

1/ Combien de processus seront créés dans le projet ?

chaque exécution de ./acquisition crée 3 processus (acquisition, terminal et validation) qui communiquent entre eux. Le programme servant à tester les différentes possibilités (./start) crée 4 acquisition, lors de l'exécution de start il y a donc un total de 13 processus mais ils ne s'exécutent pas tous simultanément.

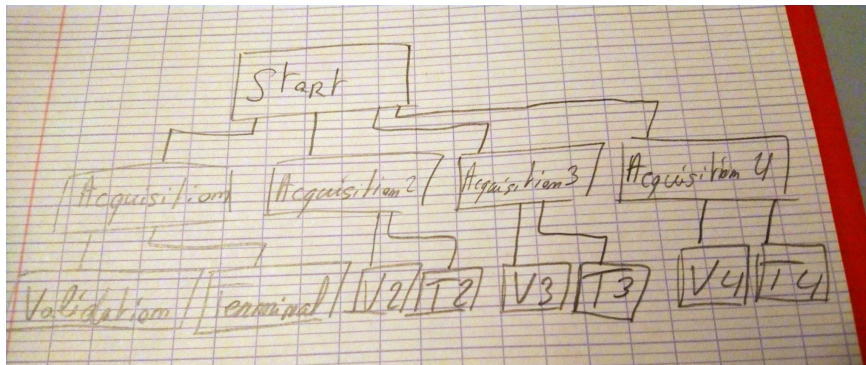
2/ Combien d'exécutables allez-vous programmer ? Est ce qu' un seul exécutable est acceptable ?

Il y a un total de 4 exécutables : terminal, validation, acquisition et start. Un seul exécutable n'est pas acceptable car pour utiliser execlp, il faut que le programme à exécuter soit compilé sous forme d'exécutable.

3/ Qui (quel processus) est créé par qui (en précisant les arguments de création) ?

Le processus start crée 4 acquisition avec chacun un numéro et une valeur de test. Chaque processus acquisition crée un processus terminal avec en argument deux pipes afin de pouvoir communiquer après le execlp ainsi que le numéro et la valeur qu'il avait lui même en argument, il crée aussi un processus validation avec en argument deux autres pipes également pour communiquer.

4/ Dessinez l'arbre des processus du projet.



5/ Combien de tubes sont nécessaires pour le projet ?

Chaque Acquisition doit pouvoir communiquer avec Terminal et Validation, il lui faut donc 4 tubes (2 pour écrire vers chaque processus et 2 autres pour lire depuis chaque processus).

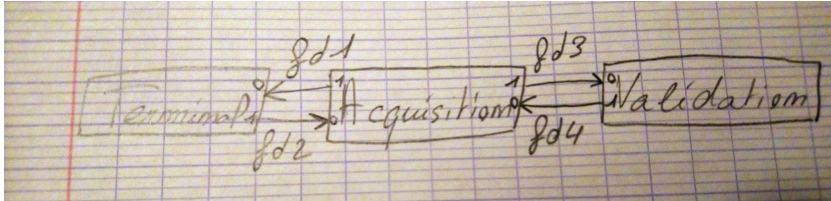
6/ Quels types de tubes allez-vous utiliser ?

J'ai utilisé des tubes non nommés.

7/ Qui (quel processus) crée quels tubes ?

Le processus acquisition crée tous les tubes

8/ Dessinez vos processus et vos tubes.



9/ Avec quel mécanisme de communication pourront nous remplacer les tubes ? et pourquoi ?

On pourrait remplacer les tubes par des sockets, cela rendrait même possible la communication entre machines, ce qui rendrait le projet plus réaliste.

10/ Est ce que l'utilisateur pourrait taper manuellement (au niveau du terminal) une requête (numéro de test PCR et date de passage) ? (si la réponse est non, justifier la raison)

Oui car le processus acquisition prend en arguments un numéro et une valeur donc l'utilisateur peut taper ./acquisition numéro valeur.

11/ Que se passe-t-il si le numéro du test PCR de la requête ne figure dans aucune base de données de résultats ?

Le processus validation affiche d'abord "[validation] : Erreur de correspondance !!\n" puis transmet une réponse négative (avec une valeur de 0). Le processus terminal affichera donc à la fin "[terminal] : Le test n'est pas valide.\n".

12/ Est t'il possible pour un serveur d'acquisition de gérer plusieurs terminaux à chaque fois ? (si la réponse est oui, justifier l'idée ou la méthode à utiliser)

Mon code actuel n'est pas prévu pour. Le problème est qu'un serveur d'acquisition attend de pouvoir lire dans le tube lié à son terminal pour continuer son exécution donc cela ne semble pas possible en utilisant des tubes.

13/ Le serveur interArchive peut communiquer avec tous les serveurs d'acquisitions qui lui sont reliés (à travers les pipes), comment est gérée cette communication ? (exemple: si une architecture contient 4 serveurs d'acquisitions, 1 terminal et un serveur interArchive, ce dernier reçoit une requête, comment il va faire l'acheminement de cette requête vers le bon serveur d'acquisition ?)

Je n'ai pas été jusque là dans la projet, mais il faudrait que le processus interArchive soit celui qui crée tous les processus Acquisition afin de leur transmettre 2 tubes chacun. Pour acheminer les requêtes il devra alors vérifier si les premiers chiffres du numéro du test correspondent à un serveur Acquisition et envoyer la requête vers le serveur correspondant via le tube précédemment créé.

14/ C'est quoi le critère d'arrêt de fin de vos processus (Signifiant l'arrêt de l'exécution) ?

Le terminal a reçu la réponse et affiché si le test est valide ou non.

15/ Dans le but de perfectionner le projet en cherchant à appliquer le maximum de parallélisation, à quelle niveau (processus) peut-on utiliser du traitement multithreading et comment ?

Le multithreading pourrait être utilisé afin d'exécuter chaque centre de données (chaque duo Acquisition/Validation) parallèlement.