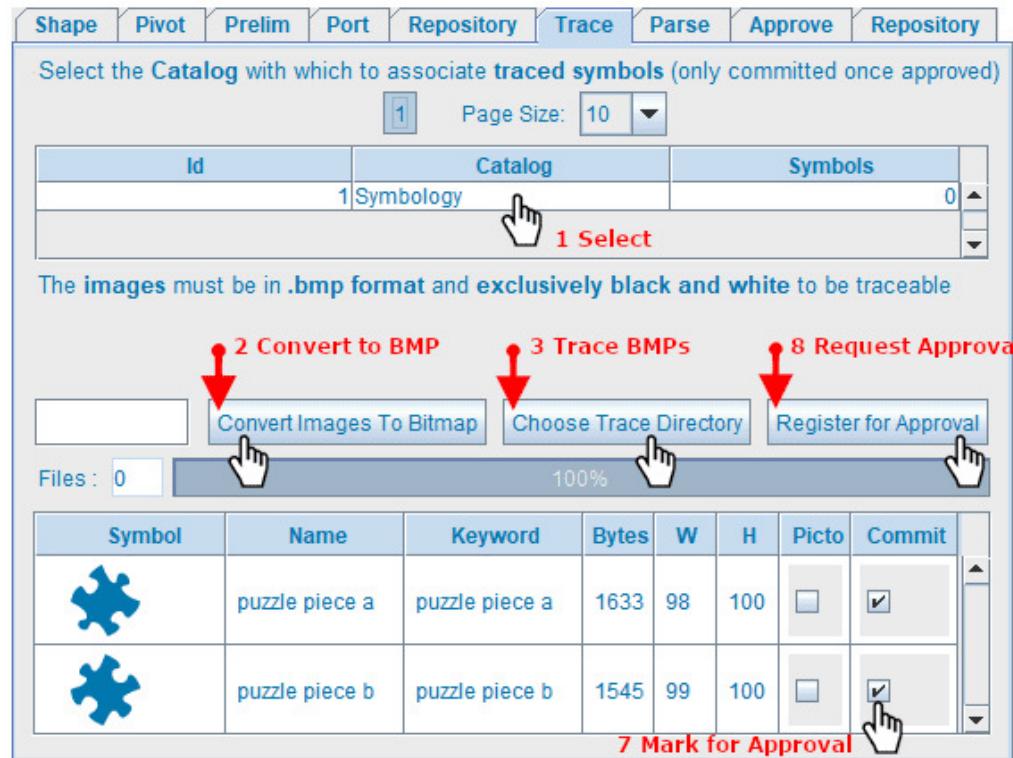
There are 2 distinctly different types of symbols which can be accessed and manipulated from within the symbol designer namely that of the :

1. **Shaped Symbol** : which is created through the shaping process of a symbol defined in the formulation of a principal form.
2. **Imported Symbol** : locally parsed from SVG files, or traced from BMP files or pooled from the online Symbol Catalogs.



Trace Symbol

➤ Studio is integrated with Po-Trace (industry standard) tracing shapes from black & white BMP images First a directory containing the BMP files must be located and once traced their forms populate the listing as illustrated and are manipulable through a path editor in the Symbol Designer.



6 Tracing Process Runs and once completed the listing at left is populated

**Parse
Symbol**

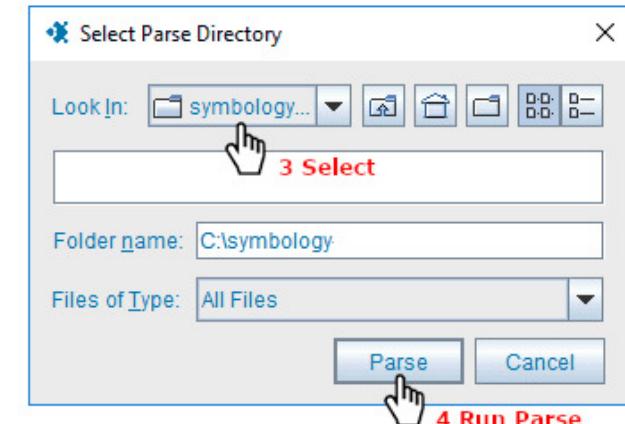
Shapes occurring as path elements in SVG can be parsed in the Parse pane. First the file directory containing the SVG files must be located, then once it has been selected they are parsed by the system. Once parsed their forms populate the listing as illustrated. However they can be partially manipulated later through flipping or rotating. Unfortunately no other graphic elements such as the standard shapes like Rectangles and Ellipses etc .. can be parsed

The Parse pane interface includes tabs for Shape, Pivot, Prelim, Port, Repository, Trace, Parse, Approve, and Repository. A message at the top says: "Select the Catalog with which to associate traced symbols (only committed once approved)". Below this is a table with columns Id, Catalog, and Symbols. A row is selected with the value "1 Symbology". A red arrow labeled "1 Select" points to this row.

A note below the table states: "The parser only extracts paths then merges and re-scales them to fit a 100 X 100 PX space". Two red arrows point to buttons: "2 Parse Paths in SVG File" points to "Choose Parse Directory", and "7 Request Approval" points to "Register for Approval".

Below these buttons is a progress bar showing "Files: 0" and "100%". A red arrow labeled "6 Mark for Approval" points to a checkbox in a table below. The table has columns Symbol, Name, Keyword, Bytes, W, H, Picto, and Commit. It contains two rows:

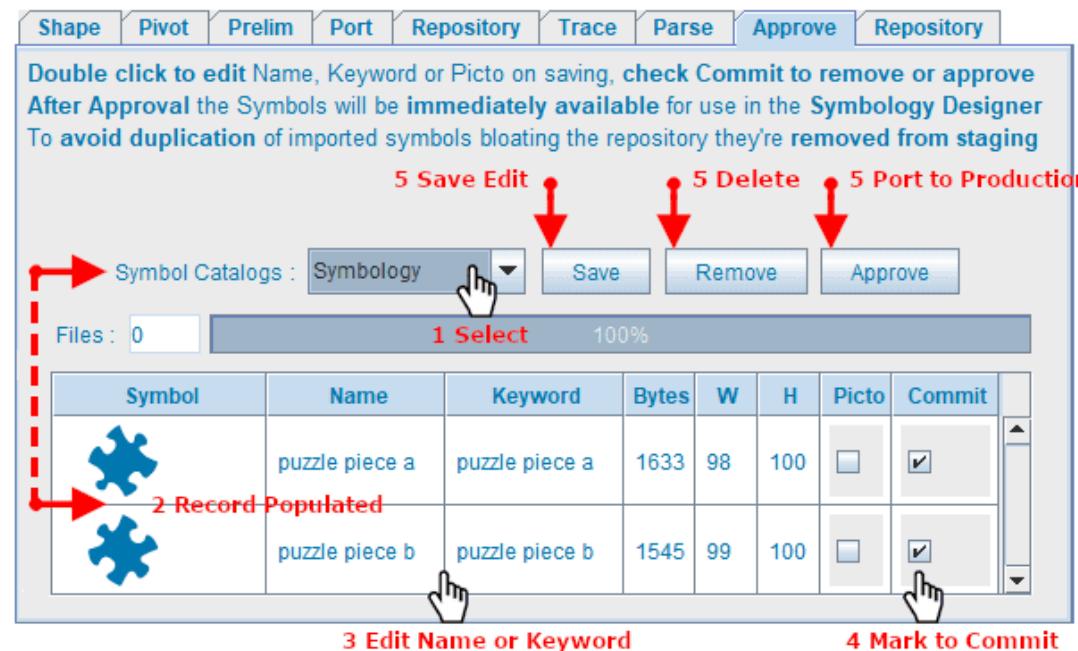
Symbol	Name	Keyword	Bytes	W	H	Picto	Commit
	puzzle piece a	puzzle piece a	1633	98	100	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	puzzle piece b	puzzle piece b	1545	99	100	<input type="checkbox"/>	<input checked="" type="checkbox"/>



5 Parsing Process Runs and once completed the listing at left is populated

Approve Import

Imported Symbols can be browsed through in the second Repository pane. It facilitates viewing symbols along with editing their name and keywords. Undesired symbols can be removed, or those deemed acceptable can be approved and released into production for use in the symbology designer. Symbols that are sent to a specific symbol catalog in a repository from the send symbol page in the web portal also need to be approved.





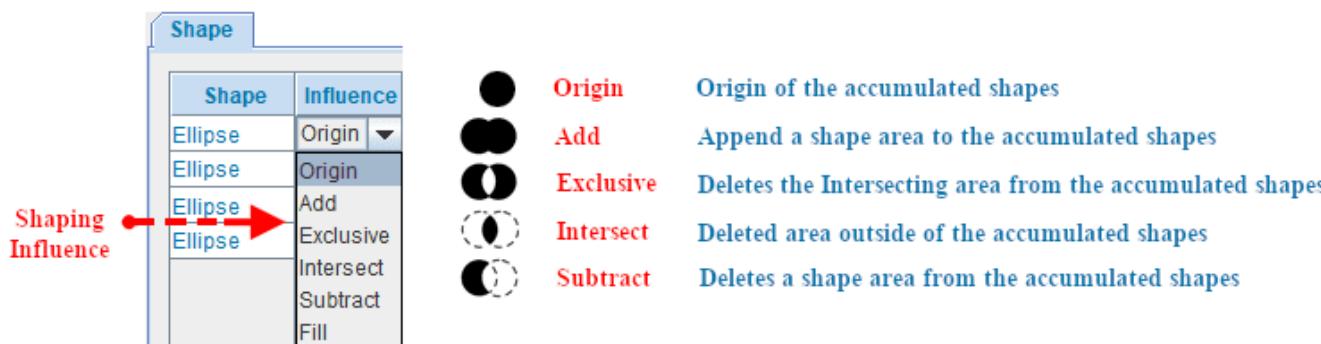
The  **Symbology Designer** has the necessary  utilities required to perform the common place manipulation of  **SVG**. In order to understand how to make use of its full functionality acquaint yourself with the instructional usage in the  help tutorial and try your hand out at some experimentation by making use of  design templates

Shape to Form Customize a  design approach to constructing forms according to the complexity of their pre-requisite which is established by the structure of their contour
There are basically 3 types of forms that can be  designed, each having their appropriate design strategy but can also conveniently be used in combination :



1. **Shaped Forms** (accumulate Archetypal Shapes : Rectangle, Ellipse, Triangles and Diamond with implicitly defined area drafted by a single drag motion)
2. **Pathed Forms** (accumulate Path Shapes : Linear Hand, Polygon, Free Hand and Fluid with explicitly defined area drafted by a path of connecting points)
3. **Hybrid Forms** (accumulate Archetypal and Path Shapes : As per a more complicated combination of the previously mentioned forms of Shape and Path)

Shape Influence In order to understand how you can create forms effectively, you need to be acquainted with the principle of using shape areas to influence one another. The  archetypal shapes of rectangles, ellipses, diamonds and triangles can be combined to produce the required form by accumulating their respective areas (so creating a compound shape), according to specified influences as illustrated below indicated by the black ellipses. The alternative is using a single path shape.



Path Editor

The **Path Editor** facilitates the creation and maintenance of  **SVG Paths** allowing the various segments it consists of to be manipulated through the basic console of a menu and controls. A Path consists of individual segments that are joined together at their end points. The collection of primary segments are :

- **Move To** : Produces no visible segment and merely instructs the path to skip from the existing to another vertical and horizontal position.
- **Quadratic Curve** : A segment that has 2 end points and one control point which accentuated the curve into a specific direction and slope
- **Cubic Curve** : A segment that has 2 end points and two control points being the equivalent of 2 Quadratic curves with their own behavior
- **Line To** : A segment that has 2 end points which are joined by a straight line without any control points to manipulate the form of the line

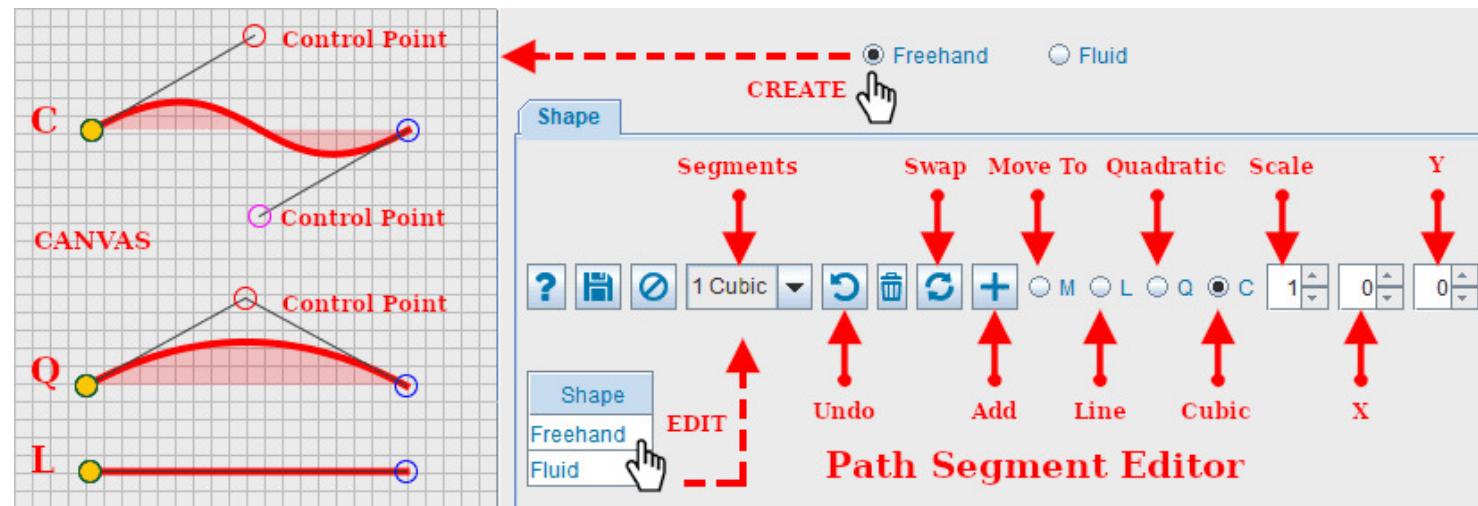
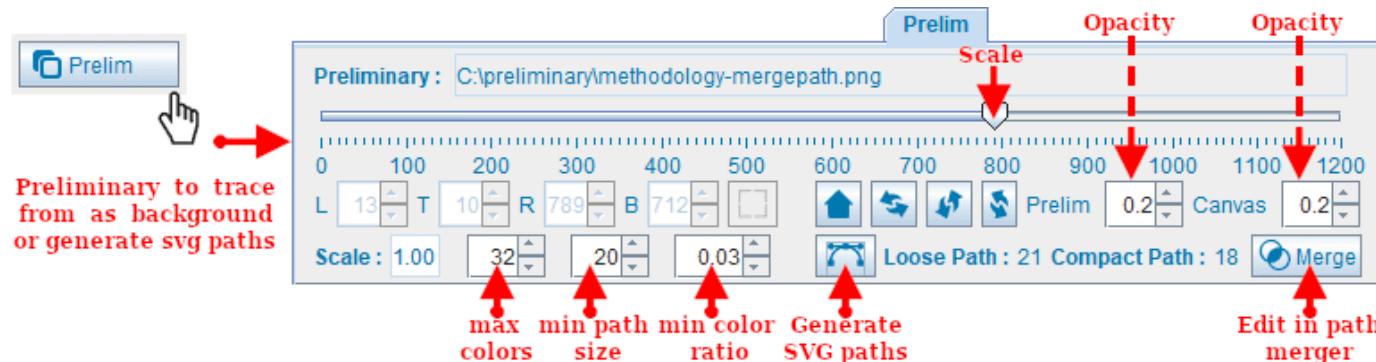
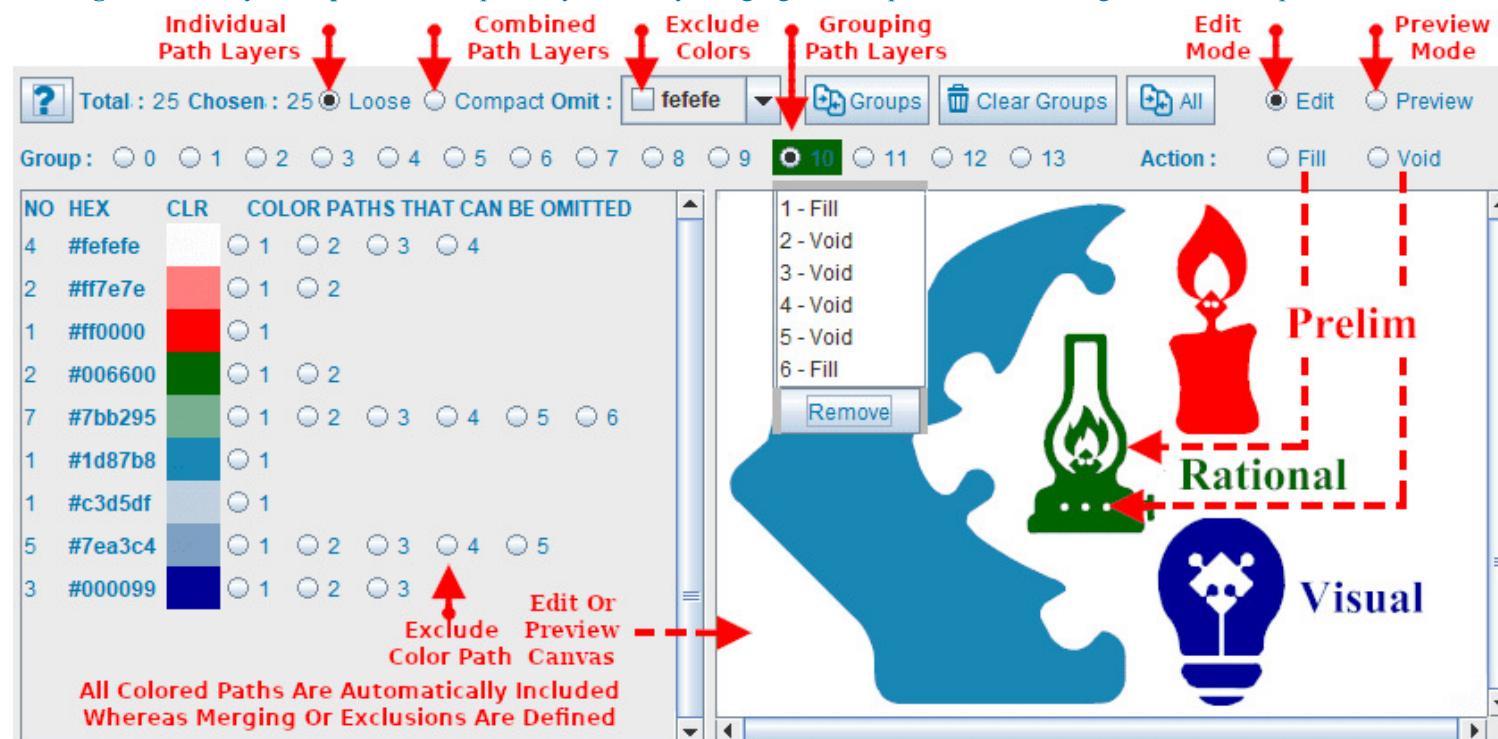


Image To SVG

The Preliminary Image can be imported by clicking the **Prelim Button** in the left navigation bar which then automatically selects the **Prelim Tab**.

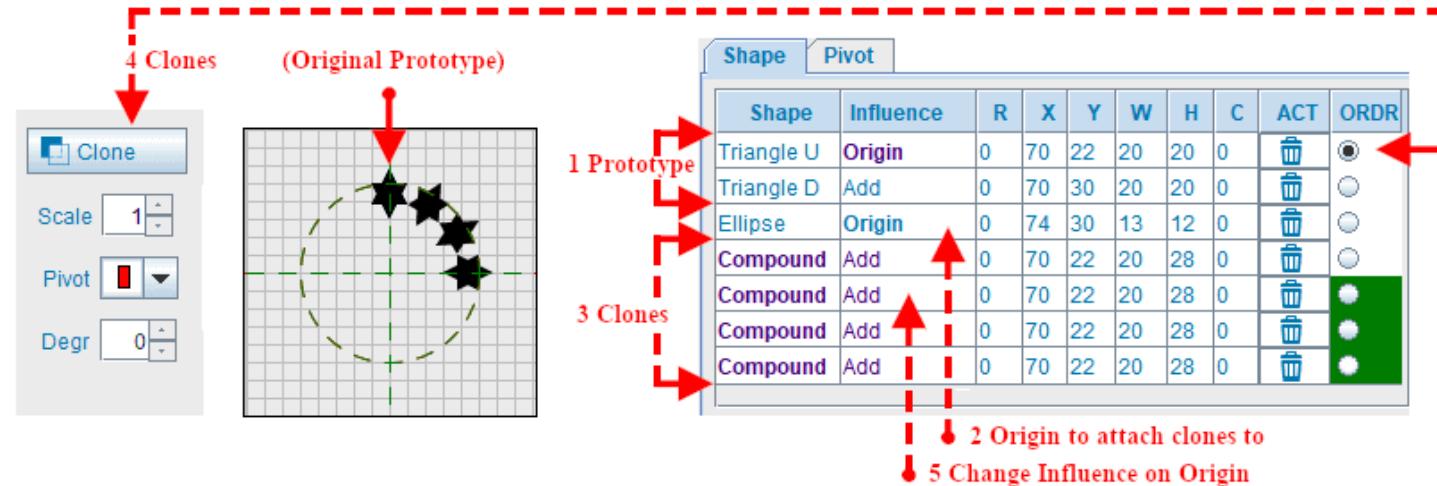


The **Path Merger** is accessed after importing a **Preliminary Image** as above then clicking the **Trace Button** and finally then the **Merge Button**. The **Path Merger** is a **utility** that **optimizes** the paths by manually merging wanted paths but excluding the unwanted paths in order to **shrink** it.



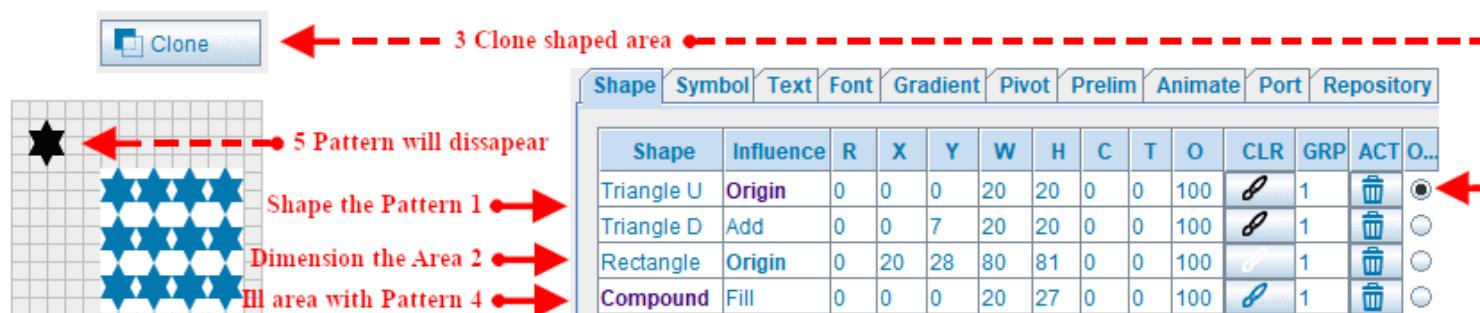
Prototype Cloning

When requiring a repeating visual element in your composition, create a prototype of it and then **clone** it so that it can be used in all those instances instead of having to repeat the effort of composing it multiple times for every needed occurrence thereof. Changes made to the prototype automatically cascades through to the clones. **clones** can also be manipulated further through **scaling**, **flipping** or **rotating** etc ... and also exert influence on other compound **shapes**



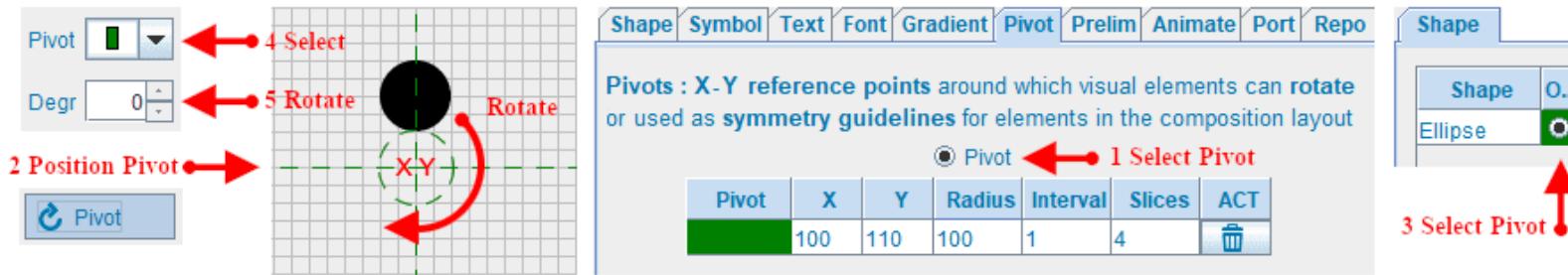
Patterning

Whenever a compositional **shape** must be tiled, create a **pattern** by composing a compound shape at the top left corner and fill the **target shape** with it. A **background color** must be specified for the **shape** being filled. Through this means you can have a background **pattern** without needing to repeat the process for every needed occurrence thereof. Any changes made to the **pattern** will automatically cascade through to all of the **shape** that are filled by it



Pivot Symmetry

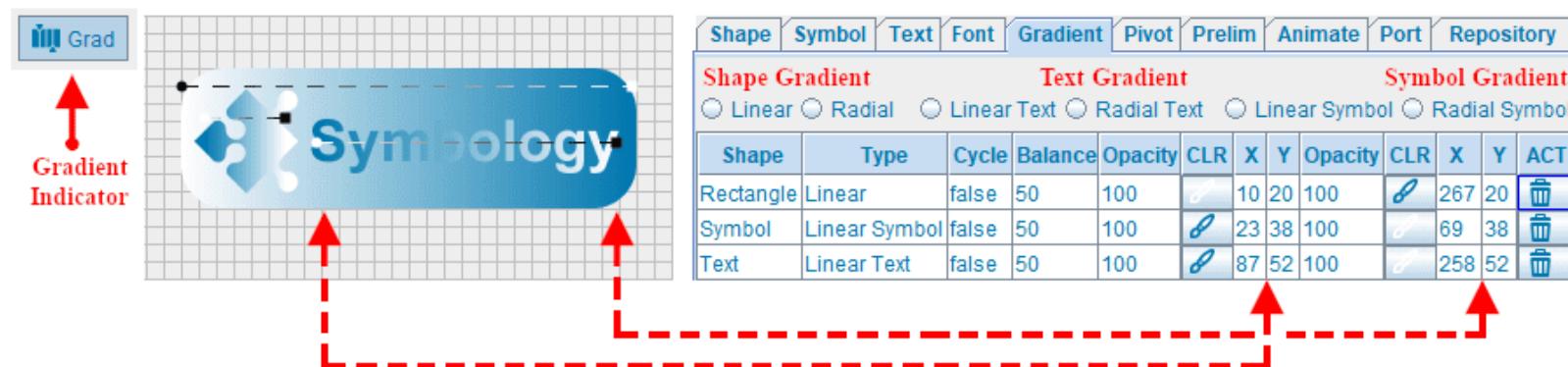
When needing radial symmetry in a composition (as opposed to the linear symmetry of the canvas gridding lines), its considered best practice to use the Pivot functionality to pivot all the compositional elements around a single central point making for a radially symmetrical layout that is consistently balanced



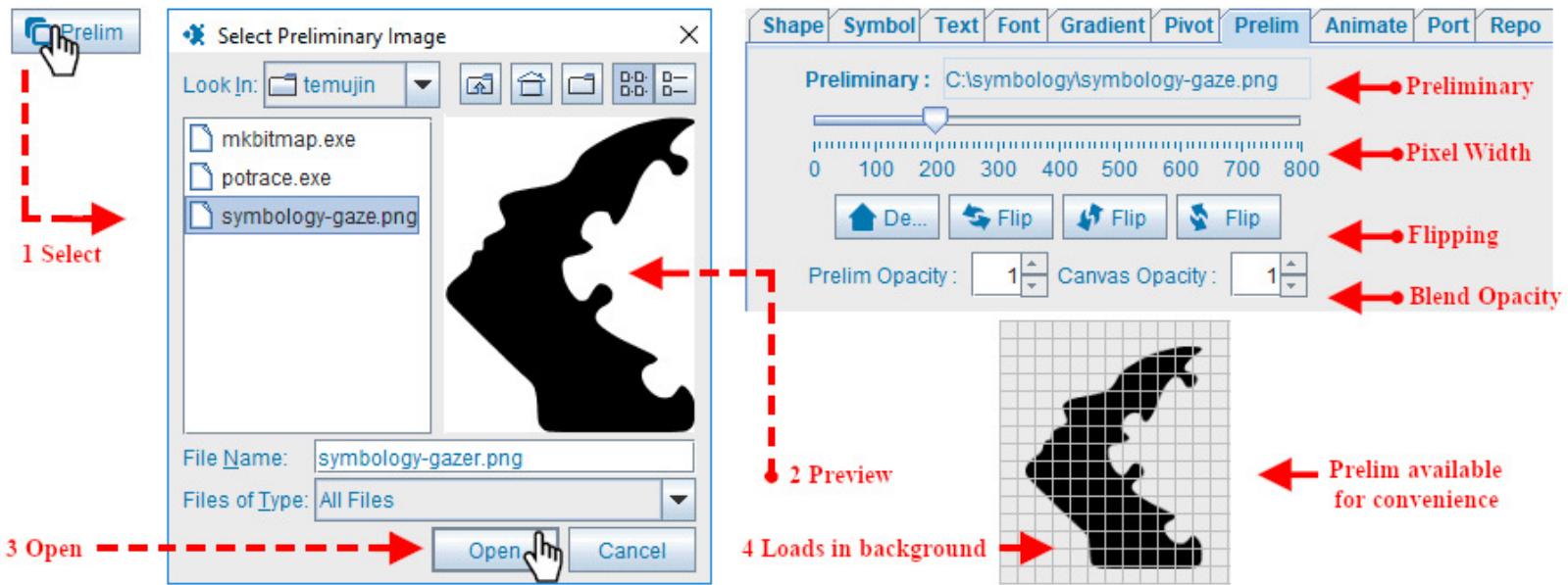
Gradients

Shapes, Symbols and Text can be painted with a color gradient rendering a gradual change from one color to another inside either a linear or radial expanse. The gradient type must first be selected from the below illustrated menu and either of the following should be followed depending on the element :

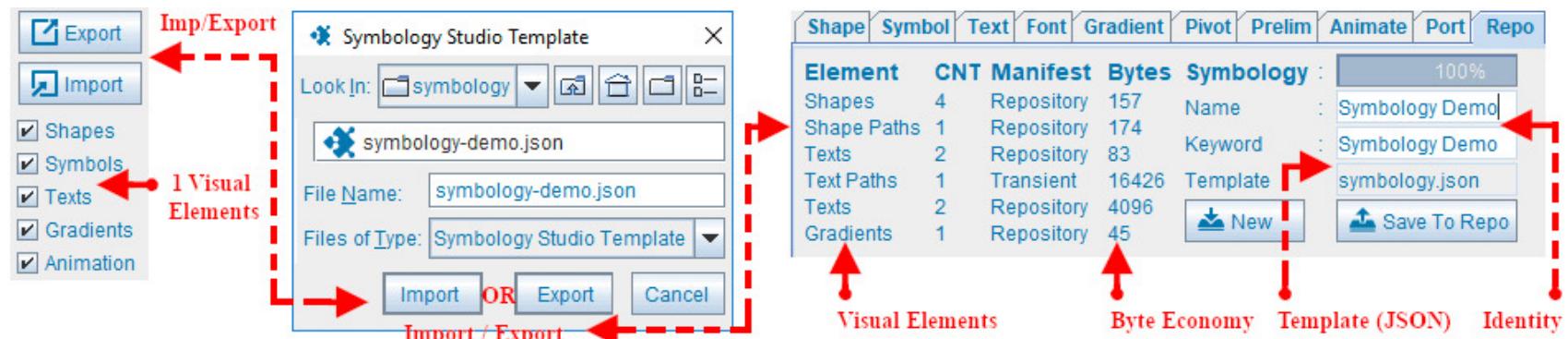
- If it is a **Shape Gradient** then the gradient start point is triggered by clicking and dragging on a shape in the canvas with a swiping motion.
- If it is a **Text Gradient** then the gradient start point is triggered by clicking and dragging on a text path in the canvas with a swiped motion.
- If it is a **Symbol Gradient** then the gradient start point is triggered by clicking and dragging on a symbol in the canvas with a swipe motion.



Preliminary When at all possible before starting to design forms in a composition, it is usually a good idea to have a preliminary image to work from. By means of this you can trace forms or use it as a visual guide while dimensioning and laying out all the elements of the composition. The Preliminary Tab Pane is there for this specific purpose, so use it whenever you can to allow for a convenient visual guide. Once the preliminary image is selected it will appear as background in the canvas and the Preliminary Tab Pane, also gaining focus where upon you can adjust relevant settings as per illustration

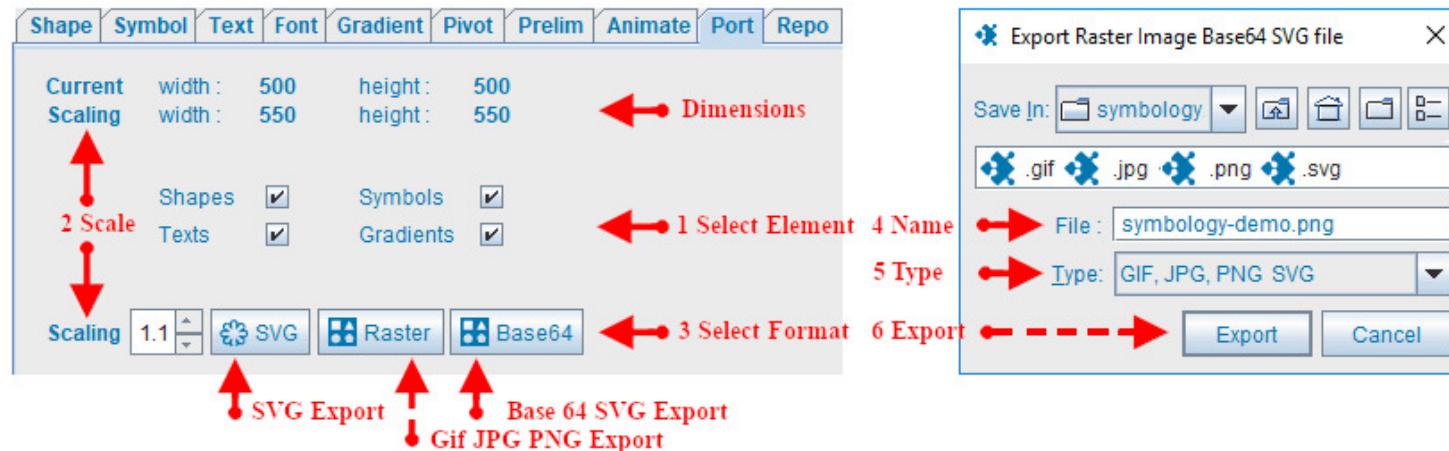


Templating As storage space in a repository is a legitimate concern, make an effort to avoid littering the repository with redundant entries. So when designing any composition (via Symbology or Symbol Designer), first create the template and save to it during your design sessions. Only once you are satisfied with the end result (preferably also subjecting it to an advised review for approval), only then should you save the finalized version to the repository



Graphic Ex/Import

Always  design keeping in mind a possible need of having to refine the  composition in future. Make the design large, so that it is easy to add more intricate details in a larger visual space later without having to re-adjust it to fit in the new details. Remember that it is  SVG which ideally shrinks or expands without any degradation so a large sized graphic will scale perfectly into a smaller or larger space. It will also have the same Byte Economy (memory footprint: file size) For your convenience the graphics can also be  ported to different dimensions in various formats (gif, jpeg, png and svg) which are accessible in the Port pane



Byte Economy

Always be aware of the various elements byte size while you are designing any  composition, specifically those of the path shapes and text paths. Out of the compositional elements these 2 are typically the largest and if not monitored can unintentionally grow to be unjustly excessive in size. The aim must always be to do as much with as little possible, keeping in mind that  byte economy is essential for a light payload to devices which is what you should ideally aim for.

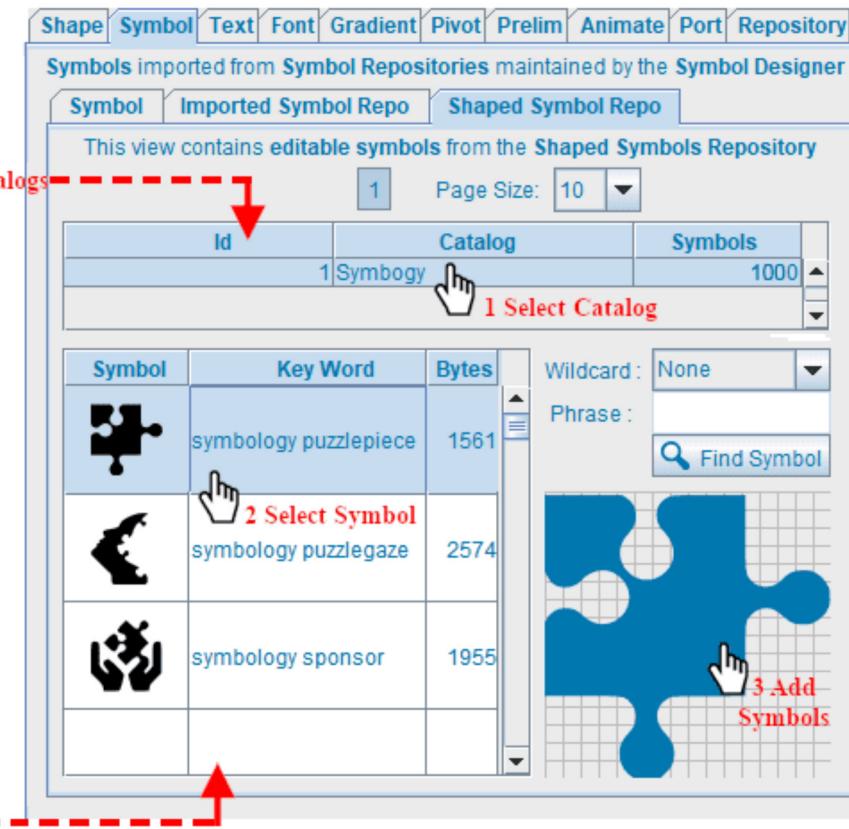
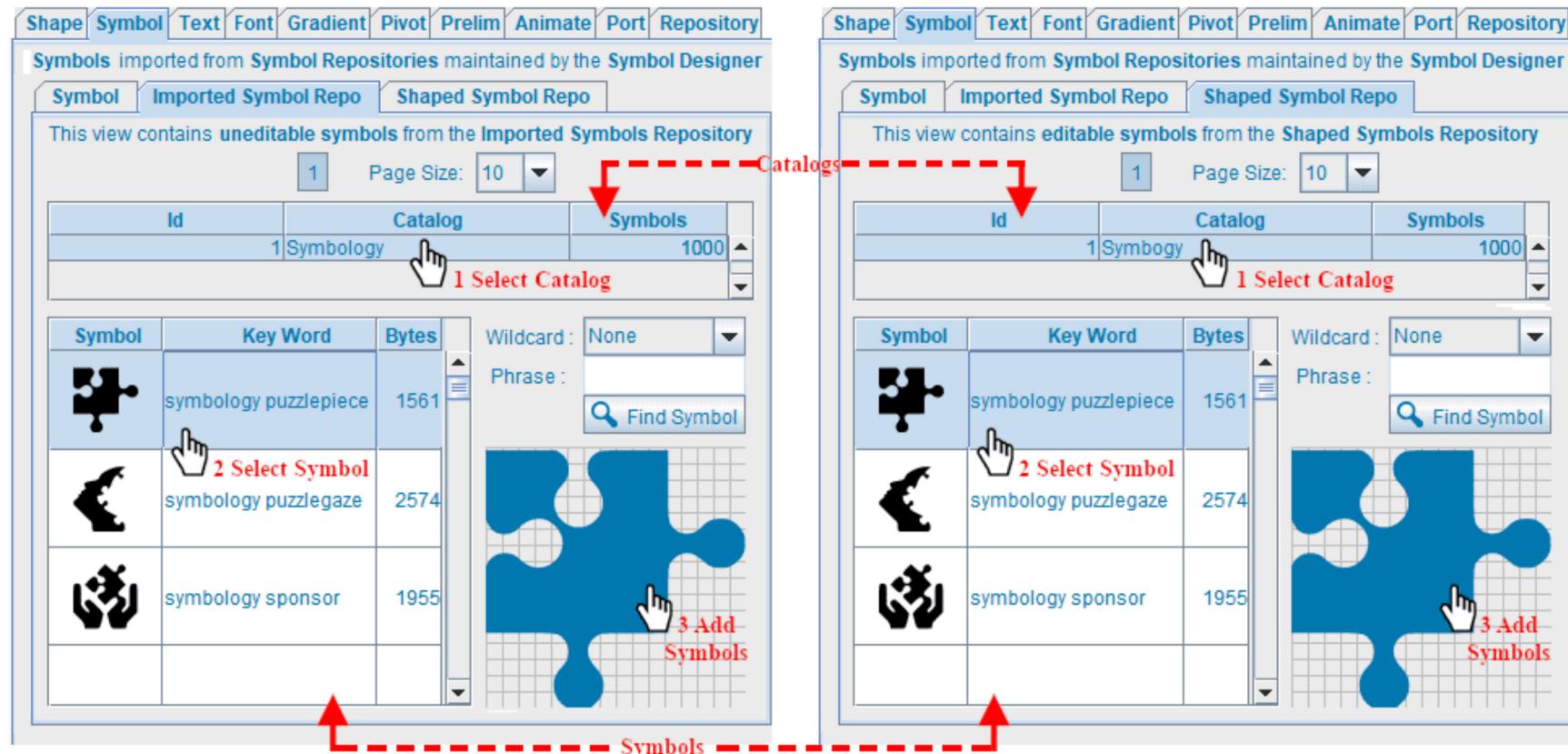
The screenshot shows the 'Repository' tab selected in the top navigation bar. On the left, a table lists 'Graphic Elements' with columns: Element, CNT, Manifest, Bytes. The data is:

Element	CNT	Manifest	Bytes
Shapes	4	Repository	157
Shape Paths	1	Repository	174
Texts	2	Repository	83
Text Paths	1	Transient	16426
Texts	2	Repository	4096
Gradients	1	Repository	45

Annotations include 'Graphic Elements' pointing to the table, 'Byte Economy' pointing to the 'Bytes' column, 'Template (JSON)' pointing to the 'Template:' dropdown, and 'Identity' pointing to the 'Save To Repository' button. On the right, there's a sidebar with fields for 'Name:', 'Keyword:', 'Template:', and buttons for 'New' and 'Save To Repository'.

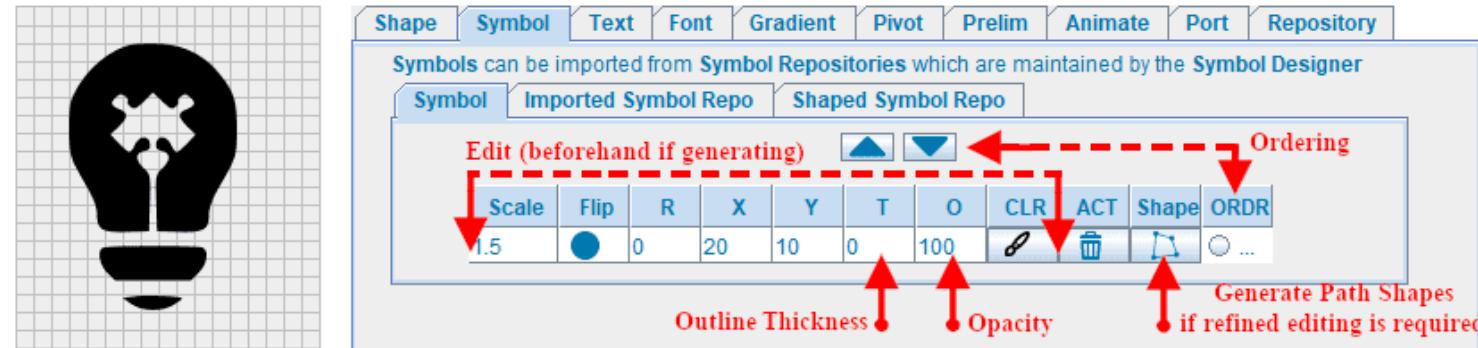
Symbol Importing

⌚ Symbols can be imported into compositions from the imported or shaped Symbol repositories. They are first located in their pane by searching for it with a phrase and wildcard or selecting the catalog which then renders a list of ⌚ symbols to be selected from and added for editing in the Symbol pane



Symbol Handling

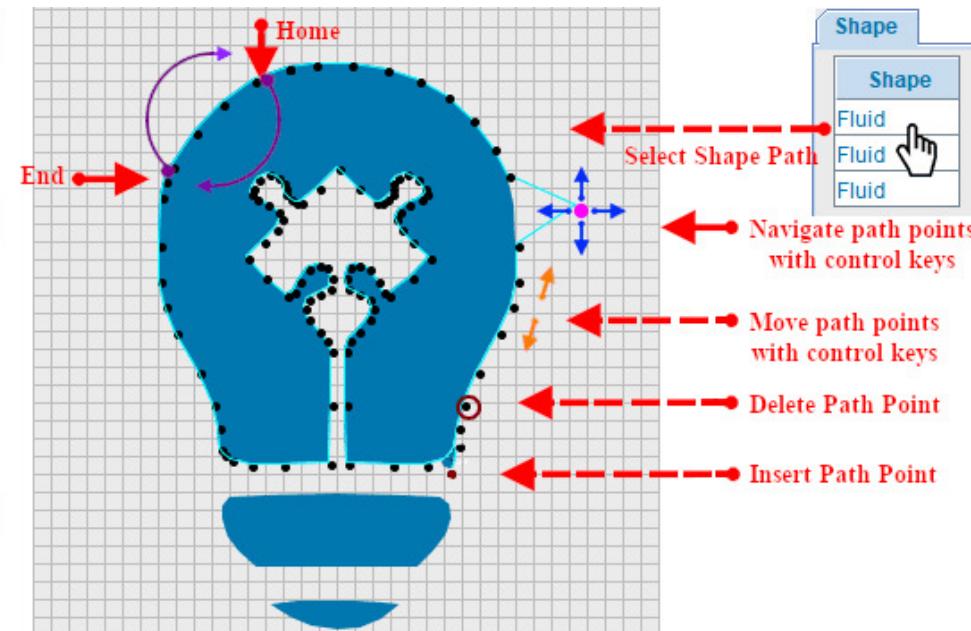
A  symbol is manipulated in the Symbol pane by  scaling,  flipping,  rotating,  positioning,  coloring or  opacity. If further manipulation of its structure is needed it can be converted into a composite shape of  fluid shapes (path shapes) and the applicable influences on one another by pressing a Shape button in the Symbol pane (for a specific symbol) which then adds them as actual  shapes to the Shapes pane where they can be manipulated further



Symbol Editor

The structure of  path shapes (linear hand, polygon, free hand and fluid) can be manipulated further by clicking the desired Shape field in the Shapes pane. The Manipulation Guide will display a help pane indicating the manipulation types available in the editor. The structure of the path can then be navigated and changed through the use of control keys mentioned in the help pane. Note that all colors in the illustration correlate with the colors in the Manipulating Guide.

Path Navigation	
Home	Step Beginning
End	Step End
Space Bar	Step Forward
Backspace	Step Backward
Path Manipulation	
Up	Move Up
Down	Move Down
Right	Move Right
Left	Move Left
Insert	Insert Point
Delete	Delete Point
Ctrl	Extend Points
Saving And Exit	
Enter	Exit Saved
Esc	Escape Unsaved



Font Selection

Only use Text Paths on rare occasions when you need to make a strong visual impact that can only be satisfied by a unique **A** font. When you only want to render plain text for information purposes only, rather opt for one of the first 5 **A** system fonts. These will not be rendered by weighty paths but are instead rendered as plain text with negligible bytes. Give some attention to the font properties view because it provides all of the pertinent information regarding it.

Unshaped Text

Phrase ... 4 Capture Text

Drop Text Into Shape

Placement Markers

5 Configure

Marker moves Shaped Text along the shapes perimeter

Rotate Outwards or Inwards

Outline Thickness

Opacity

Shape Symbol Text **Font Gradient Pivot Prelim Animate Port Repository**

TTF (True Type Font) are custom fonts that can be installed onto an operating system or can be referred to by their file path and thus be rendered on any supporting devices Select the Text Type below and drag the necessary Font into position over the Canvas

1 Designate Text Shaped Text (Follows a shape's perimeter)

TTF Download TTF Dir C:\webstudio\fonts-master\fonts-master

Name : Aclonica Regular Manifest : Type : TTF Size : 68 KB

Glyphs : 371

Phrase : 5591 Bytes **Byte Economy**

Display : Style : Plain Size : 16

Phrase(s) : Phrase

ABCDEFGHIJKLMNOPQRSTUVWXYZ

abcdefghijklmnopqrstuvwxyz

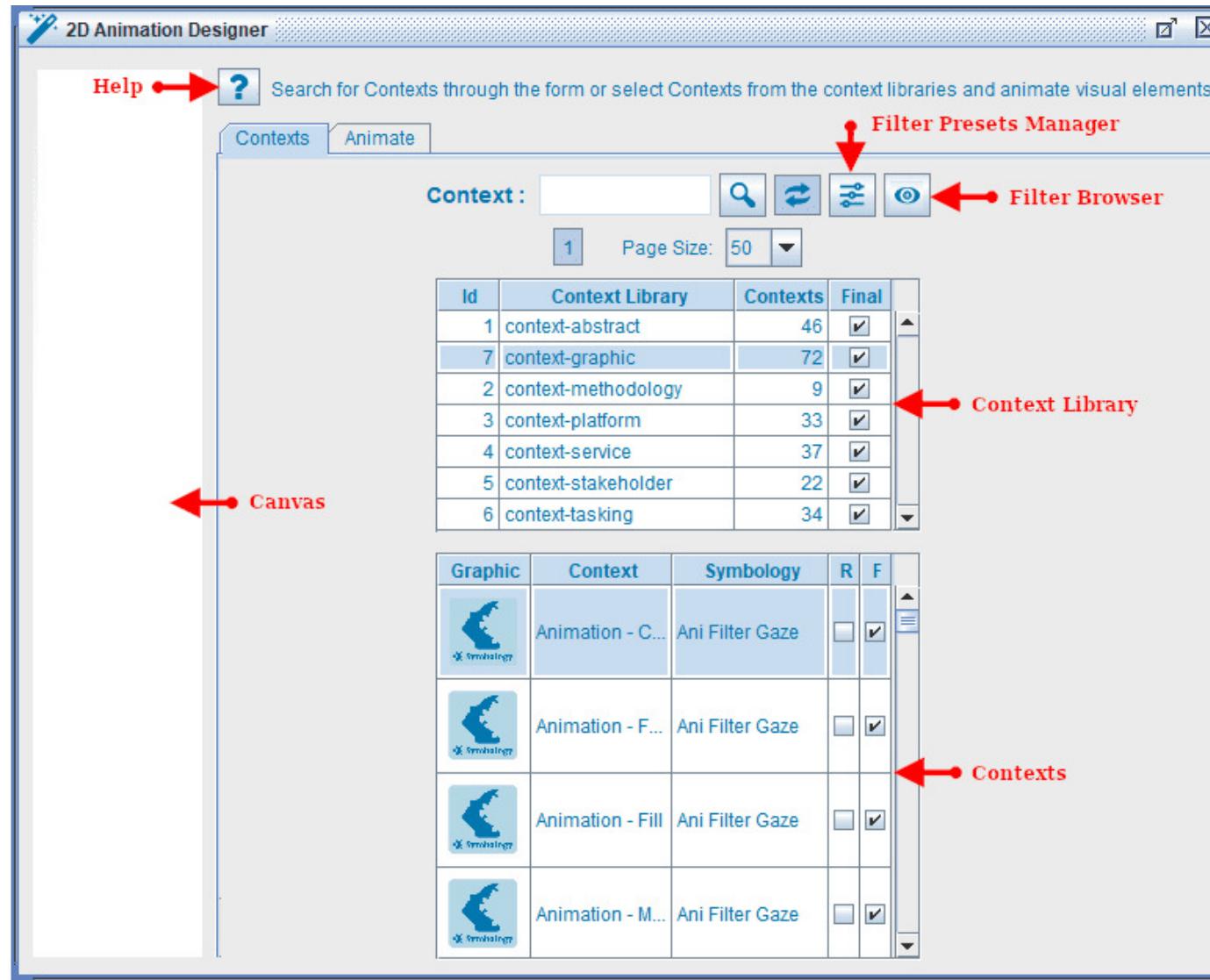
1234567890 !?@#\$%^&*

X	Y	MRKR	Skew	R	Font	Style	CLR	Size	T	O	Phrase	ACT	O...
100	100	0	None	0	Aclonica-Reg...	Plain	0	16	0	100	Unshaped T...	<input type="radio"/>	
145	229	-5	Out	0	Aclonica-Reg...	Plain	0	16	0	100	Shaped Text	<input type="radio"/>	

↗ Animation Designer : Help

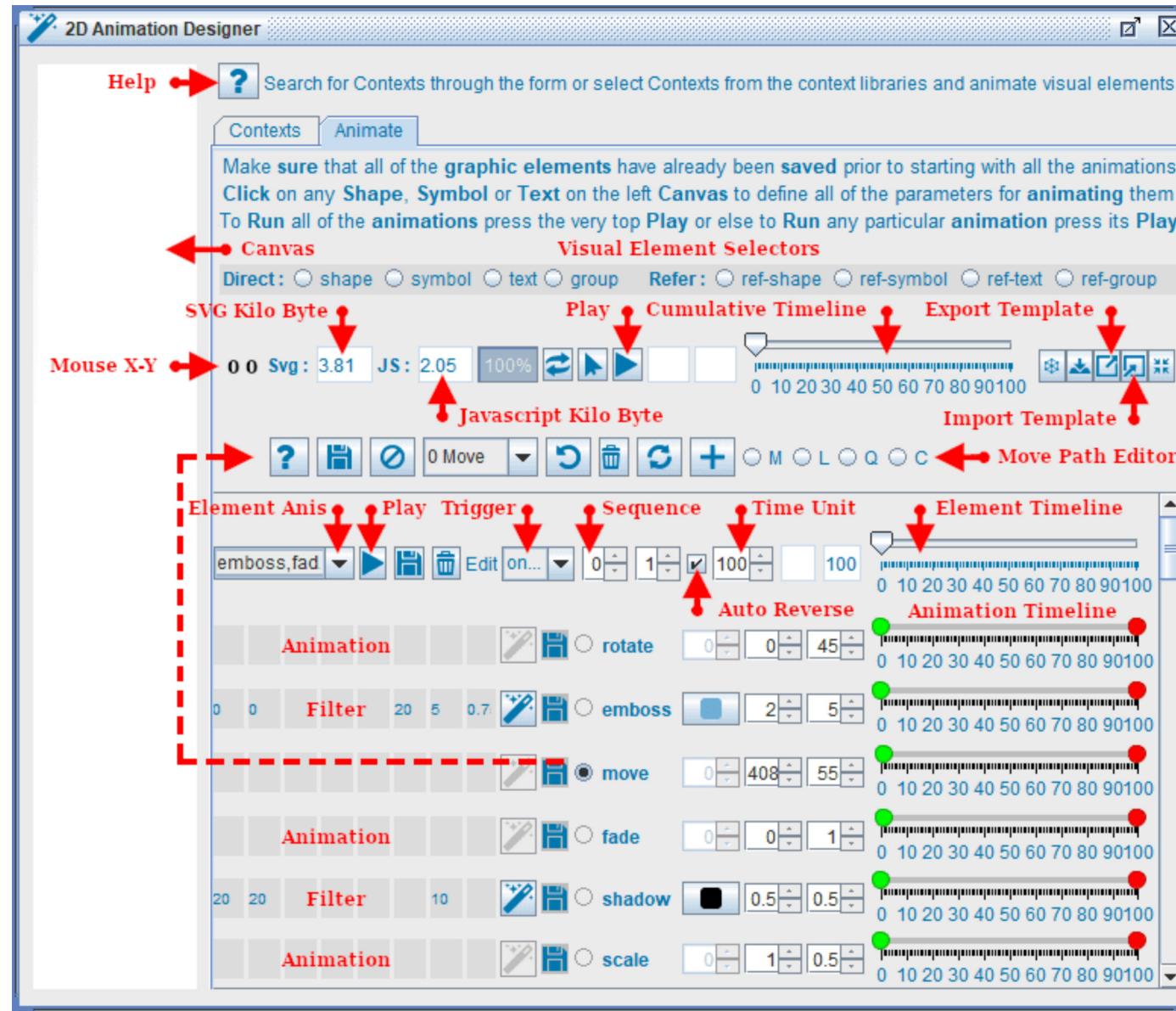
Animation Navigator

The ↗ **Animation Designer** has a mechanism to navigate ⇢ **Contexts** and select one to add dynamic behavior to ⇢ **SVG** compositions by decorating visual elements with ↗ **filters** and assign event driven ↗ **animations** as well. There are 25 distinctly different **filters** which are configurable into many permutations and then stored as presets in the **Filter Presets Manager** for reuse. A property steadily transforming a visual element can be configured.



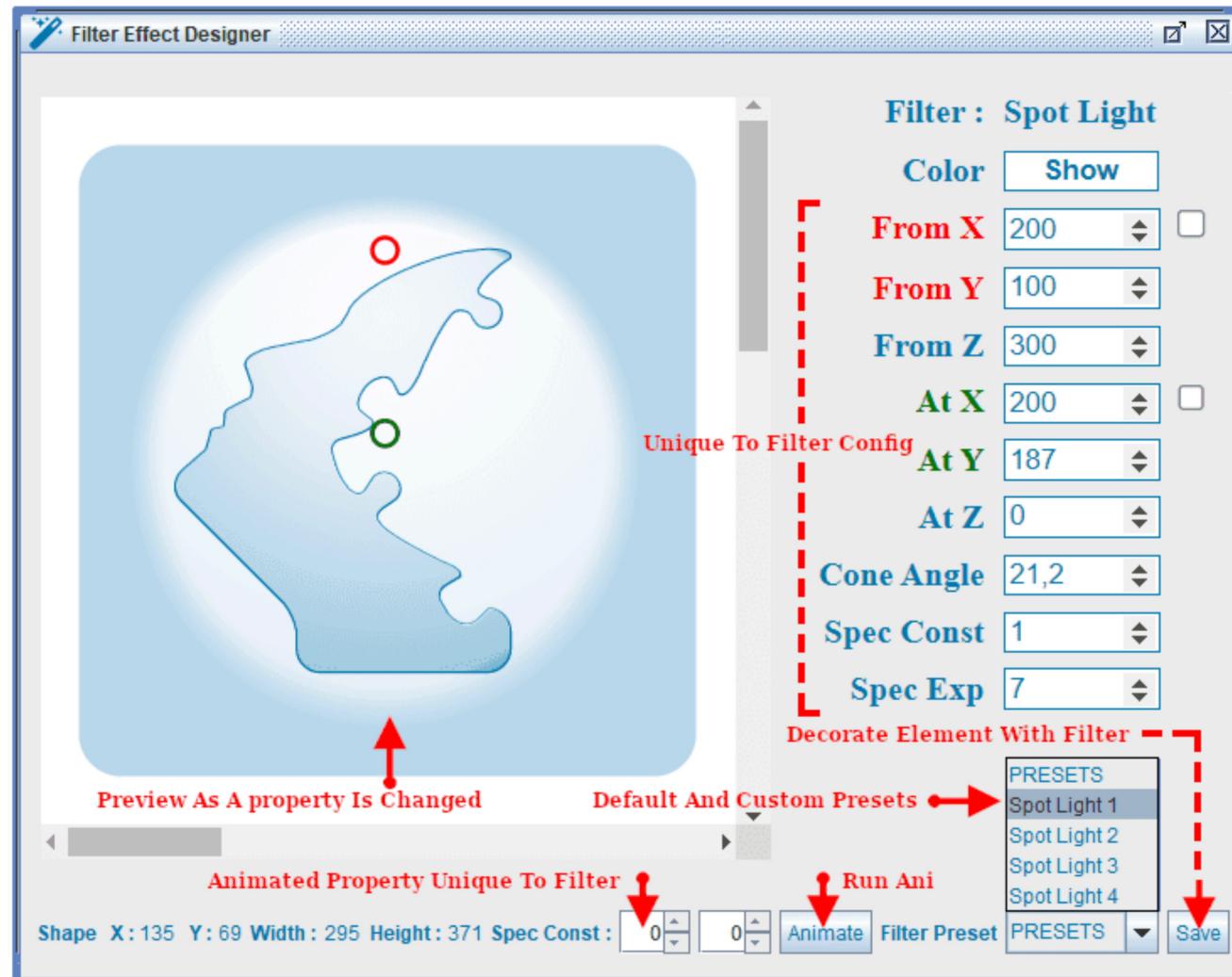
Animation Designer

Once a  **Context** is selected dynamic behavior can be composed by checking a selector and clicking on its corresponding element on the **canvas**. A console is added where controls can manipulate the element over a timeline. A  **Filter Designer** is then accessed through clicking a  **Wand**.



↗ Filter Designer : Help

The ↗ **Filter Designer** will be populated with the appropriate controls that are specific to the filter being designed. As the properties are edited the canvas on the left will then respond accordingly as the values are changed. The visual element can then be **gradually transformed** over a **timeline** according to the ↗ **configured behavior** whether it be in or out of the **decorated state** and whether it will then reverse or not into their **plain state**.



Filter Manager

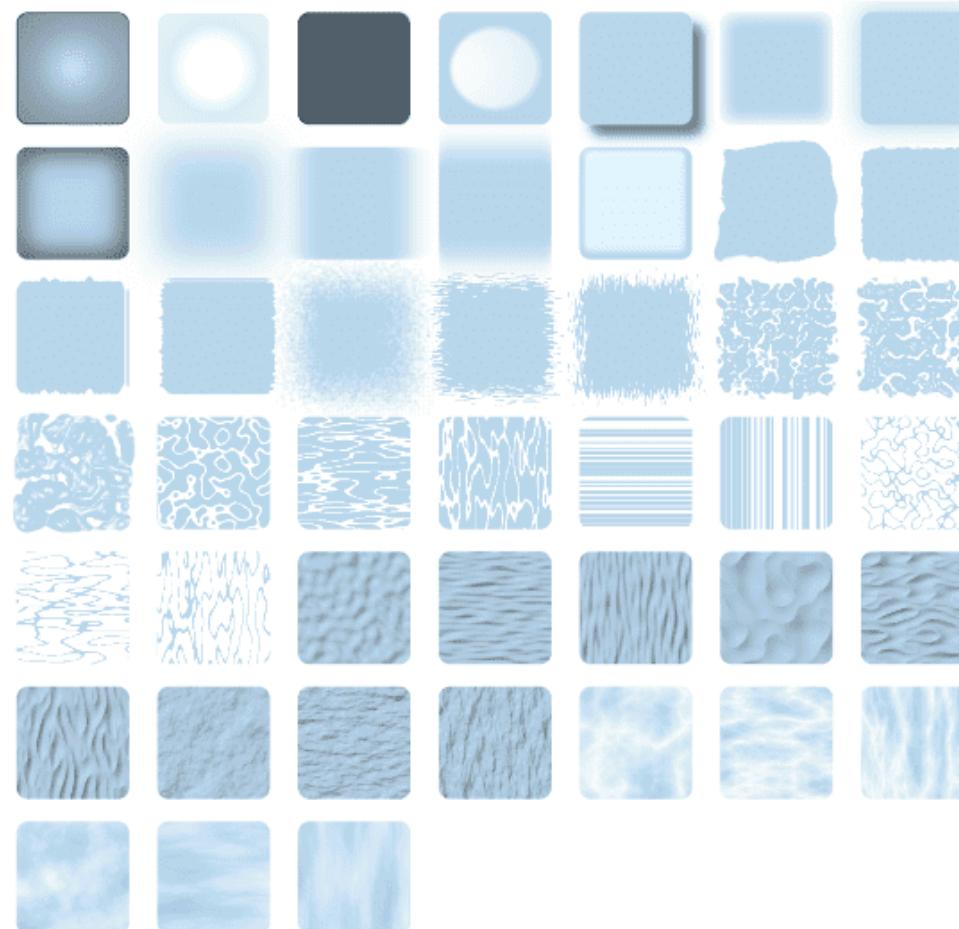
The  **Filter Presets Manager** facilitates the location, organization, viewing and saving of filter presets to **templates** that are automatically detected by the Studio application. Not all templates will be re-usable as the dimensions and positions of visual elements will differ according to their design in compositions. Unlike lighting templates only solid and texture filter configs will be re-usable as their parameters are relevant to the elements position.

Filter Preset Designer

FILTERS		<input type="checkbox"/> Random	<input type="checkbox"/> Align	Config Filter	Presets	Open Filter Presets	Save Filter Presets	Clear Filter Presets					
Name	Type	Spec	Start	End	Par1	Par2	Par3	Par4	Par5	Par6	Par7	Par8	Del
Point Light 1	pointLight	#fffff	40	0	200	200							Del
Spec Light 1	specLight	#fffff	40	0.0	200	200	280						Del
Dist Light 1	distLight	#fffff	0	360					30				Del
Spot Light 1	spotLight	#fffff	1	0.0	0	200	300	20	170	200	0	7	Del
Spot Light 2	spotLight	#fffff	1	0.0	200	100	300	21.2	200	187	0	7	Del
Spot Light 3	spotLight	#fffff	1	0.0	310	200	300	22	220	200	0	7	Del
Spot Light 4	spotLight	#fffff	1	0.0	200	310	300	21.2	200	215	0	7	Del
Shadow 1	shadow	#0000...	0.4	0.0	20	20					10		Del
Shadow 2	shadow	#0000...	0.3	0.0	15	15					5		Del
Bright 1	bright	#fffff	10	0				0.8			10		Del
Bright 2	bright	#fffff	5	0				0.8			5		Del
Glow 1	glow	#b9d8...	20	0				0.8			10		Del
Glow 2	glow	#b9d8...	10	0				0.8			5		Del
Burn 1	burn	#0000...	10	0				0.8			10		Del
Burn 2	burn	#0000...	5	0				0.8			5		Del
Blur 1	blur		10	0									Del
Blur 2	blur		5	0									Del
Motion 1	motion		10	0	0				10	0			Del
Motion 2	motion		0	10	0				0	10			Del
Emboss 1	emboss	#fffff	4	0.0	0	0				5	0.75	20	Del
Emboss 2	emboss	#fffff	6	0.0	1	1				4.6	0.75	20	Del
Warp 1	warp		20	0				0.01	0.01	turb		3	Del
Warp 2	warp		20	0				0.01	0.01	frac		3	Del
Torn 1	torn		5	0				0.09				4	Del
Torn 2	torn		13	0				0.27				1	Del
Serrated 1	serrated		10	0				0	0.01			1	Del
Serrated 2	serrated		6	0				0	0.15			1	Del
Serrated 3	serrated		50	0				0.01	0			1	Del
Serrated 4	serrated		12	0				0.15	0			1	Del
Splash 1	splash		0.05	0.0						30	0	1	Del
Splash 2	splash		1.7	0.0						30	0	1	Del

Filter Previews

The **Filter Previewer** facilitates a quick short cut to preview the default filter presets which can be further refined to a desired effect. There's 45 presets already available satisfying most required visual effects. These presets will automatically be detected by the **Filter Designer** when it is opened from the **Filter Presets Manager** or **Animation Designer**. We highly recommend patient experimentation with these configurations to create a wider range of effects that you can use to liven up your compositions. However these may improve the look and feel of the compositions always be aware that other browsers viewing them might not have the same spec of resources as the machine you are authoring them on and might not respond with the same performance. Also be aware the **dynamically generated java script** adds significantly to the networks **HTML payload**.



Designer

Facilitates a mechanism whereby ❁ domain imagery of ☐ repositories can be navigated through from different perspectives re-using a ✎ composition in a specified ❁ context temporarily configuring visual elements to ↗ link to other ✎ compositions used in a particular ❁ context applying the same principles

The screenshot illustrates the Scavecgraph Context Designer interface, which facilitates navigating through domain imagery and repositories using contexts.

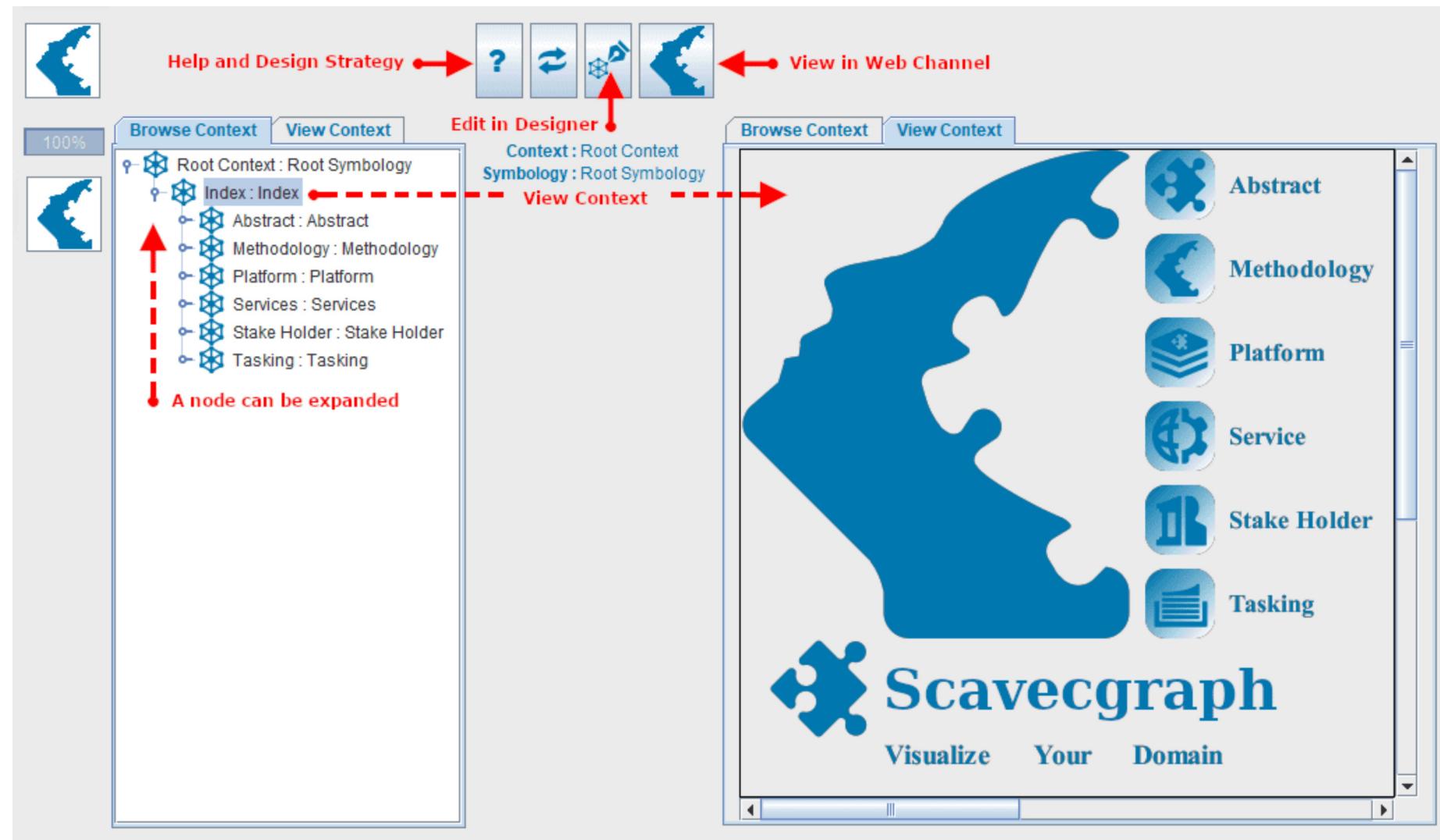
Browse Context: This panel shows a search interface with fields for "Context" and "Find". It includes a "Page Size" dropdown set to 50. A table lists contexts such as "context-abstract", "context-graphic", and "context-methodology". Below this is a table for managing symbols, showing columns for "Graphic", "Context", "Symbology", and "Root". A "Save Linking" button is present. A "Element" table lists items like "SHPDSMBL" linked to "Abstract", "Methodology", "Platform", "Services", and "Stake Holder".

Link Context: This panel also has a search interface with "Page Size" set to 50. It lists contexts like "context-abstract", "context-graphic", and "context-methodology". Below is a table for managing symbols, showing columns for "Graphic", "Context", "Symbology", "Root", and "Act". A "Save Context" button is present. A "Symbol" table lists items like "symbolology-abstract", "symbolology-graphic", "symbolology-methodology", and "symbolology-platform".

Central Canvas: The central area features a large blue puzzle piece graphic. A central text area says "Scavecgraph" and "Visualize Your Domain". Red annotations explain the process:

- 1 Search OR 1 Select: Points to the search and selection fields in the top left.
- 2 Select populates canvas: Points to the central puzzle piece.
- Populates: Points to the table below the puzzle piece.
- 3 Drag onto Canvas: Points to the "Link Context" panel.
- 4 Save Context Link: Points to the "Save Linking" button in the "Browse Context" panel.
- Populates Linked Contexts: Points to the table in the "Link Context" panel.

Browser Facilitates navigating the  domain imagery of a  repository through a hierarchical menu with expandable nodes simulating navigation from the current parent  context (containing  link elements) to its child  contexts (applying the same principle) effectively viewing the  compositions from different perspectives essentially re-using imagery of the  domain. To view the actual imagery click on the name of the  context as opposed to expanding its node



Designer Facilitates temporarily attaching an **article** (substantial body of text) to a graphic composition through contextualizing it in a given **context** (wrapping the composition). In this way substantial bodies of text in the **subject matter** content can be re-used in the **domain** imagery from different perspectives

The screenshot illustrates the Article Designer interface with two main panels: **Context** and **Article**.

Context Panel: This panel contains a search bar with a magnifying glass icon and a dropdown menu, labeled **1 Search**. Below it is a table titled "Context Library" with columns "Id", "Context Library", and "Contexts". The rows show entries: "1 context-abstract" (35 contexts), "7 context-graphic" (38 contexts), and "2 context-methodology" (9 contexts). A second table below shows "Graphic", "Context", "Symbology", and "Root" columns. The first row has "Abstract" in all columns. The second row has "Archetypal" in all columns. A red arrow labeled **1 Select** points to the "Abstract" entry in the first row. A red arrow labeled **Populates** points to the "Abstract" entry in the second row. A red arrow labeled **2 Select populates Canvas** points to the "Abstract" entry in the second row.

Article Panel: This panel contains a search bar with a magnifying glass icon and a dropdown menu, labeled **1 Search**. Below it is a table titled "Article Library" with columns "Id", "Article Library", and "Articles". The rows show entries: "1 Symbology Demo" (7 articles), "3 testlib3" (3 articles), and "4 testlib4" (3 articles). A red arrow labeled **3 Contextualize** points to the "Contextualize" button at the bottom right of the panel.

Central Area: A large central area displays the **Abstract** article content. It features the title "Abstract" with a puzzle piece icon, followed by the text: "The abstract aspects of this platforms solution is define by certain proprietary concepts and underlying constr...". Below this is a section titled "Objective Concepts" with icons for objective and concepts. Further down, the text states: "The platforms abstractions consist of two primary constructs namely a Linguistic and Graphic model with their secondary constructs of Narrative, Symbolism, Symbology and Context". At the bottom of the central area, there is a summary: "The platforms abstractions consist of two primary constructs namely a Linguistic and Graphic model with their secondary constructs of Narrative, Symbolism, Symbology and Context".

Bottom Buttons: At the bottom of the interface are several buttons: "Save Contextualized Articles" (highlighted with a red arrow labeled **5 Link Article**), "Contextualize" (highlighted with a red arrow labeled **3 Contextualize**), "Update", "Clear", and a trash bin icon.

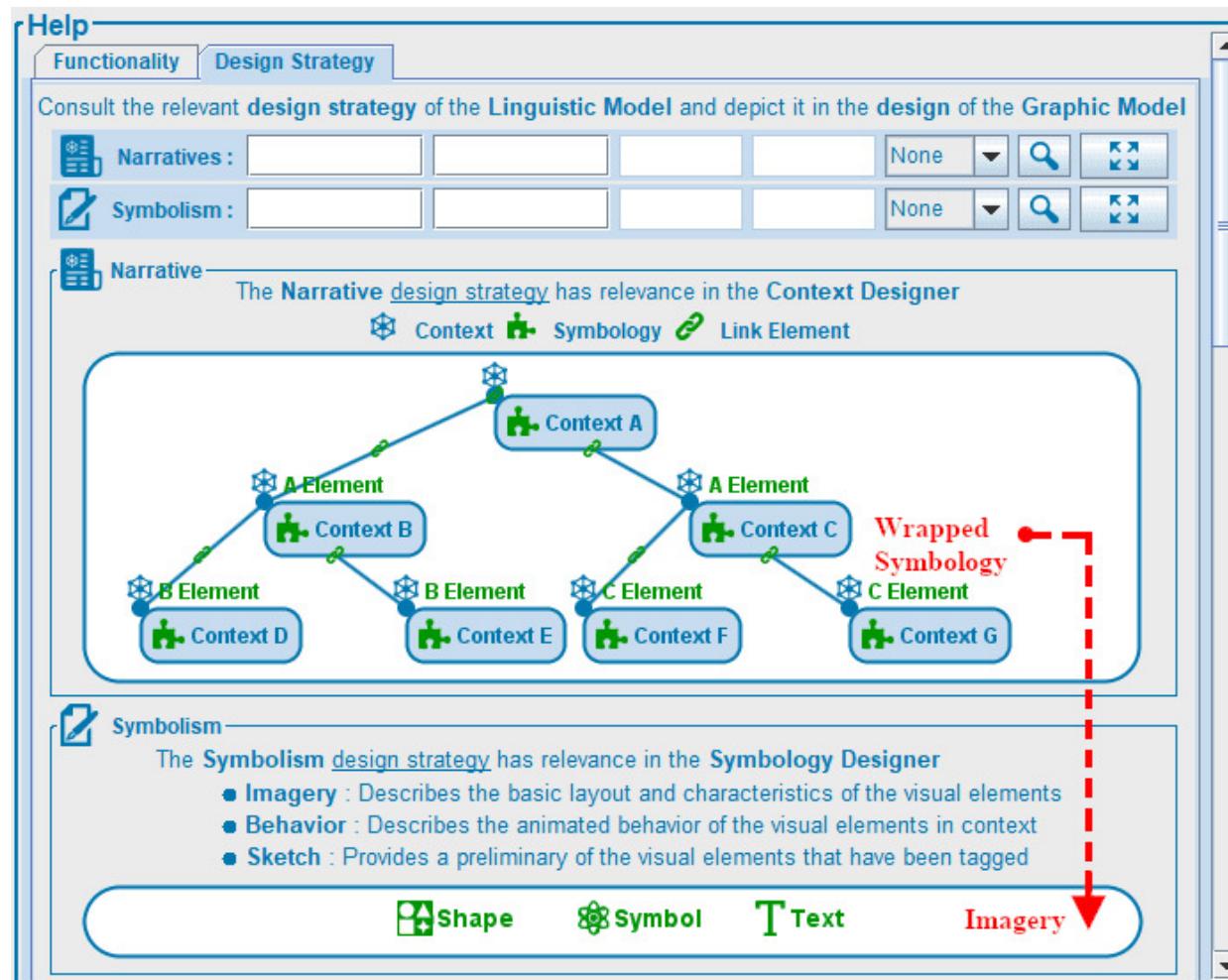
Design Strategy

A  **Linguistic Model** of specs (defined by domain expert) is comprised of 2 constructs specifying intended graphic content and their inter linking navigation

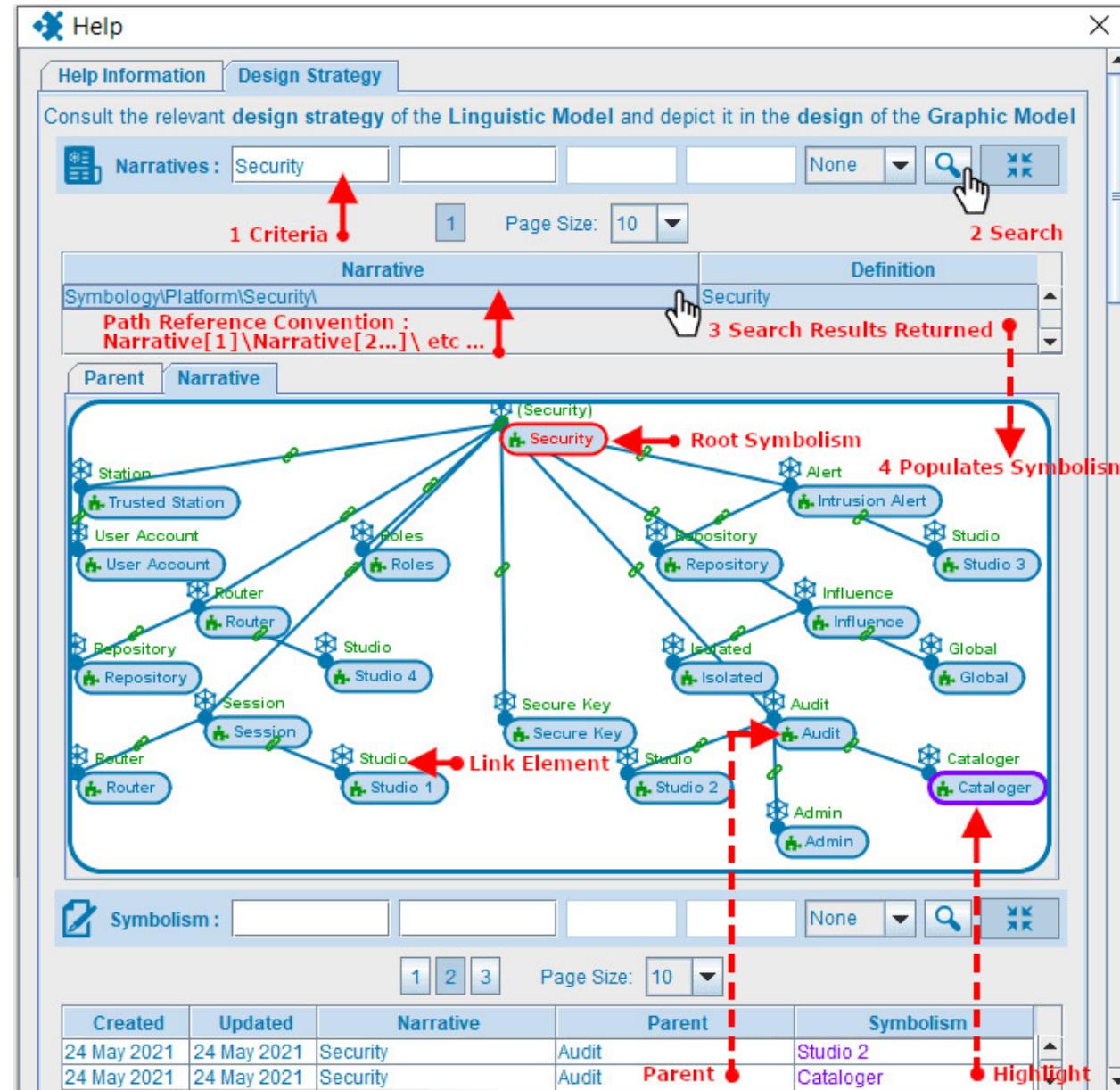
 **Narrative** : specifies the linking navigation of the graphic compositions used to visualize the imagery of the domain (relevant in the  context Designer)

 **Symbolism** : specifies the visual elements of graphics compositions used to visualize the imagery of the domain (relevant in the  symbology designer)

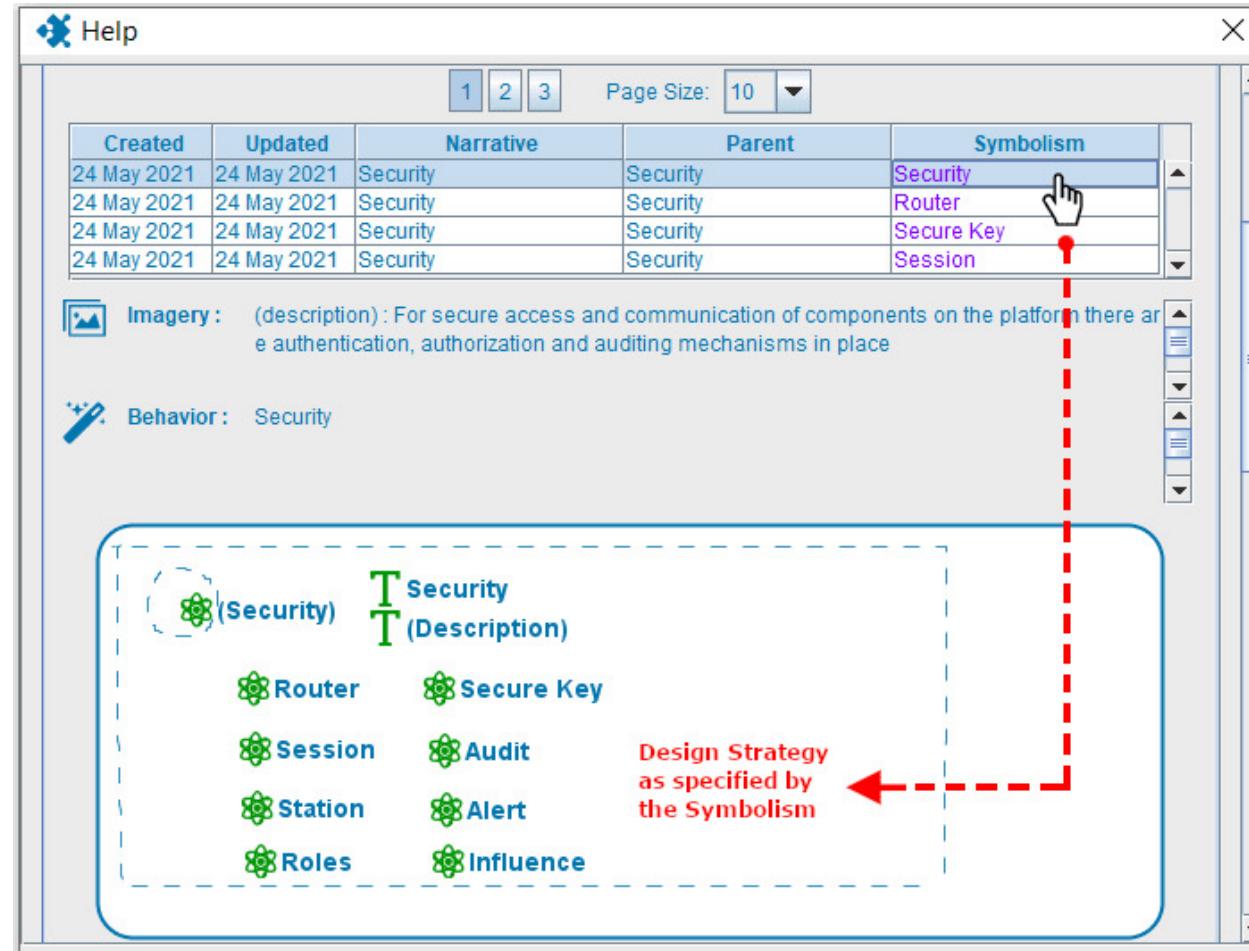
It can be accessed from the  help button that features in all the  designers and  browser, as well as the  in-tray located in the  studio app interface
Facilitates a mechanism where by graphics specifications created by a  domain expert (knowledgeable of domain) can be referred to as a  design strategy when a  graphic designer (no knowledge of domain) creates the  compositions also temporarily  loosely links them to each other through  contexts



■ **Narrative** : specifies linking of symbolism translated into contexts in the **Context Designer**



☒ **Symbolism** : specifies the symbolism translated into symbology done in the  **Symbology Designer**



The screenshot shows the Symbology Designer interface. At the top, there is a table with columns: Created, Updated, Narrative, Parent, and Symbolism. The rows show various security-related items: Security, Router, Secure Key, and Session. Below the table, there are two sections: 'Imagery' and 'Behavior'. The 'Imagery' section contains a description about authentication, authorization, and auditing mechanisms. The 'Behavior' section is labeled 'Security'. At the bottom, there is a large callout box containing a grid of symbols and their corresponding labels: Security, Router, Session, Station, Roles, Description, Secure Key, Audit, Alert, and Influence. A red arrow points from the text 'Design Strategy as specified by the Symbolism' to the grid.

Created	Updated	Narrative	Parent	Symbolism
24 May 2021	24 May 2021	Security	Security	Security
24 May 2021	24 May 2021	Security	Security	Router
24 May 2021	24 May 2021	Security	Security	Secure Key
24 May 2021	24 May 2021	Security	Security	Session

 **Imagery:** (description): For secure access and communication of components on the platform there are authentication, authorization and auditing mechanisms in place

 **Behavior:** Security

 (Security)	 Security
	 (Description)
 Router	 Secure Key
 Session	 Audit
 Station	 Alert
 Roles	 Influence

Design Strategy
as specified by
the Symbolism

Work History

The full  activity history is available for  querying the creation or changes that have been made to any of the graphic entities in the  repositories. The graphic entities that have been created or edited in  Studio App can be navigated to in their applicable  designer for further editing if required. A full  audit trail is kept for all activity of consequence on the  platform to provide information for investigation or aid in accountability measures.

A full Audit History of all the Users sessions and their specific activity on the system are available here. Simply search for the Session and select it to list all the changes the User has made in the Repository.

Session

From: 01-05-2021 To: 02-09-2021  Export All CRUD Report Demo Sessions

1 2 3 4 5 6 7 ... 30 Page Size: 10  1 Search

ID	Date	Time	Public IP	IP Address	System	OS Version	Java Version
1	01-06-2021	09:20	127.0.0.1	10.0.0.104	Windows 10	10.0	1.8.0_221
2	02-06-2021	09:50	127.0.0.1	10.0.0.104	Windows 10	10.0	1.8.0_221
3	03-06-2021	10:28	127.0.0.1	10.0.0.104	Windows 10	10.0	1.8.0_221
4	04-06-2021	11:03	127.0.0.1	10.0.0.104	Windows 10	10.0	1.8.0_221
5	07-06-2021	11:44	127.0.0.1	10.0.0.104	Windows 10	10.0	1.8.0_221
6	08-06-2021	12:12	127.0.0.1	10.0.0.104	Windows 10	10.0	1.8.0_221
7	09-06-2021	12:56	127.0.0.1	10.0.0.104	Windows 10	10.0	1.8.0_221

Activity

From: 26-08-2021 To: 02-09-2021 Action: Get Repository Capacity  Export

1 2 3 4 5 6 Page Size: 10  2 Select

ID	Date	Time	Activity	Entity ID	Entity	Icon	Name
1	01-06-2021	09:22	CREATE	1	Symbol		symbology-abstract
2	01-06-2021	09:22	CREATE	1	ShapedSymbol		symbology-abstract
3	01-06-2021	09:22	UPDATE	1	ShapedSymbol		symbology-abstract
4	01-06-2021	09:23	CREATE	2	ShapedSymbol		Linguistic Model
5	01-06-2021	09:23	UPDATE	2	ShapedSymbol		Linguistic Model
6	01-06-2021	09:23	CREATE	3	ShapedSymbol		Domain
7	01-06-2021	09:23	UPDATE	3	ShapedSymbol		Domain

3 Open In Designer

In-Tray

Facilitates a mechanism that contains the  work load of the  user and functionality to be able to view their assigned  tasks and so either attend to  notes (task items) or add additional  notes of their own. Its intended to be included as part of the users daily  work flow to monitor their work

The In-Tray is a storage mechanism containing the work load of all Tasks that have been assigned to a specific Repository User
In conjunction to attending to the Tasks and their Notes, also consult the relevant Design Strategy available in all the Designers

 Refresh 

Tasks

Created	Updated	Expected	Priority	Entity	Status	Name	Description
26 May 2021	26 May 2021	21 Jun 2021	Major	Graphic Model	In Progress	Symbology	Refer to the narrative Symbology and create all the graphic entities a

1 Select  Open Symbology designer with Narrative

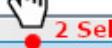
General Instructions 

Refer to the narrative Symbology and create all the graphic entities accordingly
Make sure the Graphic Model reflects the Linguistic Model
Make sure that the Symbology reflect the Symbolisms
Make sure that the Contexts reflect the Narratives

 In Progress 
5 Status In Prog 
Save Task Status 
6 Save

Notes

Created	Updated	Entity	Note	Attended
26 May 2021	26 May 2021	Catalog	Create Symbols using templates	<input type="checkbox"/>
26 May 2021	26 May 2021	Catalog	Review Symbols templates	<input type="checkbox"/>
26 May 2021	26 May 2021	Catalog	Commit Symbols to repository	<input type="checkbox"/>
26 May 2021	26 May 2021	Symbology	Create Symbology using templates	<input type="checkbox"/>
26 May 2021	26 May 2021	Symbology	Review Symbology templates	<input type="checkbox"/>

2 Select 

Specific Instructions 

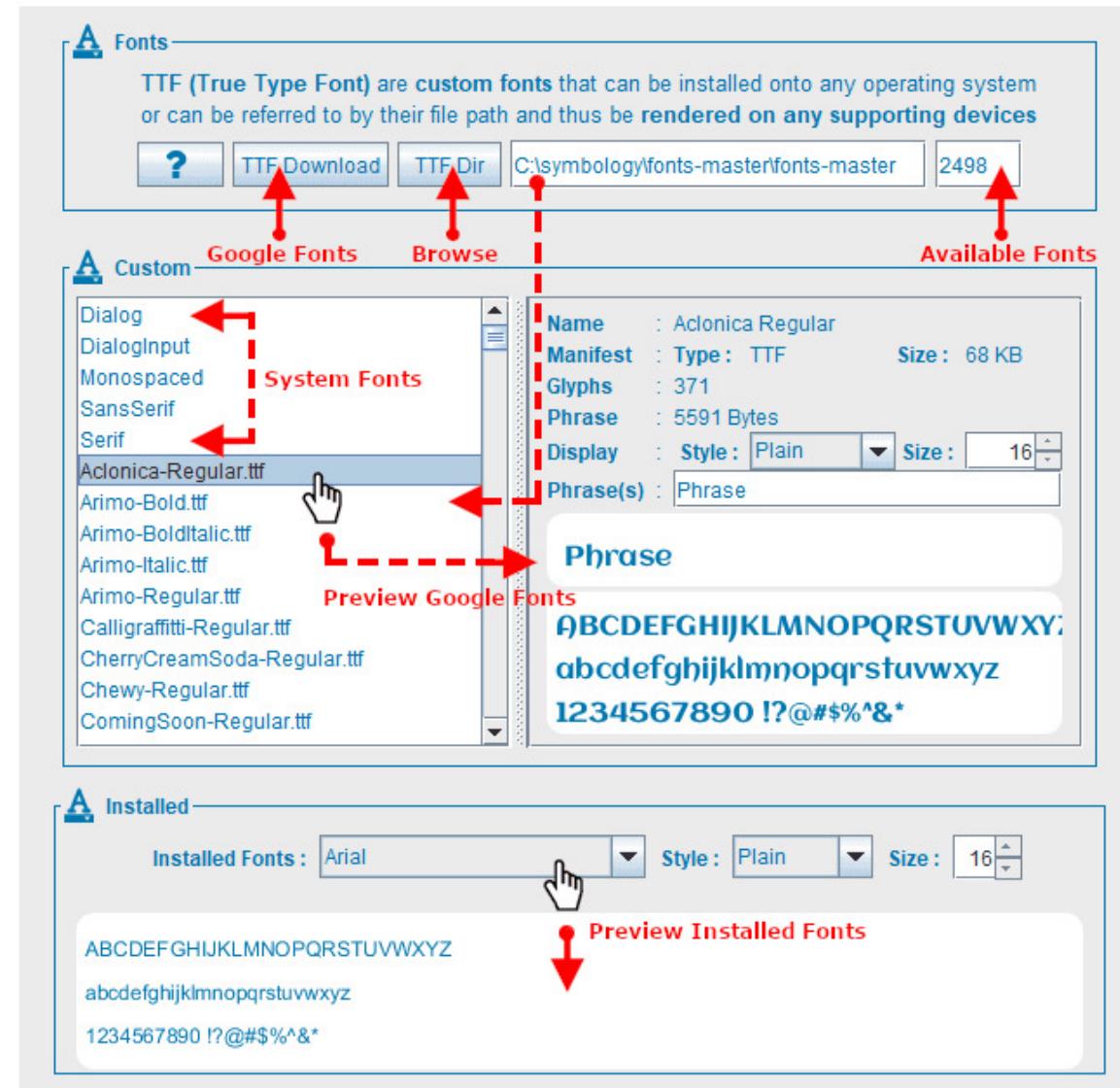
3 Check 

 Catalog 
Save  Add 
4 Save 

A Font Manager : Help

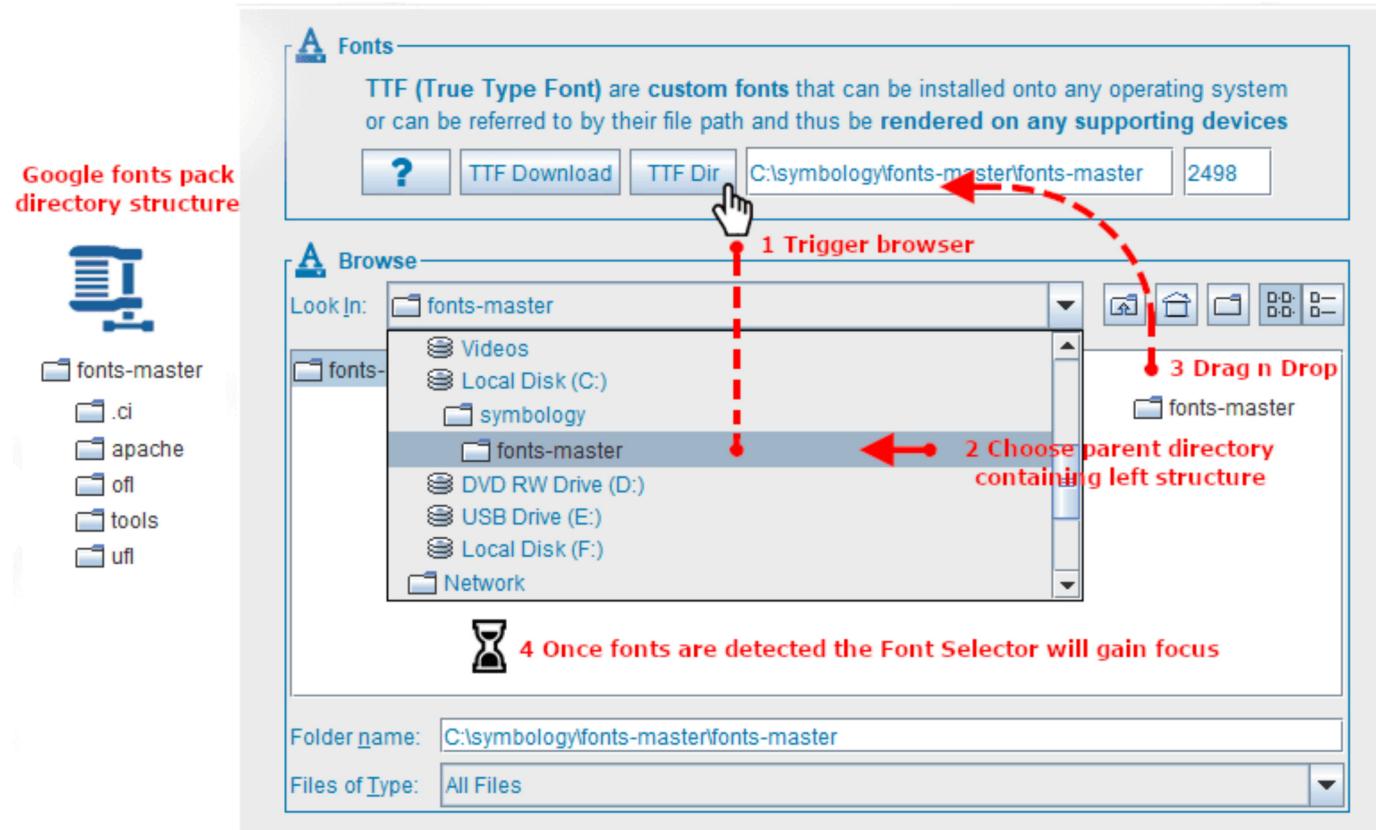
Font Manager

Facilitates a mechanism whereby the **G** google fonts pack can be imported into the **S** studio app for its **A** fonts to be used in the **H** compositions as well as providing a convenient preview utility to look at System, TTF and Installed fonts in different **T****C** styles asserted onto the specified phrase



Pack Selection

Once the google fonts pack has been downloaded successfully from the [GitHub](#) repository at <https://github.com/google/fonts> it must still be imported into the [Studio](#) app by browsing for it locally on your machine and locate the directory and drag it onto the TTF Dir text field for it to register all the [A](#) fonts



Repository The  repository has internal catalogs that it stores all its  SVG content in, of which the main graphic entities taking up the most storage space are the primary  symbology and all of their secondary  visual elements namely that of all the  text,  shaped symbols and  imported symbols To effectively conserve the storage space in the  repositories it is critical that  best practices be followed consistently during a users  work flow.

-  **Redundancy** : avoid littering a repository with redundant entries due to redrafts by defining  design strategies and so implementing them
-  **Duplication** : reuse all of the graphic entities to the maximum extent through employing the use of  cloning,  symbols and  contexts

All the  Designer Views can open their relevant  Web Console to browse the  SVG content which they are solely responsible for maintaining.



[Refresh PC](#)