|  |
| --- |
| Hogeschool Utrecht |
| Architecture Graphics Manual |
| Architecture Graphics Team & Service |

|  |
| --- |
| **HUSACCT Team 2 – Graphics**  Stan Verhoeckx Dennis Bullee Tim Bazuin Ruwan Kellaert Mark Wallenburg  **HUSACCT Team 5 – Architecture Graphics**  Tom de Bruijn Johan van der Slikke Guido van Tricht Patrick van der Willik  **Clients**  Michiel Borkent (michiel.borkent@hu.nl) Christian Köppe (christian.koppe@hu.nl) Leo Pruijt (leo.pruijt@hu.nl)  **Team mentor**  Christian Köppe (christian.koppe@hu.nl)  **Date**  20-06-2013 |

Contents

[Introduction 2](#_Toc359836361)

[Terminologie 2](#_Toc359836362)

[Interactieve diagrammen 3](#_Toc359836363)

[Architecture Graphics User Interface 4](#_Toc359836364)

[1. Tekenvlak waar de HUSACCT het diagram tekent 4](#_Toc359836365)

[2. Actie: Inzoomen op selectie 4](#_Toc359836366)

[3. Actie: Uitzoomen van huidige locatie 5](#_Toc359836367)

[4. Actie: Tekening vernieuwen 5](#_Toc359836368)

[5. Optie: Dependencies tonen / verbergen 5](#_Toc359836369)

[6. Optie: Violations tonen / verbergen 5](#_Toc359836370)

[7. Actie: Diagram exporteren als plaatje 6](#_Toc359836371)

[8. Muis tool selecteren 6](#_Toc359836372)

[9. Opties menu openen 7](#_Toc359836373)

[11. Smart lines 7](#_Toc359836374)

[12. Layout strategy 7](#_Toc359836375)

[10. Optie: Schaal van het diagram aanpassen 8](#_Toc359836376)

[11. Locatie balk 8](#_Toc359836377)

# Introduction

This document contains an overview of the different screens you will encounter while using HUSACCT (HU Software Architecture Compliance Checking Tool), along with an explanation of what the different parts do.

Before anything else, a glossary of used terms is provided. Subsequently, a description of the process to open diagrams within the application is given. Finally, the functionality in the main screen is described.

# Terminology

The following terms are used within the HUSACCT application:

* *Dependency*
  + A relationship between layers, packages or classes.
* *Violation*
  + A violation of the defined architectural rules.
* *Zoom in*
  + Zooming in this context likely means going down into the architectural hierarchy. By zooming on a package, the package is unfolded and all the underlying packages and classes are shown.
* *Zoom out*
  + Step one level “up” in the architectural hierarchy.
* *Multizoom*
  + Zooming in on multiple packages/classes in order to view their connectedness.

# Interactive Diagrams

The purpose of interactive diagrams is visualizing software architecture. The physical and logical diagrams are made based on the analysed code and the defined software architecture. These diagrams are a part of HUSACCT. Where other screens are primarily focused on entering data, these diagrams are primarily meant to display it.

Two types of diagrams are offered within HUSACCT. One based on the analysed software, which can be opened immediately after analysing an application and one based on the architecture as defined by the user. The analysed diagram shows the structure of the application on a physical level. This can be compared to a explorer type system where the dependencies between objects is also shown. This screen can be opened through the main menu of HUSACCT under “Analyse” -> “Analysed architecture diagram” as shown in figure 1.

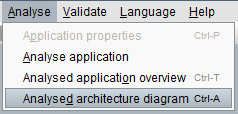


Figure 1 Opening the analysed architecture diagram

The defined diagram is based on user input. Before anything can be displayed, an architecture needs to be defined. Based on this information and the linked physical objects, a logical architectural diagram is shown. Using this diagram it is possible to compare the documented architecture to the actual architecture. This diagram supports the creation of logical layer models and component models. This diagram can be accessed as shown in figure 2.

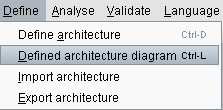


Figure 2 Opening the defined architecture diagram

The following examples use the Analysed architecture diagram as an example, but the options and functions shown can be applied to both diagrams as their interfaces are identical.

# User Interface

Figure 3 shows the main screen of the graphics component. The numbers in the image correspond to the following chapter numbers



Figure 3 Graphics service main screen

## Drawing surface

This surface is meant to display the diagram. The figures and lines that are involved with visualizing a software architecture are shown here. This area is interactive – modules can be dragged around to facilitate reader comprehension. Selecting figures and lines allows the end-user to access more detailed data about them. Double clicking on a figure zooms in on it.

## Zoom

Left clicking this button zooms on the selected figures. Right clicking on this button displays a menu that offers the different zoom options: “Zoom with context” and “Zoom in”.  
“Zoom with context” keeps the other diagrams displayed at the same level, while zooming in on the selected diagrams.  
“Zoom in” only zooms in on the selected diagrams.

Figure 4 shows an example of two packages that have been zoomed in on at the same time.

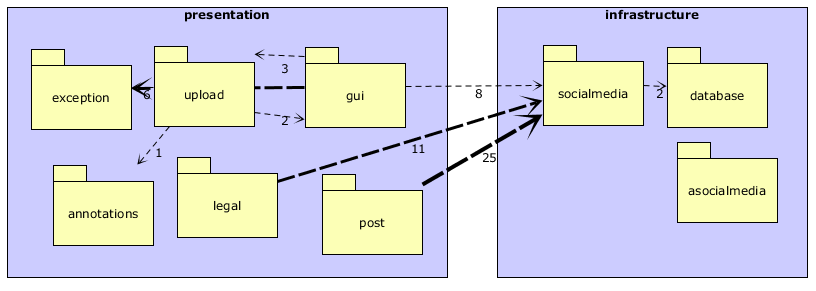


Figure 4 Multi zoom diagram.

## Zoom out

This button steps up one level of abstraction in the diagram.

## Refresh diagram

Clicking this button results in the diagram being refreshed from internal data. This is primarily useful when/if things have been changed using other screens within HUSACCT.

## Toggle dependencies

This button toggles the dependency lines between figures in the diagram. Figure 5 shows an example of a diagram where the dependency lines have been turned off.

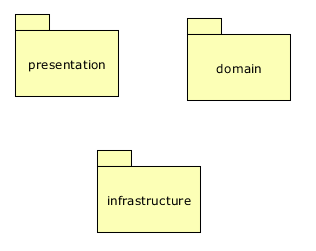


Figure 5 Dependency lines off

## Toggle violations

By default the violations are not shown. These arrows, when drawn between figures, indicate violations of the defined architectural rules. These lines are not black in order to facilitate distinguishing between them and dependency lines. This colour can be changed using the validation configuration screen.

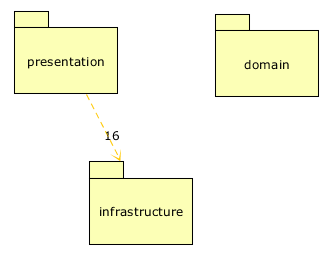


Figure 6 Violations enabled, Dependencies disabled

## Actie: Diagram exporteren als plaatje

Het diagram dat getoond wordt in de applicatie is vluchtig. De applicatie moet opgestart zijn om deze te kunnen inzien. Voor gebruikers is het handig als deze informatie ook buiten de applicatie beschikbaar is. Om deze reden kan het huidig getekend diagram ge-exporteert worden naar een PNG plaatje.

## Muis tool selecteren

Standaard is de selection tool geselecteerd. Dit is te zien aan de magenta rand eromheen. Deze tool stelt de gebruiker in staat elementen in het diagram te selecteren om te verplaatsen of eventueel op in te zoomen. Deze tool heeft als icoon het standaard muisicoon. Naast de selection tool staat de pan tool, deze kan gebruikt worden om door te klikken op en de muis over het canvas te bewegen te navigeren door het diagram.

## Opties menu openen

Het opties menu is een klein scherm met bijna alle functionaliteit van de interactiviteit van het diagram. Het bevat bijna alle opties van de menu balk. De nummers 1-10 komen overeen met de nummering van de onderdelen in het hoofdscherm. De nummers 11 en 12 zijn functionaliteiten die alleen beschikbaar zijn via dit opties menu.

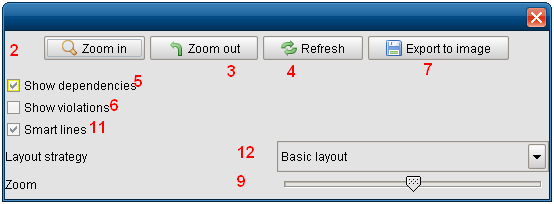


Figure 7 Opties scherm

### 11. Smart lines

In complexe diagrammen kunnen grote aantal lijnen voor een onoverzichtelijk diagram zorgen. Door de optie Smart lines aan te zetten worden overlappende lijnen naast elkaar geplaatst zodat ze elkaar niet in de weg zitten. Ook wordt de lijn dikte aangepast naarmate er meer relaties (dependencies en violations) er tussen twee figuren lopen.

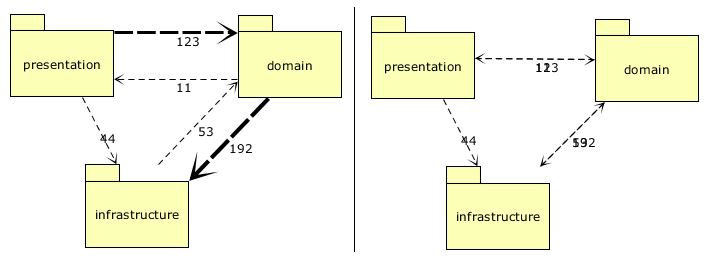


Figure 8 Vergelijking smart lines. Links smart lines aan. Rechts smart lines uit

### 12. Layout strategy

De layout strategy bied een automatische oplossing voor het sorteren van figuren. Dit zorgt voor een gestructureerde layout zonder dat de gebruiker eerst tijd kwijt is met het zelf organiseren hiervan. De automatisch oplossing is altijd nog aan te passen door de gebruiker. Er zijn een aantal layout strategies beschikbaar.

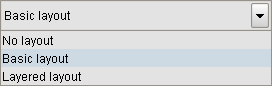


Figure 9 Alle huidige geïmplementeerde layout strategies

* No layout strategy
  + Er wordt geen layout toegepast op de figuren. Alle figuren worden in de linkerbovenhoek geplaatst.
* Basic layout strategy
  + Standaard optie. Deze layout strategy zet de figuren naast elkaar tot deze niet meer op het scherm zichtbaar zijn. Daarna positioneert deze de figuren op een nieuwe rij.
* Layered layout strategy
  + Experimentele layout strategy. Deze strategie houd rekening mee met het aantal dependencies tussen figuren. Op basis hiervan probeert het een zo overzichtelijk mogelijk diagram te tekenen. Deze layout strategie is nog niet af.

## Optie: Schaal van het diagram aanpassen

Om overzicht te scheppen kan het diagram geschaald worden. Door het punt in de slider te verslepen kunnen de figuren binnen het diagram vergroot of verkleind worden.

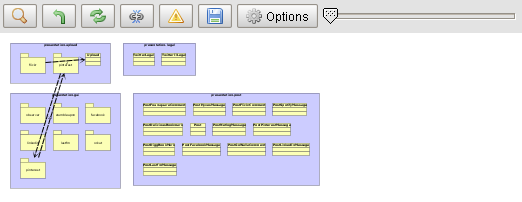


Figure 10 Geschaald diagram

## Locatie balk

Navigatie balk om de huidige locatie binnen architectuur te tonen en navigatie naar specifieke niveaus mogelijk maken. Elk niveau wordt aangeduid met een apart knop. Door op deze knop te drukken kan direct naar dat niveau worden uitgezoomt.



Figure 11 Locatie balk met inzooming van één niveau, twee niveaus diep



Figure 12 Locatie balk met inzooming van meerder niveaus