

# Glamour packages - User study

## Presentation

The goal of this study is to assess packages forming an application. You will assess packages through their classes and class references, analyzing what they use and how they are used, both internally and externally to the application.

This study is organized in three sections which require more and more in-depth look at packages and their classes. Each section implies that you play a different role when assessing the application, first as newcomer and potential client who wants to use the application, second as an architect who needs to assess the organization, third as a developer who performs maintenance. There are **11 questions** in this study, each question relating to a task.

You will perform the study on Glamour packages. Glamour is an engine for scripting browsers for any kind of models. If you are unfamiliar with Glamour, do not hesitate to test it before the study to get a basic understanding of Glamour capabilities. Information on usage and samples are available on: <http://www.moosetechnology.org/tools/glamour>

**Tool used:** System browser or OB System browser

## Instructions

- Use only the tool indicated for the study. Do not use another browser/tool.
- Process questions in the given order (do not read questions in advance!)
- Please provide only accurate answers like the name of a class or a package, the association between two classes, the method which makes references.
- **Time yourself** for each question.
- Do not spend more than **20 minutes** on a question. If you reach this limit, write it down, stop the task, and proceed to the next question.
- You also have a time limit of **1h30** to answer the 11 questions so take care of your time.

## A. Application assessment

As a potential client, you are assessing the package dependencies of the application. You want an idea about the size of the application and the kind of dependencies needed, especially if it involves new dependencies to be loaded with the application.

1) How big is the application?

*Time taken: 4min*

(a) In number of packages : 11

(b) In number of classes (one of the following ranges):

[ ] <100; [ X ] 100-200; [ ] 200-300; [ ] > 300

2) What are the most important packages?

*Time taken: 20 min out of time and not finished*

(a) In terms of outgoing dependencies :

To painful to do using the browser. Moreover, I guess I've found a bug.

Select category 'Glamour-Core' and do 'browse package'.

In the new opened window, doing 'refactoring scope>dependencies' raise a DNU

Trait(Object)>>dNU: #superclass

...

(b) In terms of incoming dependencies

(c) Overall, considering both outgoing and incoming dependencies

3) Focus on package Glamour-Morphic:

*Time taken: 11min*

(a) list all package dependencies which are external to Glamour.

15

(b) in this list, please signal any external package which is not part of Pharo base (i.e., package must be loaded with Glamour).

Magritte-Morph-Model

Mondrian-Events

(c) are there other unexpected/unwanted package dependencies?

No

## **B. Application architecture assessment**

As an architect, you now want to check the organization of your packages. You want your packages to have a good rationale for existence in the application. You want some parts of the application to be modular.

4) Please characterize **each** Glamour package as either:

*Time taken: 8min*

- a provider package for external clients (package with which external clients interact)  
Glamour-Browsers, Glamour-Examples, Glamour-Tools, Glamour-Scripting

- an internal package (package which should not be accessed by external clients)

Glamour-Core, Glamour-Tests\*, Glamour-Morphic, Glamour-Presentations, Glamour-Helpers, Glamour-Announcements

5) Are some Glamour packages optional/modular (package can be unloaded without impacting application core)?

*Time taken: 2min*

Glamour-Scripting, Glamour-Tests, Glamour-Test-Morphic, Glamour-Examples, Glamour-Tools

6) What are the important classes (consider incoming, outgoing, inheritance dependencies) in Glamour-Core? If possible, explain their roles.

*Time taken: 9min*

GMLBrowser : main class representing a Browser

GLMRenderer : abstract renderer of visual elements (Presentations, Panes, Browsers, ...)

GMLPane : building block of a browser that can be connected through their GMLPort

GMLTransmissions : links GMLPorts

7) Are there direct cyclic dependencies from Glamour-Core to another package?

*Time taken: To difficult to know with the standard browser...*

### **C. Detailed assessment**

As a developer, you want a detailed comprehension and assessment of dependencies between classes and packages and optionally to refactor such dependencies, assessing impact of change.

*First give a precise answer then provide your explanation.*

8) What are the most cohesive packages of the application?

*Time taken: To difficult to know with the standard browser...*

9) There is a dependency to DeprecatedPreferences in Glamour-Morphic. Can you detect the faulty class? Explain the dependency: do you see an easy way to solve it?

*Time taken: 10min*

*For rendering purpose, the old Preference class is used to get the standardMenuFont.*

*To remove this dependency, Glamour should use the new Settings package that inverse the dependency (Settings system access to Glamour preferences).*

*Add glamourFont accessors on a GlamourConstants class.*

*Create a package Settings-Glamour and add a class side method (tagged with the systemsettings pragma) ...*

10) Can you explain the organization of Glamour-Morphic and its relationship with other packages?

*Time taken: 9min*

*GMLMorphicRenderer inherits from GMLRenderer (Glamour-core) and is responsible to create a Morphic view for each model elements (Pane, ...)*

*Some classes are part of the glamour-morphic event system and they are subclasses of Announcement package classes.*

11) Multiple packages of Glamour have dependencies to external library Mondrian. List such packages. Could you extract this dependency and make it optional (you can propose a solution)?

*Time taken: To tedious with standard browser.*

### **D. Personal remarks**

You can provide any additional remarks about the study itself, the tasks, the tool used.

The browser is not convenient to answer the questions.

The difference between category and package in the contextual menus is a mess.

For example, browse package on 'Glamour-Morphic' open only this category whereas on Mondrian-Events it opens on all Mondrian categories since it is the MC package. Is a browse category entry in the menu missing? But the menus are already too long... I think that redoing this study by implementing small scripts using SystemNavigation would really be more efficient for me.