



Les processus et services

M1 - Architecture Système et réseau



Définition

- Un **Programme** est une suite d'instructions que le système doit faire accomplir au processeur. Ces instructions sont rangées dans un fichier.
- Un **processus** est un programme chargé en mémoire.
- Les **services** sont des processus chargés par le processus de session (au démarrage).

Contexte d'exécution

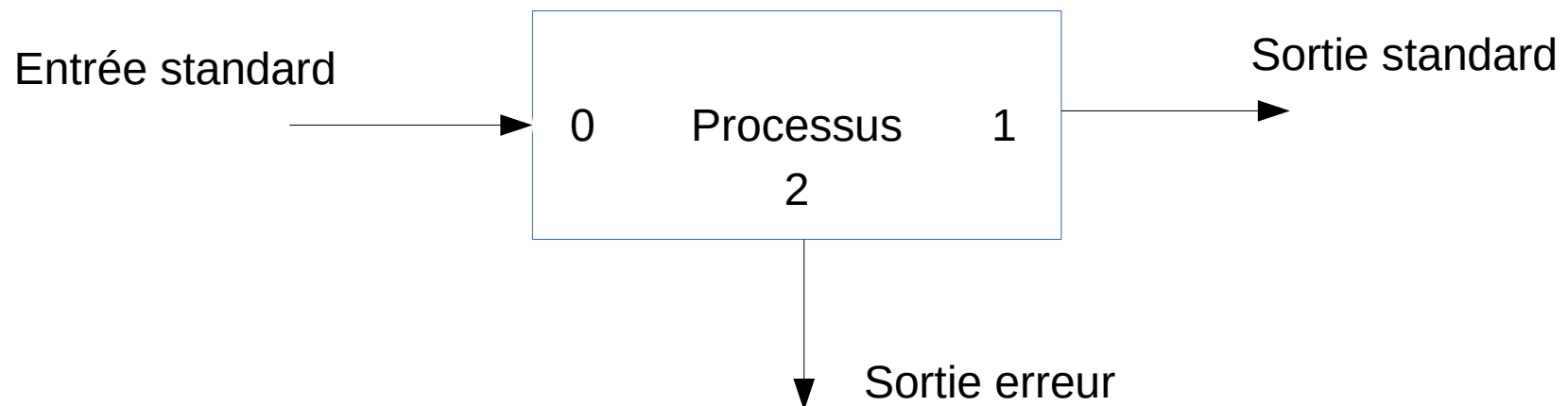
- Le noyau maintient une table pour gérer l'ensemble des processus.
- Chaque processus est identifié par un index dans cette table **son numéro d'identification ou PID**
- Chaque entrée de la table correspond aux informations sur ce processus :
 - Numéro d'identification du processus père
.....**PPID**
 - Identification de l'utilisateur qui exécute le processus.....**UID**
 - Identification de groupe qui exécute le processus.....**GID**
 -

Les processus

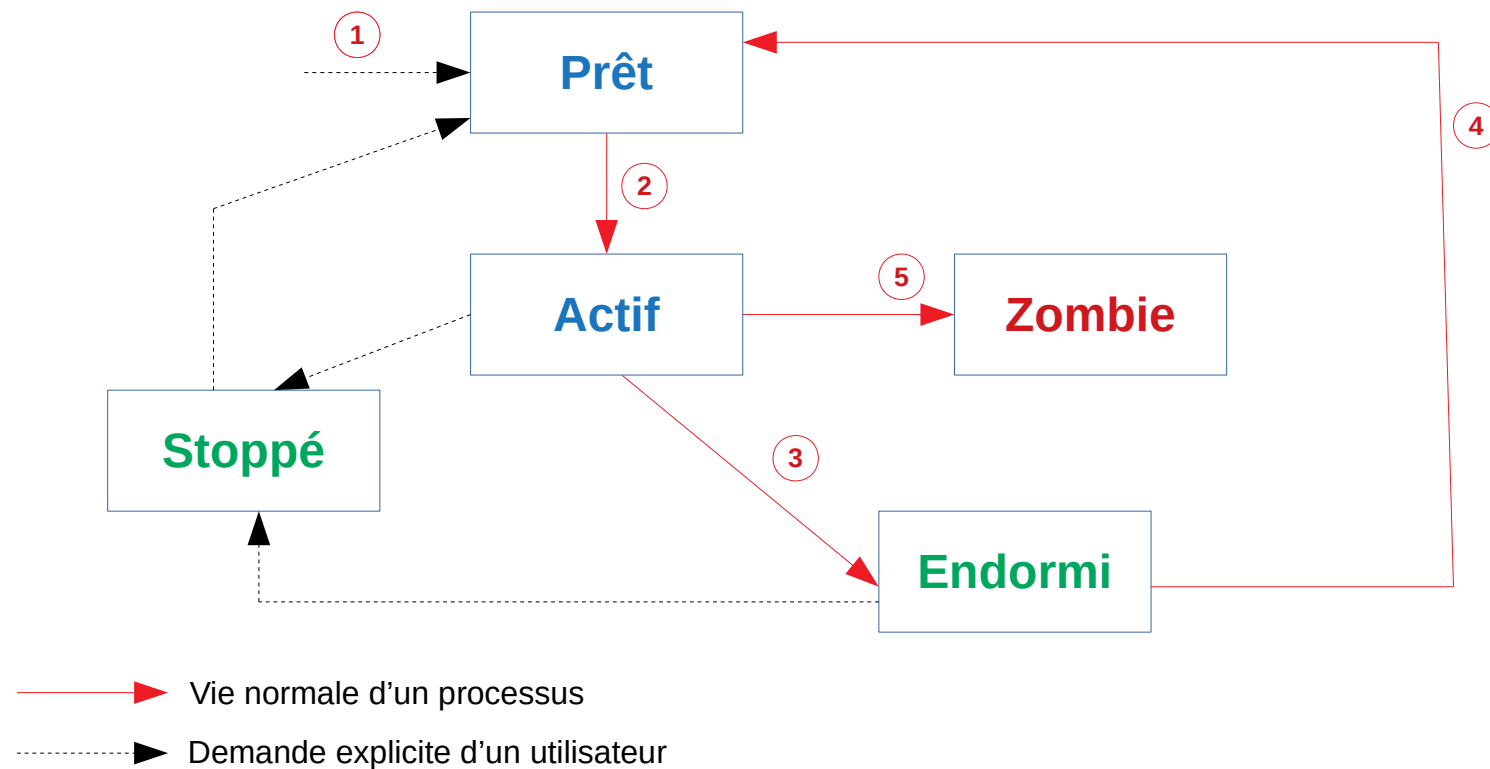
Chaque index de cette table est appelé un descripteur de fichier.

Les trois premiers descripteurs sont :

- l'entrée standard
- La sortie standard
- La sortie d'erreur



Changements d'états



Modes de fonctionnement

- Un processus peut
 - Attendre la fin de son fils pour continuer, **avant plan** (**foreground**)
mode synchrone : les processus s'exécutent en séquence
 - Ne pas attendre la fin du fils **tâche de fond** (**background**).
mode asynchrone : les processus s'exécutent en parallèle.
- Pour chaque commande exécutée le shell crée un nouveau processus.
 - Par défaut en mode synchrone
 - Les commandes peuvent être séparées par :
 - Des **points-virgules**
; Attendre la fin d'une commande pour passer à la suivante
 - Des **esperluettes**
& Ne pas attendre la fin d'une commande pour passer à la suivante
 - Des **tubes**
| Démarrer les commandes en parallèles en les connectant.

Les services

- Les services sont lancés par des scripts (System V ou LSB) situés dans /etc/init.d.
- Depuis debian 8, systemd offre une alternative au démon sysvinit apportant comme avantages :
 - un meilleur cadre pour la gestion des dépendances entre services
 - le chargement en parallèle des services au démarrage
 - la réduction des appels aux scripts shell
- Quelques modifications :
 - service example status → systemctl status example
 - service example stop → systemctl stop example
 - service example start → systemctl start example
 - update-rc.d example defaults → systemctl enable example

Le planificateur de tâches

- crontab : **exécuter des commandes périodiques**
- crontab est une commande qui permet d'exécuter des actions à des intervalles de temps réguliers.
- Ses avantages :
 - Exécution sans présence de l'utilisateur
 - Possibilité de sortie des commandes par mail
 - Choix d'intervalles multiples : minutes, heures, jours et même en mois.