

Diagramme de séquence



420-LAB-JQ Projet de fin d'études
Semaine # 2

Diagramme de séquence	1
1. Exemple de départ	2
Diagramme de classe Système d'exploitation - Plateforme	2
Diagramme d'objets	2
2. Diagramme de séquence	3
Diagramme de séquence Créer_Sys_Exp_Simple.....	3
Diagramme de séquence Créer_Plateforme_Simple	4
3. Améliorations	5
Diagramme de séquence système d'exploitation réutilisable	5
Diagramme de classe MVC.....	6
Diagramme de séquence Créer_Sys_Exp final.....	7
Diagramme de séquence Créer_Plateforme final	8

1.Exemple de départ

Diagramme de classe Système d'exploitation - Plateforme

Nous considérons le diagramme de classe simplifié suivant :

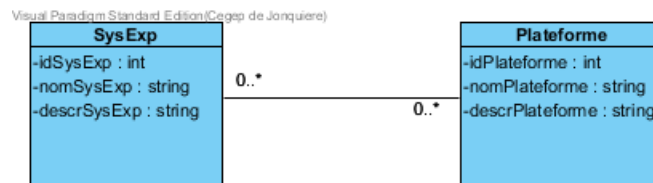


Figure 1:DCA

Diagramme d'objets

Nous obtenons le diagramme d'objets suivant qui permet d'obtenir une vue du système :

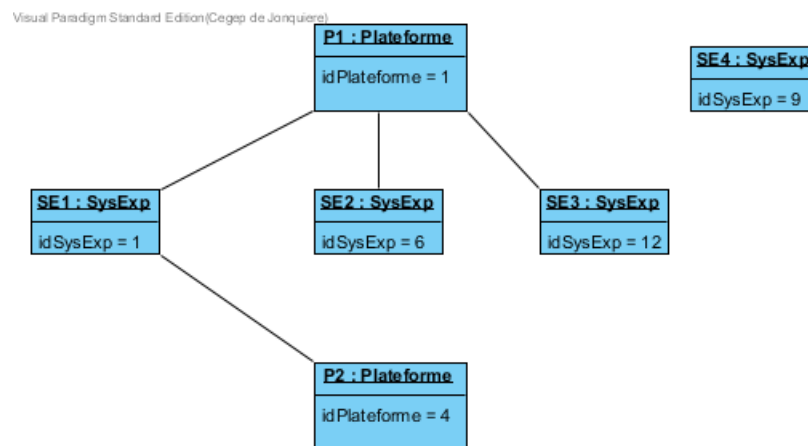


Figure 2: DCO

2. Diagramme de séquence

Pour tous ce qui concerne les diagrammes de séquences :

- http://www.visual-paradigm.com/support/documents/vpuserguide/94/2577/7025_creatingsequ.html
- <https://msdn.microsoft.com/fr-fr/library/dd409377.aspx?f=255&MSPPError=-2147217396>

Diagramme de séquence Créer_Sys_Exp_Simple

Le diagramme de séquence associé à la création d'un système d'exploitation est illustré à la Figure 3.

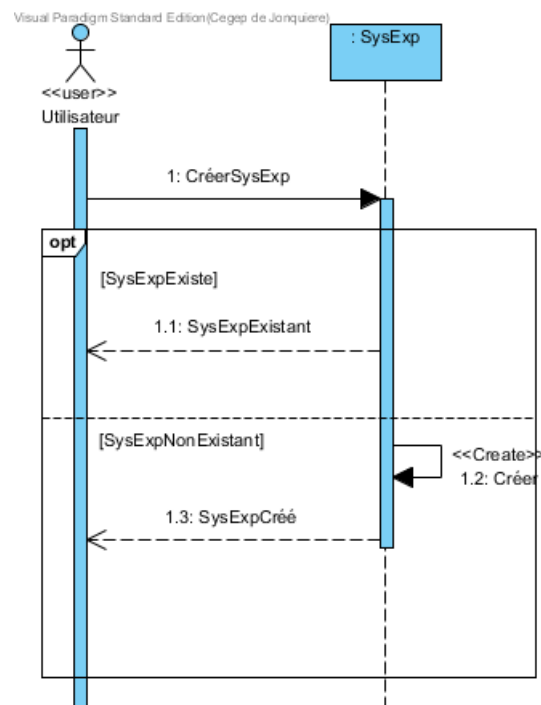


Figure 3 : DS_Créer_Sys_Exp_Simple

- Le fragment combiné *opt* indique une condition posée sur l'existence du système d'exploitation.
- Le stéréotype « create » indique la création d'un objet.

Diagramme de séquence Créer_Plateforme_Simple

Le diagramme de séquence associé à la création d'une plateforme est illustré à la Figure 4.

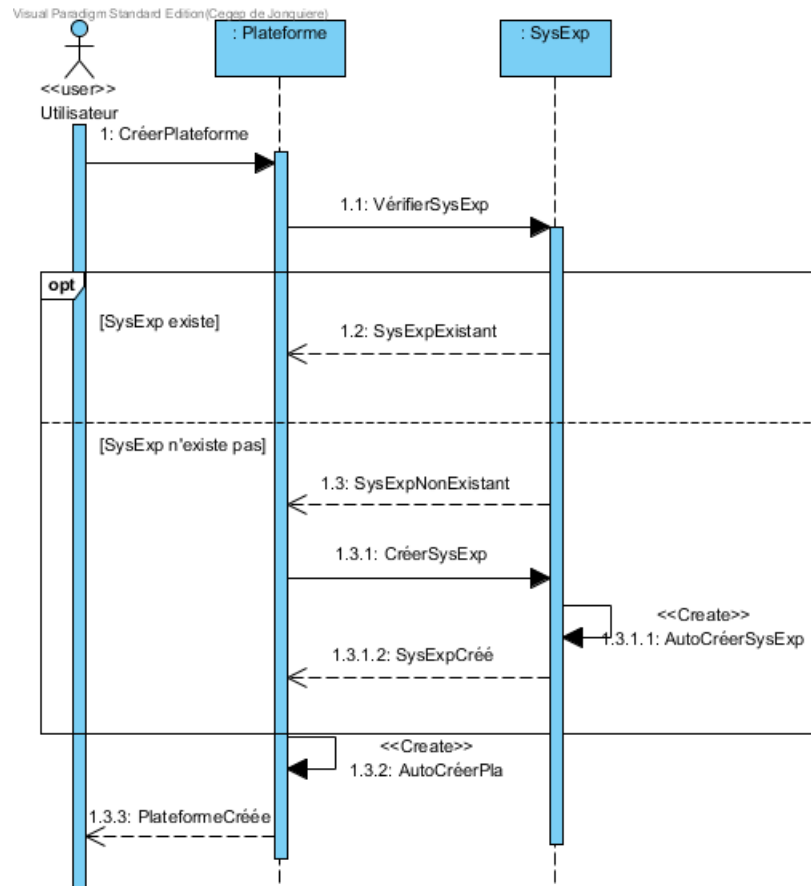


Figure 4 : DS_Créer_Plateforme_Simple

- Nous avons supposé ici que la plateforme n'existait pas.
- Comme vous pouvez le remarquer nous ne pouvons pas réutiliser le diagramme de séquence de création d'un système d'exploitation

3. Améliorations

Diagramme de séquence système d'exploitation réutilisable

- L'idée est de ne plus « communiquer » directement avec un objet SysExp mais de passer par une interface homme/machine frmSysExp.
- L'objet frmSysExp possède un stéréotype *boundary* utilisé pour les interfaces en général.

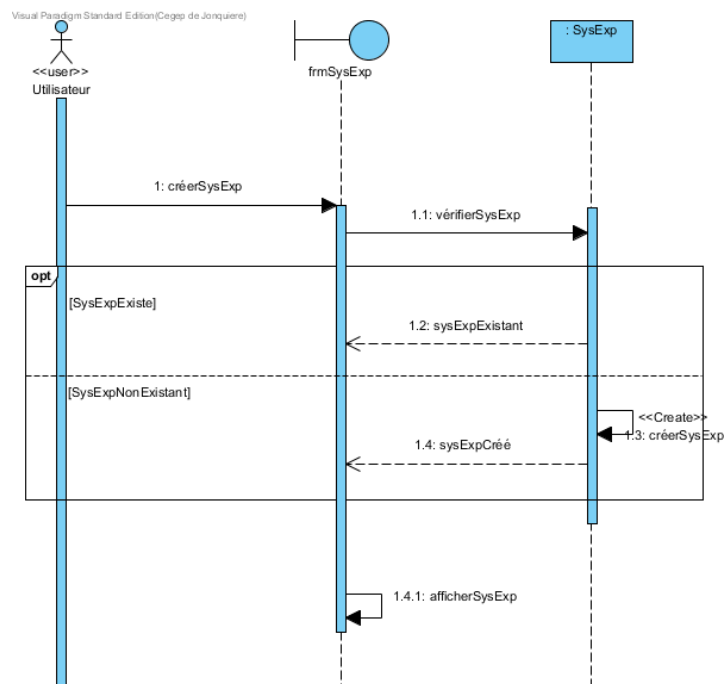


Figure 5 : DS_Créer_SysExp_amélioré

- Un inconvénient majeur ici est le fait d'intégrer le code de gestion des objets systèmes d'exploitation dans les formulaire/IHM. Vous l'avez deviné cette solution sera rejetée !

Diagramme de classe MVC

- Afin de mieux modéliser notre cas et avoir du code plus propre, le diagramme de classes de la Figure 1 est amélioré pour prendre en charge le modèle MVC.

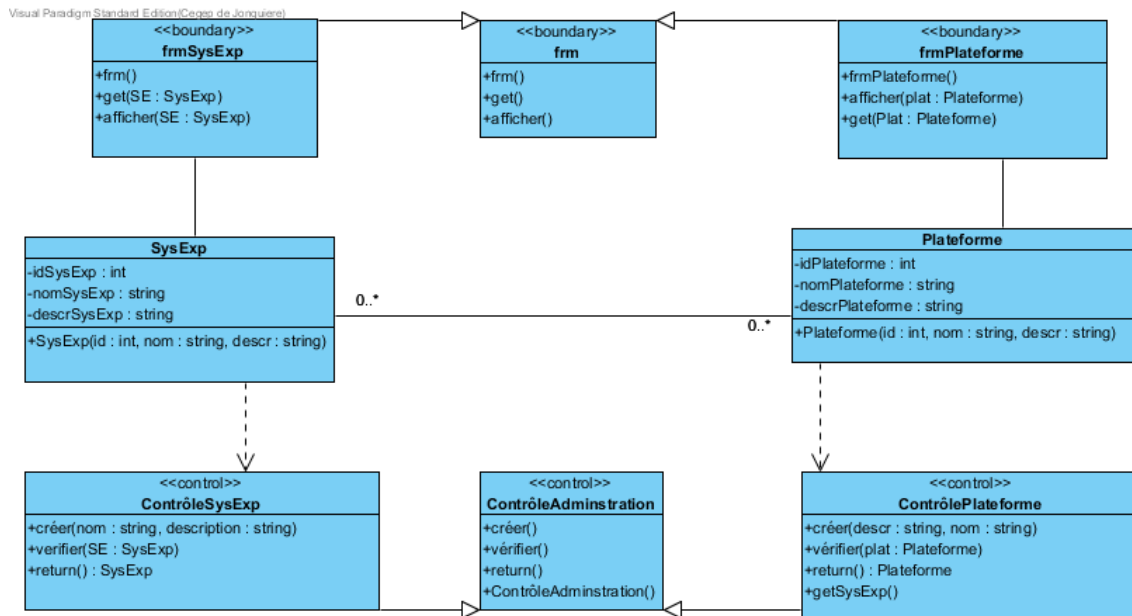


Figure 6 : DC_MVC

- Une classe ContrôleAdministration, mère de toutes les classes de contrôle est créée.
 - Cette classe servira à créer et gérer les objets dans le système.
 - ContrôlePlateforme et ContrôleSysExp héritent de cette classe les trois méthodes en les spécialisant (override).
 - Toutes ces classes ont un stéréotype « control ».
 - Les deux classes filles ont une dépendance respectivement par rapport à Plateforme et SysExp.
- De même, nous créons une classe mère frm et ses filles frmPlateforme et frmSysExp.
 - Toutes ces classes sont de stéréotypes « boundary ».
 - Les deux classes filles sont associées respectivement à Plateforme et SysExp.

Diagramme de séquence Créer_Sys_Exp final

Nous obtenons le diagramme de séquence de création d'un système d'exploitation suivant :

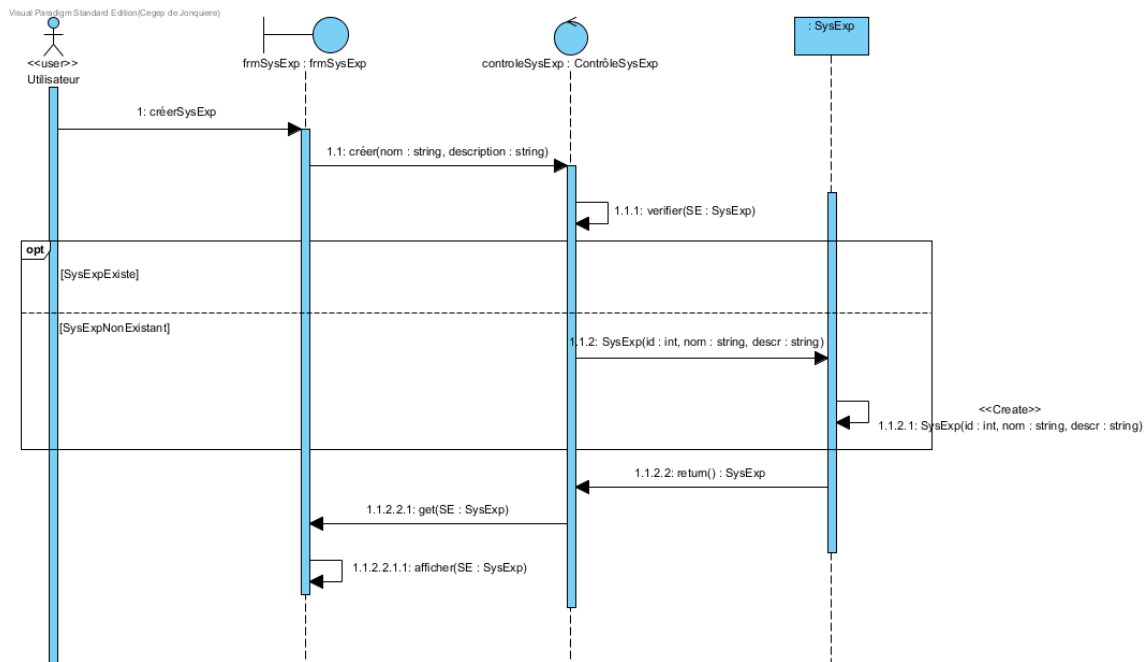


Figure 7 : DS_Sys_Exp_Final

Diagramme de séquence Créer_Plateforme final

Nous obtenons le diagramme de séquence de création d'une plateforme suivant :

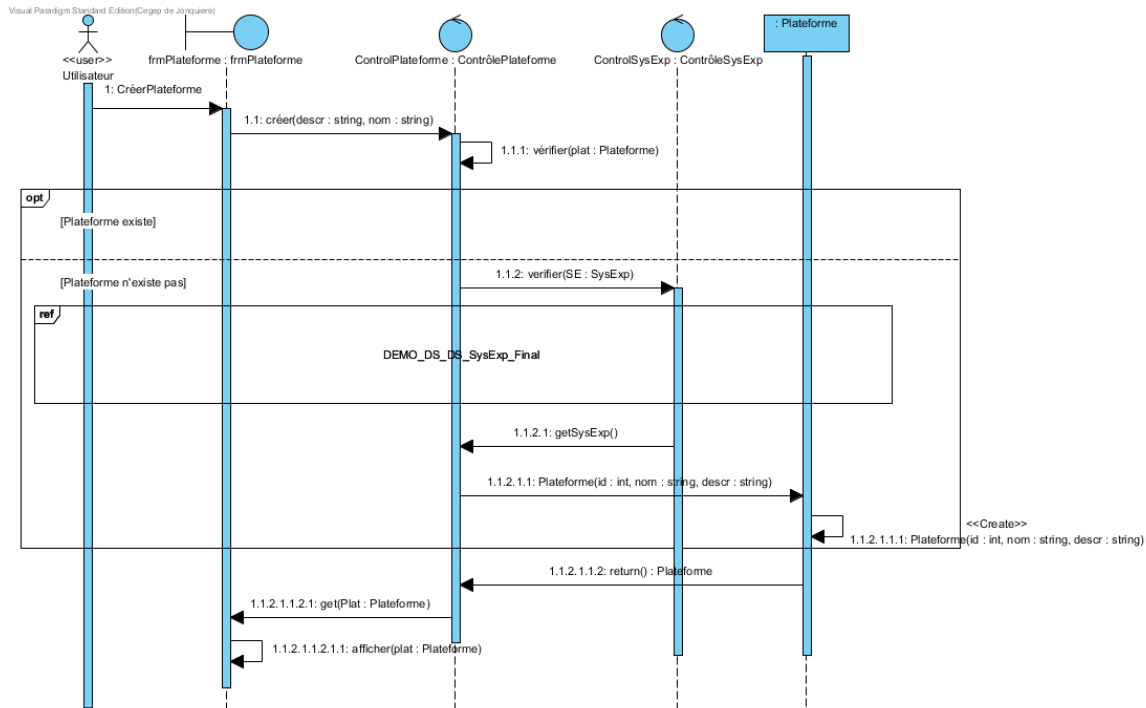


Figure 8 : DS_Plateforme_Final

- Le fragment combiné *opt* conditionne l'existence de la plateforme.
- Le fragment combiné *ref* renvoie vers le diagramme de séquence de création d'un système d'exploitation.