Cahier des charges

Sommaire

[1. Résolution du bug « Save as » 2](#_Toc348542707)

[2. Résolution du bug de non-coïncidence entre flèche de frappe et de bond 3](#_Toc348542708)

[3. Problème d’optimisation 4](#_Toc348542709)

[4. Mise à jour du graphe via des modifications sur le fichier texte 5](#_Toc348542710)

[5. Sauvegarde des paramètres utilisateurs (positions des sortes, couleur des sortes et du fond, groupes créés et leur couleur, police du texte) 7](#_Toc348542711)

[6. Ajout de fonctionnalités 7](#_Toc348542712)

[7. Résolution du bug de fonctionnement de gPH 10](#_Toc348542713)

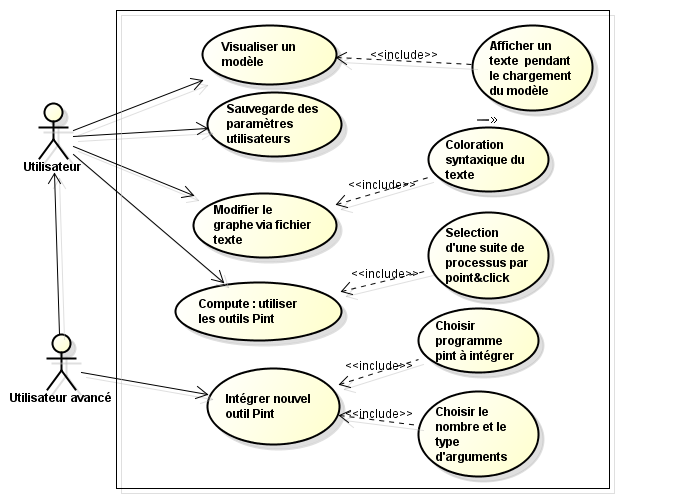
[8. Mettre une barre de chargement ou au minimum un texte « chargement en cours » lors de chargement de modèle 10](#_Toc348542714)

[9. Coloration syntaxique 11](#_Toc348542715)

[10. Permettre la sélection d’une suite de processus (en vue d’une utilisation dans le branchement d’un nouveau module) 12](#_Toc348542716)

[11. Outil Pint « Compute Reachability » 13](#_Toc348542717)

On peut trouver ci-dessous le diagramme d’utilisation pour notre projet d’application résumant les fonctionnalités que nous projetons d’implémenter dans le logiciel gPH Interface.



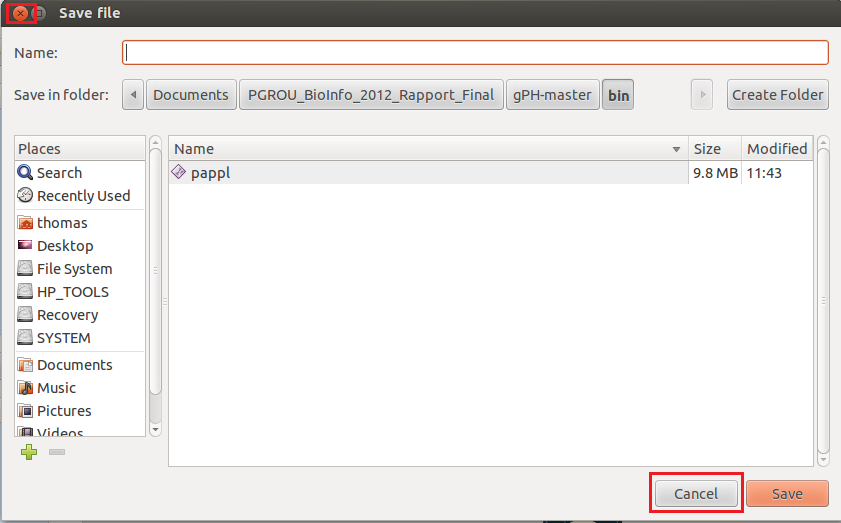
Priorité 1 :

# Résolution du bug « Save as »

Description du problème : Lors de la sauvegarde d’un modèle process hitting (File > Save as..), une fenêtre s’ouvre invitant l’utilisateur à choisir l’emplacement de sauvegarde. Il existe sur cette fenêtre un bouton « Cancel » permettant à l’utilisateur d’annuler sa sauvegarde et de fermer la fenêtre. C’est à cet instant que le bug intervient :

* Déroulement actuel : le programme crash (fermeture inopinée du logiciel)
* Déroulement attendu : fermeture de la fenêtre de sauvegarde et réactivation de la fenêtre principale

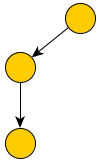
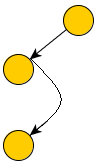
Ce bug se produit également lors de la fermeture de cette fenêtre en cliquant sur la croix en haut à gauche.



# Résolution du bug de non-coïncidence entre flèche de frappe et de bond

Description du problème : L’affichage d’un modèle process hitting doit respecter certaines règles pour obtenir une bonne visualisation. Une de ces règles concerne les flèches de frappe et de bond. Le bond étant provoqué par une frappe, la flèche de bond doit être directement liée à la flèche de frappe par la coïncidence de ces deux flèches. Il y a à ce niveau là un bug d’affichage :

Affichage actuel : Affichage attendu :

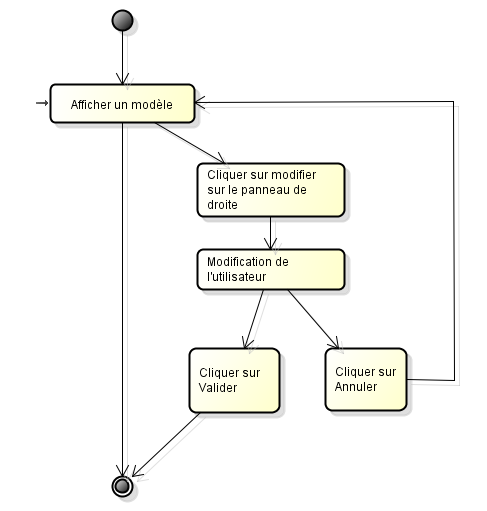
# Problème d’optimisation

Description du problème : Lorsque que l’on ouvre un nouveau modèle, le temps de chargement est trop long pour peu que le modèle possède plus de 20 sortes. Le temps de chargement croît non pas linéairement en fonction du nombre de sortes mais exponentiellement. Le problème serait lié à l’optimisation de l’affichage réalisée par Graphviz. En effet, les contraintes d’affichage, comme par exemple minimiser le nombre de croisement de flèches, sont très fortes et demandent la réalisation de nombreux calculs.

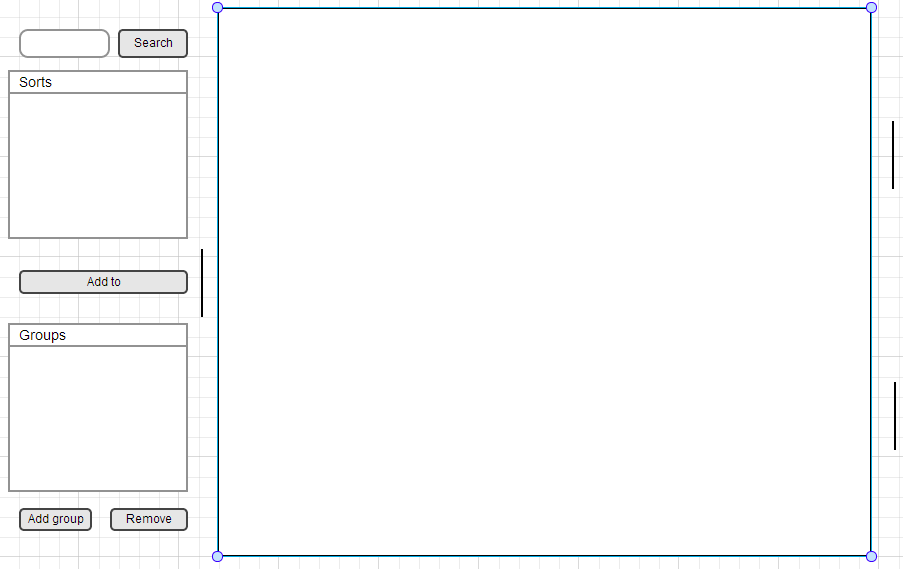
Solution envisagée : Diagnostiquer le souci dans GraphViz et tenter de trouver une solution. Le cas échéant, développer un gestionnaire de graphe plus rudimentaire plaçant les éléments « manuellement », dans ce dernier cas, on peut se permettre un affichage moins esthétique du modèle : croisement des flèches de frappes par exemple.

# Mise à jour du graphe via des modifications sur le fichier texte

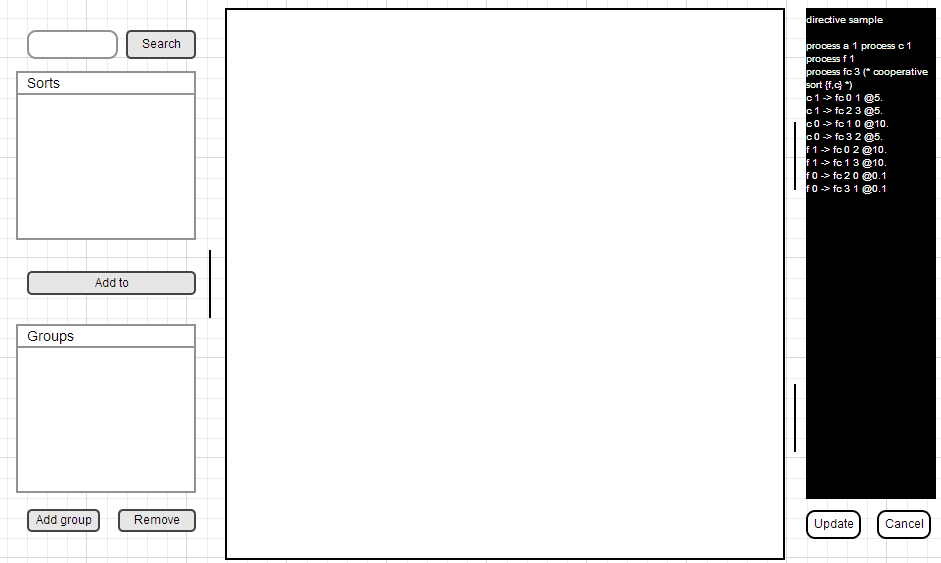
Il serait intéressant d’implémenter une fonctionnalité permettant à l’utilisateur, s’il est familier avec Pint, de modifier directement le graphe en modifiant le graphe via le fichier texte au format .ph.



Solution envisagé : Par défaut, le fichier texte serait en mode édition mais le panneau droit latéral serait rétracté afin d’éviter des modifications non voulues. Ensuite une fois le panneau déployé, l’utilisateur pourrait faire ses modifications puis les visualiser sur le graphe via un bouton « Update », ou alors les annuler via un bouton « Cancel »



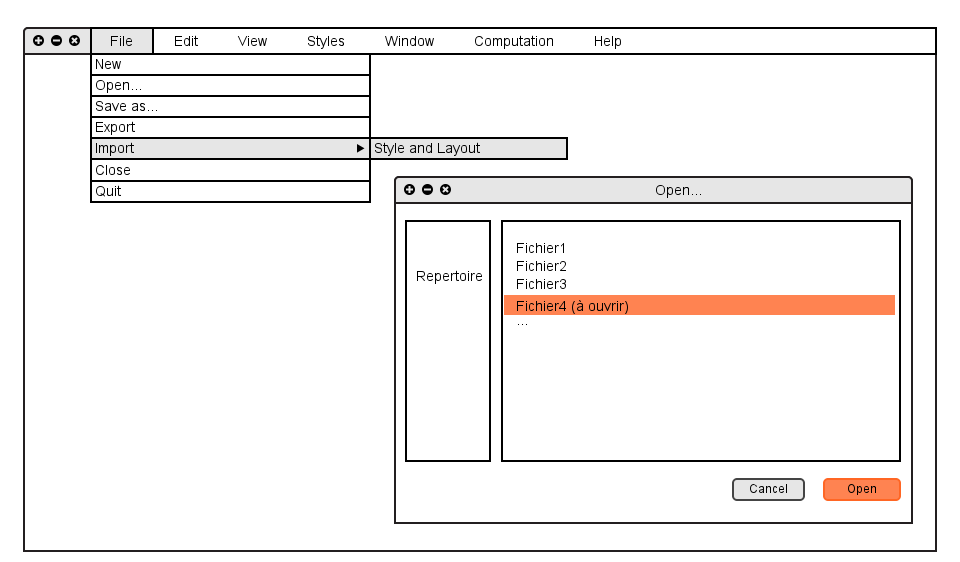
Etape 1 : Par défaut, panneau de droite rétracté



Etape 2 : Panneau déployé, modification possible

# Sauvegarde des paramètres utilisateurs (positions des sortes, couleur des sortes et du fond, groupes créés et leur couleur, police du texte)

Le groupe précédent a initié l’implémentation d’une fonctionnalité permettant de sauvegarder sous format .xml les préférences de l’utilisateur comme la couleur des sortes par exemple. Il faudrait terminer l’implémentation de la fonction export (position des sortes non exportée) et continuer en ajoutant la fonction import des préférences. Le menu *File* se verrait ajouté un sous-menu *Import* comme l’illustre la maquette suivante.



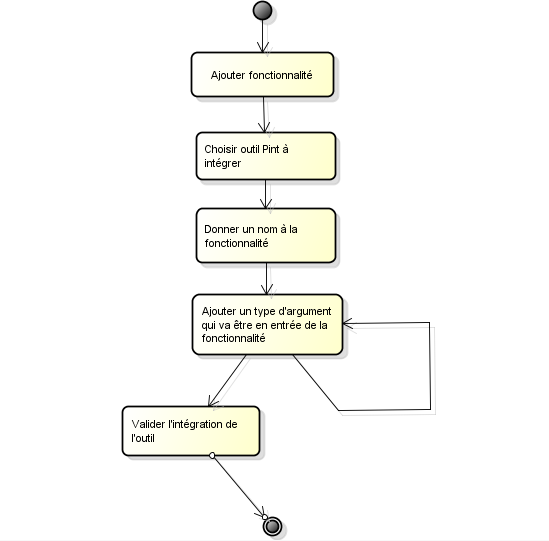
# Ajout de fonctionnalités

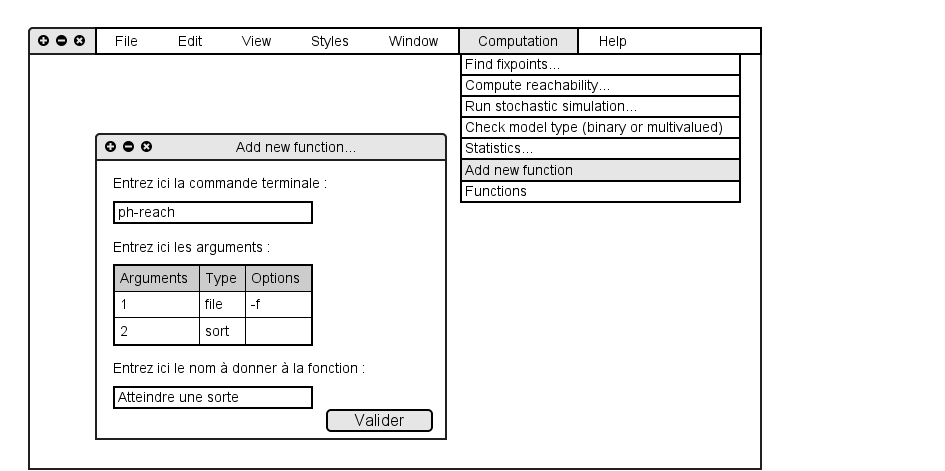
Description : Il serait intéressant d’implémenter une fonction permettant l’intégration de nouveaux outils Pint à gPH Interface, en plus des fonctions *Find Fixed Points, Compute reachability, Run Stochastic simulation, Check model type* et *Statistics.* Cette fonctionnalité comporterait deux volets.

Un premier représenté par un nouveau menu *Add Function*, utilisé par une personne un minimum familière avec Pint, qui pourrait ajouter l’outil souhaité et spécifier les arguments pris en entrée par l’outil en question. Les arguments en entrée seraient de types :

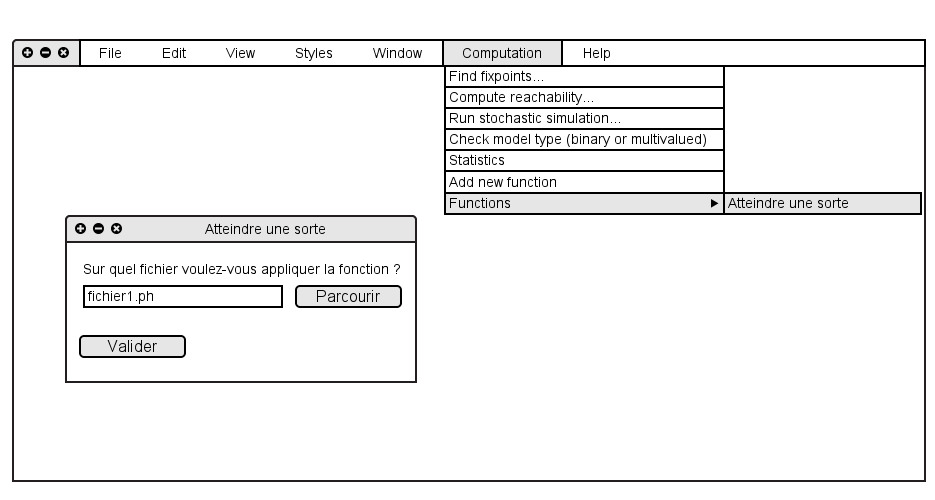
* Texte
* Entier
* Booléen
* Un processus ou une suite de processus
* Fichier ou répertoire
* Un choix à faire, par exemple : true or false, qui se ferait via un menu déroulant

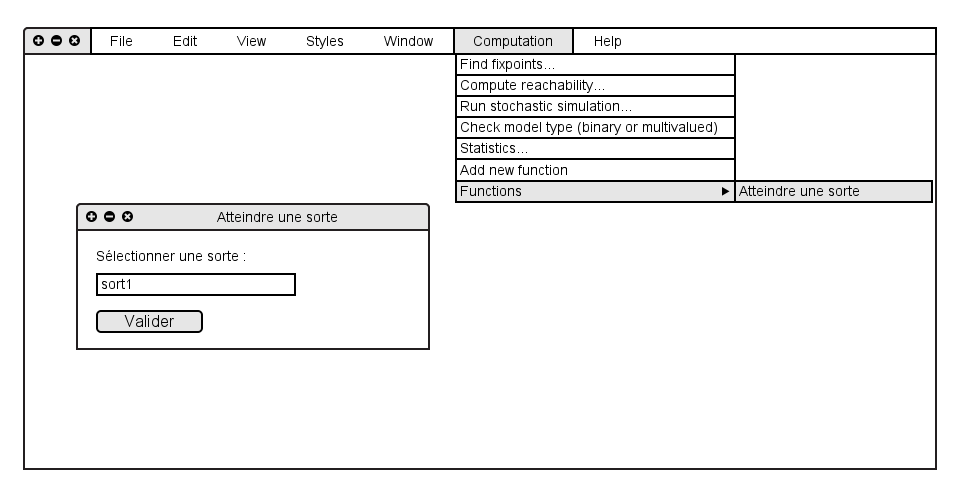
Les arguments en entrée pourrait aussi être spécifié facultatifs ou non.





Le deuxième volet est le nouvel outil intégré au programme et utilisable par le biologiste. En cliquant sur le nouveau sous-menu représentant le nouvel outil, le programme lui demandera les différents arguments nécessaires.





Au final, on aurait en sortie de l’outil, soit un fichier, soit un booléen, soit du texte.

Le formulaire de demande d’arguments n’est pas encore définitif : soit on demanderait tous les arguments sur une page, soit sur plusieurs.

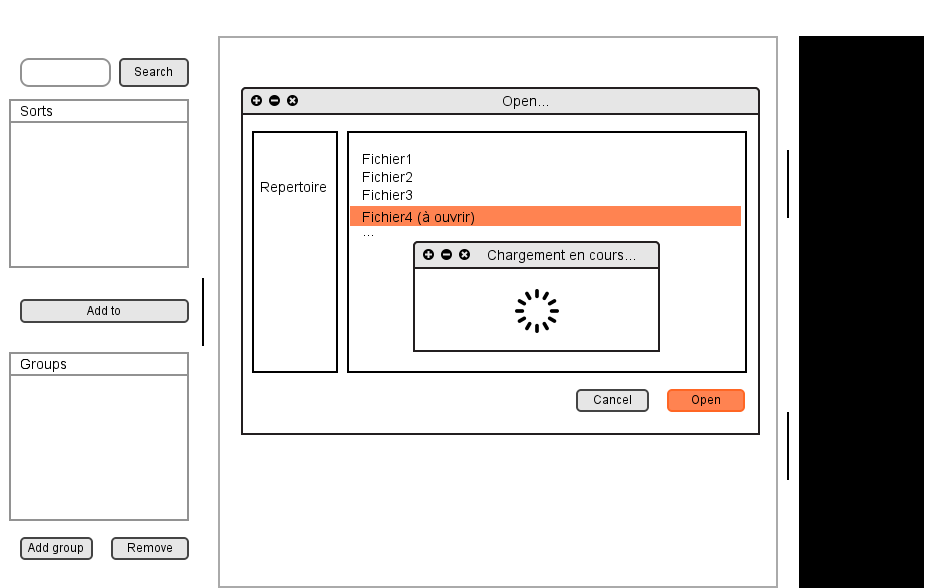
Priorité 2 :

# Résolution du bug de fonctionnement de gPH

Description : Le logiciel gPH Interface ne fonctionne pas si la langue d’installation du système d’exploitation utilisé (Ubuntu) est le français. Pour le moment, gPH ne fonctionne que sur Ubuntu installé en anglais. Le problème serait du à la mauvaise interprétation de la virgule ou du point dans les fichiers d’entrée (fichiers .ph).

# Mettre une barre de chargement ou au minimum un texte « chargement en cours » lors de chargement de modèle

Pour des modèles comportant un grand nombre de sortes, le chargement peut être long et l’utilisateur ne sait pas si le programme continue de charger ou à arrêter de fonctionner. Il serait intéressant d’afficher un message indiquant à l’utilisateur le chargement du programme.

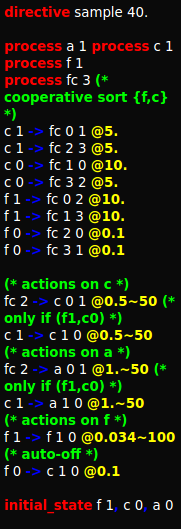


# Coloration syntaxique

Description du problème : La zone de texte décrivant le modèle (affichage du fichier .ph) dans le panneau latéral est affiché blanc sur noir. Il serait préférable d’afficher le texte sur fond blanc et de colorer ce dernier en appliquant une couleur pour chaque type d’éléments du formalisme Process Hitting.

Règles syntaxiques à appliquer :

* Signes de ponctuation en bleu
* Facteur et degré d’absorption stochastique ( @ r ~ sa) en jaune
* Les commentaires (\* …. \*) en vert fluo
* Les éléments du header, footer et body en rouge



# Permettre la sélection d’une suite de processus (en vue d’une utilisation dans le branchement d’un nouveau module)

Dans le cas de l’intégration d’un nouvel outil Pint au programme via la fonctionnalité no 6, l’utilisateur peut être amené à rentrer un nombre x de processus en argument en entrée. Il serait intéressant de permettre l’entrée d’une suite ordonnée de processus en les sélectionnant dans un volet affichant tous les processus du modèle. Il est important que le logiciel considère la sélection comme une sélection ordonnée des processus.

# Outil Pint « Compute Reachability »

L’appel de l’outil « compute reachability » via l’interface graphique ne fonctionnne pas. Il faudrait rendre cet outil utilisable.