Operational

Requirements

Specifications

Of the Eclipse inkstone Plugin

Document version : 2.0

Document date : 25/09/2012

Author : Ofer Calvo

Version remarks : Requirements update at the “INK Kiosk View” requirements.

Content List:

1 Abstract 1

1.1 Abbreviations list 1

1.2 Definitions 1

1.3 Applicable Documents 1

2 Operational principles 1

3 Graphic interface design 1

3.1 Plugin views 1

3.1.1 INK Properties View 1

3.1.2 INK Kiosk View 1

3.1.3 INK Diagram View 1

3.2 Plugin dialogs 1

3.2.1 INK Diagram Deletion Warning 1

3.2.2 The inkstone general error message 1

3.2.3 The inkstone Preference page 1

3.3 Plugin Action Commands 1

3.3.1 INK Toolbar group 1

3.4 Plugin perspective 1

4 Appendixes 1

4.1 Appendix A – INK Diagram 1

4.2 Appendix B – Operational scenarios 1

4.2.1 Scenario: Make project-based INK diagram 1

4.2.2 Scenario: Design INK-based with visual notations. 1

Figure List:

[Figure 1 – INK Properties View 1](#_Toc325923565)

[Figure 2 – INK Kiosk View 1](#_Toc325923566)

[Figure 3 – INK Diagram View 1](#_Toc325923567)

[Figure 4 - INK Diagram with compilation error 1](#_Toc325923568)

[Figure 5 – Confirm dialog of INK diagram deletion 1](#_Toc325923569)

[Figure 6 - Plugin general error message 1](#_Toc325923570)

[Figure 7 - INK Diagram Preference page 1](#_Toc325923571)

[Figure 8 - INK Kiosk Preference page 1](#_Toc325923572)

[Figure 9 - INK Toolbar group 1](#_Toc325923573)

[Figure 10 - Diagram filter options tree 1](#_Toc325923574)

[Figure 11 – INK Perspective 1](#_Toc325923575)

Table List:

[Table 1 - INK properties interactions 1](#_Toc325923576)

[Table 2 - INK kiosk interactions 1](#_Toc325923577)

[Table 3 - INK diagram interactions 1](#_Toc325923578)

[Table 4 - Actions commands 1](#_Toc325923579)

# Abstract

The inkstone project is about extending INK workbench project to enable graphical editing of INK metadata, using diagram drawing of notations (INK elements and relations between them).

The project goal is to simplify DSL design with INK, to domain specialists that are not programmers, and to enable modeling visual language between people that uses INK.

This paper is a specification document that lists the operational requirements of inkstone.

The Operational requirements will define:

* Operational principles (see chapter ‎2).
* GUI design (see chapter ‎3).
* INK notations (see appendix A, chapter ‎4.1).
* Operational scenarios (see appendix B, chapter ‎4.2).

And, be a reference document to the functional requirements and the design of the plugin.

## Abbreviations list

|  |  |
| --- | --- |
| **DSL** | Domain Specific Language |
| **GUI** | Graphical User Interface |
| **SDL** | Simple Declarative Language |
| **SVG** | Scalable Vector Graphics |

## Definitions

|  |  |
| --- | --- |
| **INK Element** | A graphical shape used to note: INK meta-class, INK class or INK Object.  More in appendix A… |
| **INK Relations** | A graphical relation line used to note: Extension, Instancing or Referencing.  More in appendix A… |
| **INK File** | A text file with “\*.ink” suffix. Written in SDL format. |
| **Swing** | Part of Oracle's Java Foundation Classes (JFC), an API for providing a graphical user interface (GUI) for Java programs. |

## Applicable Documents

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Document Name** | **Version** | **Date** |
|  | inkstone FRS | 2.0 | 25/09/2012 |

# Operational principles

The graphical interface of inkstone should be based on the Eclipse development environment visual GUI components.

It shall be simple and fast to operate. Meaning, no more than 3 hits/clicks actions per each functionality usage.

# Graphic interface design

## Plugin views

### INK Properties View

Normal icon:  Error icon: 

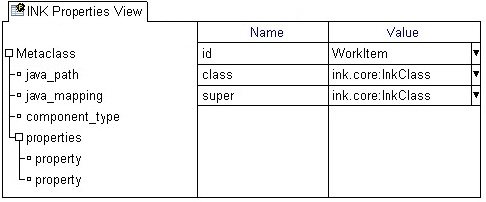


Figure 1 – INK Properties View

This view shall be used to edit selected INK elements properties.

The view is going to be built from a “tree-properties” component, to allow organize a list of properties values in a hierarchical way to each node in the tree.

The “tree-properties” component underlying data model shall be kept in an INK file (SDL syntax).

On INK compilation error, replace the view icon to the error icon.

#### Popup menu options

This view shall support the next popup menu options:  
(Keyboard alternatives: mnemonics and accelerators, are customary marked)

* **Collapse All (Alt "–")** – Close all tree nodes.
* **Expand All (Alt "+")** – Open all tree nodes.
* **Add Item (Alt A)** – Add new node/cell row to current selected position.
* **Add Sub Node (Alt S)** – Add new sub node to current selected node.
* **Delete Element** **(Del)** – Deletes current selected node/cell row from the view.
* **Go to Element (F3)** – Opens the INK file, which contains the INK element, on the elements declaration.

#### Mouse buttons options

This view shall support the next mouse commands:

* **Right mouse button** – Open the view popup menu.

#### Views Interactions

|  |  |
| --- | --- |
| When … | Do … |
| INK element is selected in the "INK diagram view" | Show its properties from the related ink file. |
| Current properties changed by the user. | Save the element to the related ink file. |

Table 1 - INK properties interactions

#### View Tool Bar

N/A

#### Display of Information

<TBD>

### INK Kiosk View

Normal icon: 

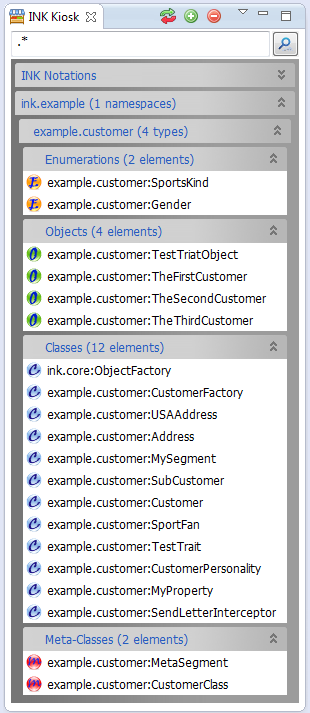
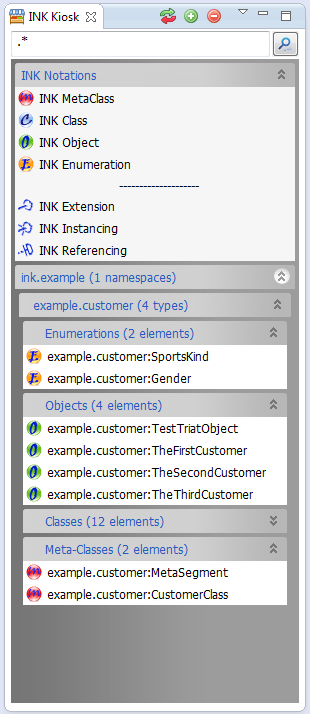
 

Figure 2 – INK Kiosk View examples

The kiosk view usage will be to store reusable notations of elements, to be dragged into the INK diagram view.

The view shall be built from a collapsible panel sections. The first mandatory section is dedicated to store basic INK notation for drawing of new diagram elements and relations. The other optional sections are dedicated to display a static display of currently selected INK elements. This while using hierarchy display of INK projects 🡪 name spaces (DSL’s) 🡪 INK type. INK types are: Meta-Class, Class, Object and Enumeration.

#### Popup menu options

This view shall support the next popup menu options:  
(Keyboard alternatives: mnemonics and accelerators, are customary marked)

Expand-bars title menu:

* **Collapse All** **(Ctrl C)** – Open all the view sections.
* **Expand All** **(Ctrl E)** – Close all the view sections.

Expand-bars element list menu:

* **Open element** **(Ctrl O)** – Open the INK core text editor, at the definition of the selected element.

#### Mouse buttons options

This view shall support the next mouse commands:

* **Right mouse button** – Open the expand-bars title, or the expand-bars element list, popup menu.

#### Views Interactions

|  |  |
| --- | --- |
| When … | Do … |
| Dragging INK notation from the view into the INK diagram view. | Place new diagram notation in the diagram. |
| INK element is selected in the "INK Kiosk view" | Show the selected element properties at the "INK diagram view" and (if exist) mark the element in the current display of the “INK Diagram View”. |

Table 2 - INK kiosk interactions

#### View Tool Bar

|  |  |
| --- | --- |
| Icon | Role |
| C:\Users\OC\workspace\ink.inkstone\src\inkstone\utils\gallery\add_icon.png | Opens for selection of the INK projects & libraries tree, from the current active INK model. Selected libraries elements shall fill the Kiosk bars, after closing the dialog.  (See paragraph ‎3.2.3 DSL’s Selection Dialog) |
| C:\Users\OC\workspace\ink.inkstone\src\inkstone\utils\gallery\delete_icon.png | Clear Kiosk view. Leaving only the basic INK notation bar. |
| C:\Users\OC\workspace\ink.inkstone\src\inkstone\utils\gallery\refresh_icon.png | Refresh current Kiosk view data. Update Kiosk static display from the current saved INK model. |

#### Display of Information

The Kiosk view shall use two display methods to help dealing with “Information Overload”.

Firstly, with the view display of selected DSL’s elements (from INK model) in tree-like order of cascading bars.  
Each bar shall be titled with the project/library/INK kind name, plus the number of sub instance in brackets.

Secondly, with the help of a filter tool, at the top of the view, that should hide display elements according to a [regular expression](http://docs.oracle.com/javase/6/docs/api/java/util/regex/Pattern.html#sum) string match of the element names.

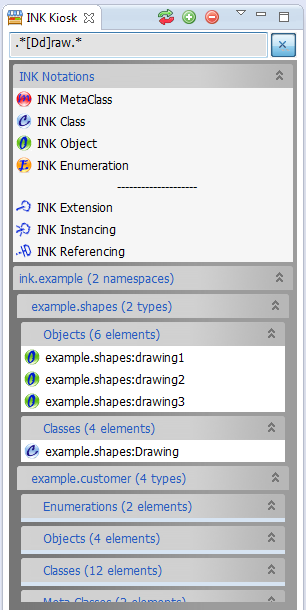


Figure – Kiosk view regex filtering example

### INK Diagram View

Normal icon:  Error icon: 

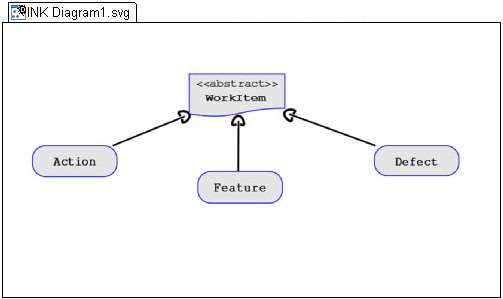


Figure 4 – INK Diagram View

The INK diagram should be built from a Swing component that will be used to display SVG documents graphics.

Each INK diagram will be connected to a single SVG file, which will store the vector graphics data of the diagram.

It will let the user manipulate the displayed SVG document and its elements, with zooming and panning commands.

Selected INK element in the diagram should be visually marked with yellow highlight border and its INK properties should be shown in the INK properties view.

INK compilation errors in the diagram should be marked with a red highlight border around elements with error. When one (or more) of the diagram elements has INK error, the view icon should change to an error icon. Also, the error tooltip should popup when clicking on such an element.

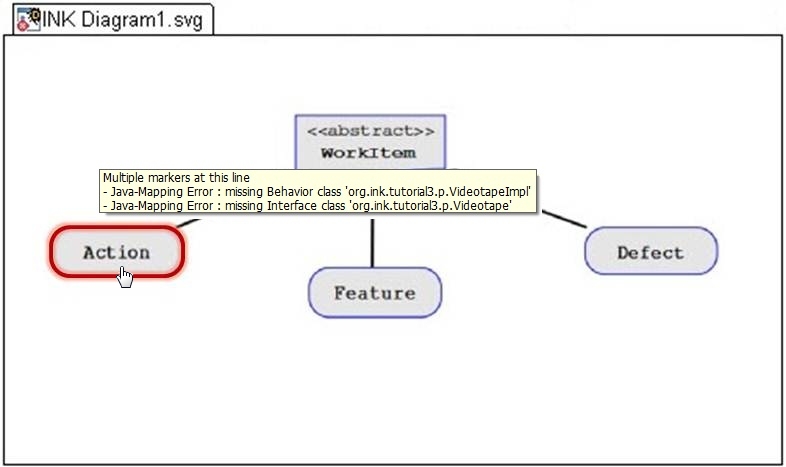


Figure 5 - INK Diagram with compilation error

#### Popup menu options

This view shall support the next popup menu options:  
(Keyboard alternatives: mnemonics and accelerators, are customary marked)

* **Add INK Element** **(Alt E)** – Add a new empty elements from the this options:
  + INK Meta-Class
  + INK Class
  + INK Object
* **Add INK Relation(Alt R)** – Add a new empty elements from the this options:
  + INK Extension
  + INK Instancing
  + INK Referencing
* **Delete Notation (Del)** – Deletes current selected notation from the diagram.
* **Zoom** **(Alt Z)** – Zoom diagram options:
  + 25% – Zoom out to one quarter of original size.
  + 50% – Zoom out to half of original size.
  + 75% – Zoom out to three quarters of original size.
  + 100% – Set diagram to original size.
  + 150% – Zoom in to one and half of original size.
  + 200% – Zoom in to double original size.
* **Copy Image** **(Alt C)**  – Copy image to the system clipboard (to paste/export image).
* **Default INK file (Alt H)** – Open project file dialog tree, to select a new default INK file to the diagram.
* **Go to Element (F3)** – Opens the INK file, which contains the currently selected INK element.

#### Mouse buttons options

This view shall support the next mouse commands:

* **Right mouse button** – Open the view popup menu.
* **Left mouse button** – Holds the diagram (while down) for panning with mouse movement.
* **Mouse wheel** – Zoom in/out in the range of 10% to 500% of original size.

#### Views Interactions

|  |  |
| --- | --- |
| When … | Do … |
| Dragging INK element around in the diagram. | Change element location and relation lines path to/from it when the all object is selected.  Resizing element size when one corner is selected.  Show element properties in the INK Properties view. |
| Drop INK **element** from the kiosk view. | Add new element to the diagram and show its properties in the INK Properties view. |
| Drop INK **Relation**, from the kiosk view, on INK element. | Change mouse icon to a cross and wait to a second left mouse click on the targeted element in the diagram.  Place the relation line if element selected, or cancel operation if "Esc" key is pressed. |
| Dropping **INK file** from the Package Explorer view, into the INK Diagram view. | Add all none already displayed elements to the diagram. |

Table 3 - INK diagram interactions

#### View Tool Bar

<TBD>

#### Display of Information

<TBD>

## Plugin dialogs

### INK Diagram Deletion Warning

Usage: confirm an INK diagram deletion. See FRS scenario: “Remove INK Diagram”.

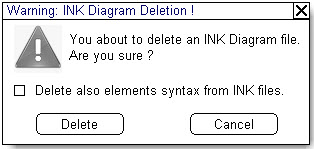


Figure 6 – Confirm dialog of INK diagram deletion

### The inkstone general error message

Usage: Display general simple error messages. The label will be the calling object (general the view) and the message will be an input from that calling object.



Figure 7 - Plugin general error message

### DSL’s Selection Dialog

Usage: Opens for selection of the INK projects & libraries tree, from the current active INK model.  
Selected libraries elements shall fill the Kiosk bars, after closing the dialog (Done button).

INK libraries are also called “DSL’s” in INK.

The tree should have check-boxes for each project/library. Select all, and Deselect all buttons should be placed in order to make mass selection more easy.

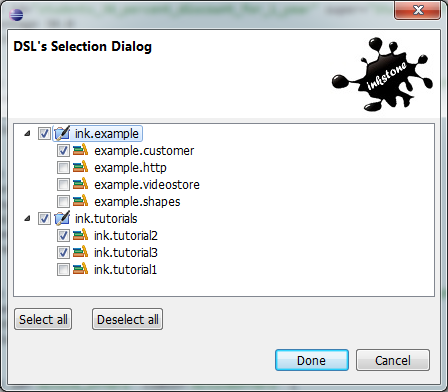


Figure – DSL’s Selection Dialog example

### The inkstone Preference page

A preference is data that is persisted between work sessions. The plugin preference will be integrated with Eclipse preference dialog and will be on a per project basis.

#### INK Diagram Preference

Set the project folder for INK Diagrams and the default INK file for every INK Diagram.

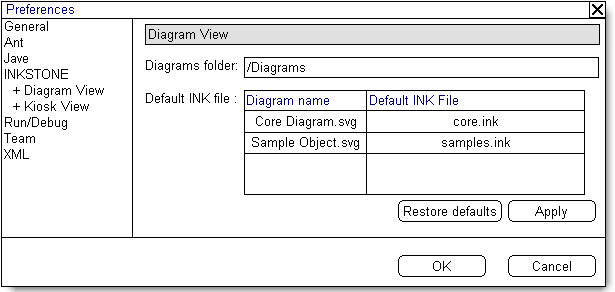


Figure 9 - INK Diagram Preference page

The Apply button is to submit changes.

The Restore defaults button set the diagram folder back to “Diagrams”.

#### INK Kiosk Preference

Set the linked INK files of the current project.

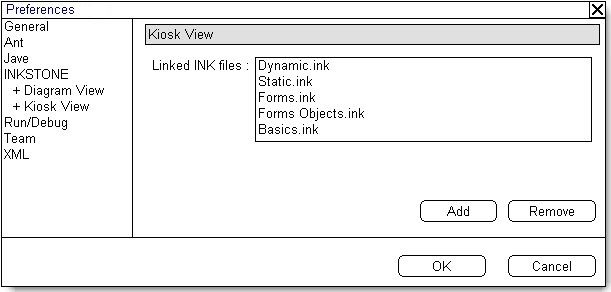


Figure 10 - INK Kiosk Preference page

The Add button opens tree dialog of projects INK files, and add them to the list.

The Remove button deletes current selected rows from the list.

## Plugin Action Commands

The plugin action commands will use the Eclipse environment GUI menus and toolbar icons (an inkstone toolbar group adding to the Eclipse main toolbar).

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Workbench Menu/s | Active when… | INK Toolbar |
| Add INK Diagram view | Ink menu.  File/New menu. | Always |  |
| Show INK Kiosk view | Ink menu.  File/New menu. | Kiosk view closed |  |
| Show INK Properties view | Ink menu.  File/New menu. | Properties view closed |  |
| Diagram Filter tool | Ink menu. | Diagram view in focus | C:\Users\OC\Documents\אופ תואר שני\פרויקט מתקדם במדעי המחשב - 22997\הצעה\ORS\graphics\Edit Filter icon.gif |
| Show inkstone perspective | Window/Open Perspective menu. | inkstone perspective not selected |  |

Table 4 - Actions commands

### INK Toolbar group

The toolbar group added to the workbench toolbar.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  | ▼ |

Figure 11 - INK Toolbar group

#### Diagram Filter Tool

A drop down toolbar menu button with a tree of check-box option to mark the visible /none visible INK notations display. None active notations will be hidden from the current selected INK diagram.

Each INK diagram shall have its own saved filter set of these options.

Every combination is allowed, but relations with hidden source or target elements will be hidden as well, even if marked as visible.

-Elements

🗹 Meta Class

🗹 Class

🗹 Object

- Relations

🗹 Extension

🗹 Instancing

🗹 Referencing

Figure 12 - Diagram filter options tree

## Plugin perspective

The plugin perspective shall set a graphical layout to INK development with a set of inkstone views layout.

It will be called: “INK”, and could be found in the Eclipse perspectives list.

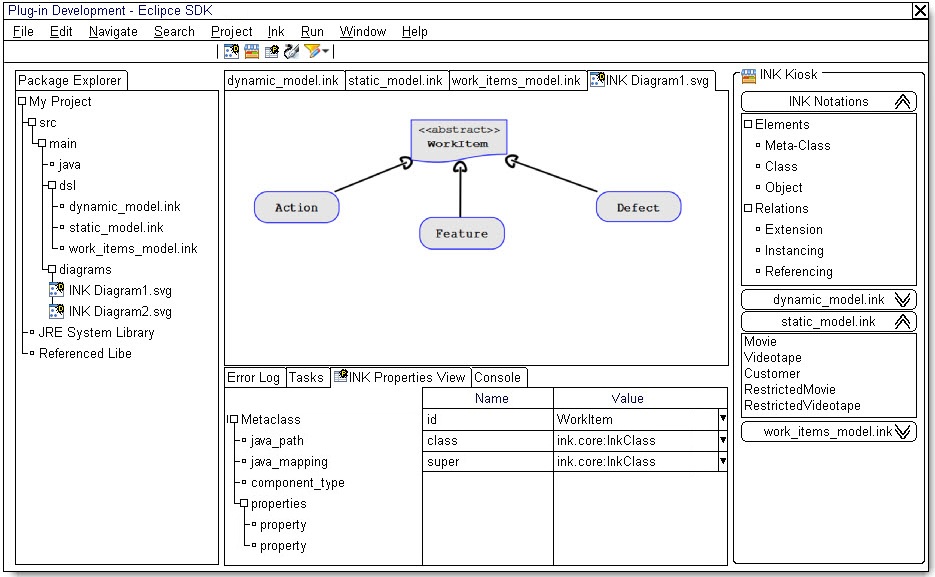


Figure 13 – INK Perspective

# Appendixes

## Appendix A – INK Diagram

The INK diagram is a graphic modeling language designed to suite the INK syntax with similarity to UML.

The next (imbedded) presentation describes the different types of INK notations with examples.



## Appendix B – Operational scenarios

These next informal scenarios define operational usage of the inkstone plugin. They should clarify the workflow of the user with the plugin.

### Scenario: Make project-based INK diagram

1. Open existing java project in Eclipse.
2. Create new INK diagram and select one of existing INK files as its defualt.
3. Drop all needed INK files into the diagram.
4. Wait to the plugin to open INK Kiosk, add all files to the kiosk and add all INK elements from files to the INK diagram.
5. Fix elements locations in the diagram. Remove, or filter out, unnecessary INK elements.
6. Set Diagram zoom and panning.
7. Copy INK diagram image to your office suite application for presentation or documentation needs.

### Scenario: Design INK-based with visual notations.

1. Open existing java project in Eclipse.
2. Open the INK perspective layout.
3. Add new INK Diagram. Select yours new INK file as its default.
4. Drag the relevant INK files, needed, from the Package Explorer view to the INK Kiosk view.
5. Drag (from Kiosk) new elements you design to the newly created INK diagram.
6. Drag existing linked elements (from the Kiosk linked sections) to relate your elements to.
7. Make necessary INK relations to new elements by dragging and assigning new INK relations from the Kiosk to your diagram.
8. Select each new INK elements and their properties types (or values in case of INK objects).
9. Check for errors in the INK properties view, and fix them.
10. The new INK file (diagram default file) is now ready and already saved. You can pass it to peers review and implementation.