

Introduction à UNIX

IFT-1214

Université de Montréal



Introduction

Historique

- 1966 – Début de la création d'UNIX
- 1969 – première version d'UNIX
- 1973 – Réécriture de UNIX en C
- 1974 – premières licences pour universités
- 1978 – premières versions commerciales
- Années 80 – Autorisation de clonage
 - System V – moyennes et petites configurations
 - BSD – configurations importantes

Introduction

Fonctionnalités

- Gestion des ressources de l'ordinateur
 - Multi tâche, multi usagers, multi processeur
- Gestion des données
 - Organisation et accès aux unités de stockage
- Communication entre utilisateurs
 - Transfert de fichiers, courrier électronique
- Environnement de programmation
 - Compilateurs (C, C++, Fortran, etc.)
 - Editeurs (Emacs, Vi, etc.)

Introduction

Documentation

- Web
 - Gildas Perrot
 - http://www.francenet.fr/~perrot/cours_unix-Contents.html
 - Séminaires UNIX du DIRO
 - <http://www.iro.umontreal.ca/~semunix>
- Livre
 - UNIX
 - Par D. Taylor et J. C. Armstrong Jr.
 - Collection "Le tout en poche"
 - Campus Press, France

Organisation de la Matière

- Bases
- Système de fichiers
- Manipulation des fichiers
- Propriétés et droits d'accès
- Aspects réseau
- Gestion des processus
- Egrep et expressions régulières

Bases

Introduction à UNIX



Bases

UNIX au DIRO

- Locaux
 - 1^{er} étage : 1340, 3^{èm} étage : 3185 et 3189
- Comptes et support
 - Secrétariat du deuxième étage
 - <http://www.iro.umontreal.ca/support>
- Sessions
 - Login :



DIRO Linux 7.3 (blc33.iro.umontreal.ca)

Login:

Password:

Session Type: default ▾

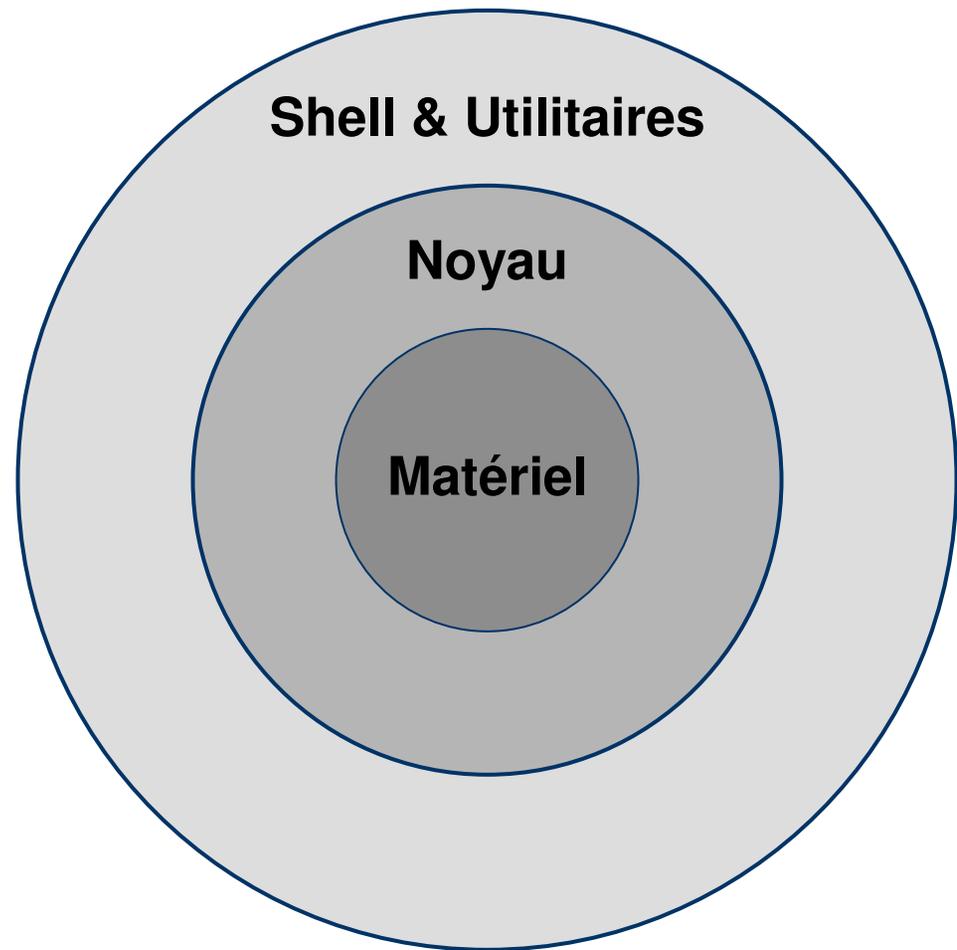
Gg! Clear Menu Shutdown...

- Logout : 

Bases

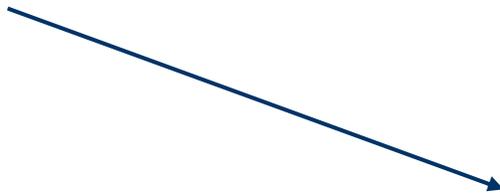
- Shell et utilitaires
 - Interface usager
 - Applications
- Noyau
 - Sys. d'exploitation
 - Gestions E/S
- Matériel
 - Processeur
 - Mémoire
 - Périphériques

Décomposition



Bases Interpréteur de Commandes

- Caractéristiques
 - Comparable au *DOS-Prompt* sous windows
 - Permet un meilleur contrôle sur les applications
 - Permet d'utiliser certains outils non graphiques
- Ouverture d'un interpréteur (terminal)



```
xterm
Sun Microsystems Inc. SunOS 5.9 Generic May 2002
You have new mail.
twister:<gorse>% █
```

Bases

Commandes

- Syntaxe
 - commande [options] <arguments>
 - Séparateur : caractère espace
- Commande
 - Action à accomplir ou application à démarrer
- Arguments
 - Objets ou fichiers auxquels la commande s'applique
- Options
 - Modification du comportement de la commande
 - Commencent généralement par un - (moins)

Bases

Exemples

- Exemple – La commande ls
 - **ls** : Affiche la liste des fichiers et répertoires
 - **ls -l** : Affiche une liste détaillée des fichiers et répertoires
 - **ls -l fic1** : Affiche les informations sur le fichier **fic1** spécifié en paramètre
- Tout savoir sur une commande : man
 - **man <commande>**
 - Affiche la page du manuel sur la commande citée en argument (**<commande>**)

Systeme de fichiers

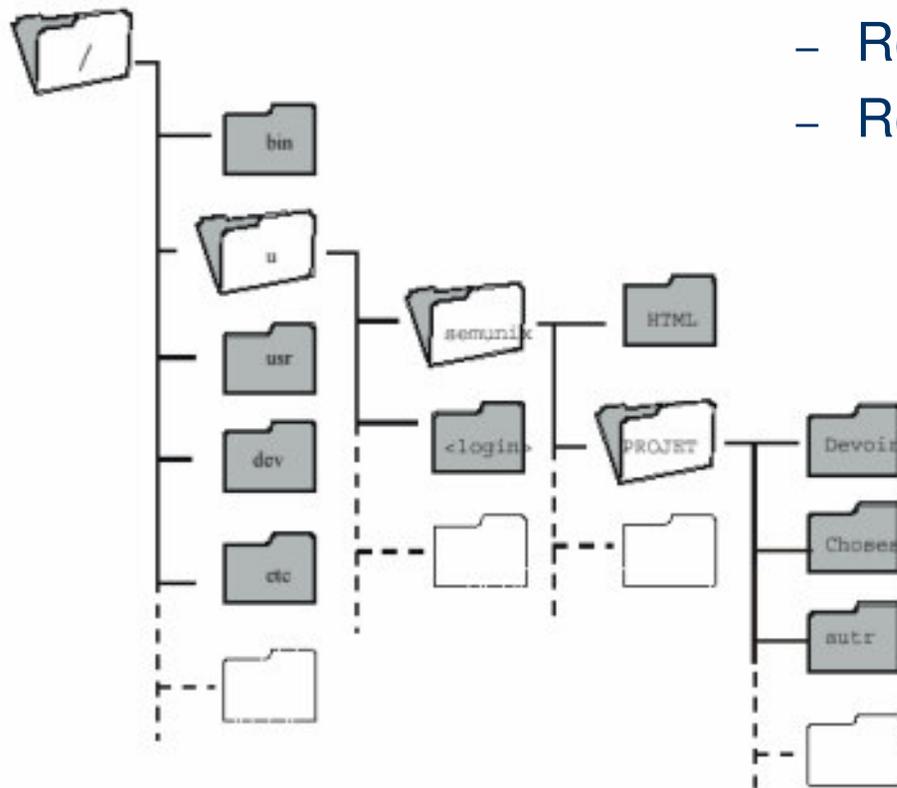
Introduction à UNIX



Systeme de Fichiers

Hiérarchie

- Une seule unité dont la racine est /



- Répertoire actuel : .
- Répertoire parent : ..

Systeme de Fichiers

Définitions

- Répertoire principal (HOME)
 - /u/votrelogin/
 - Ex. /u/gorsen
- Chemin absolu d'un fichier (nom absolu)
 - Liste des répertoires traversés pour y accéder
 - Ex. /u/gorsen/HTML/teaching/index.html
- Chemin relatif d'un fichier (nom relatif)
 - Liste des répertoires traversés à partir du répertoire dans lequel on se trouve à ce moment là
 - Ex. HTML/teaching/index.html

Systeme de Fichiers

Déplacement

- pwd
 - Donne le chemin du répertoire actuel (.)

```
twister% pwd
/u/gorsen
```

- cd
 - Permet de changer de répertoire

```
twister% cd PhD
twister% pwd
/u/gorsen/PhD
```



```
twister% cd /u/gorsen/PhD
twister% pwd
/u/gorsen/PhD
```

```
twister% cd ..
twister% pwd
/u/gorsen
```

Systeme de Fichiers

Listes

- Commande : ls
 - Affiche la liste des fichiers et répertoires
- Options courantes ls
 - -l : affichage des fichiers avec les champs suivants
 - Permissions
 - Nombre de références (sous répertoires)
 - Usager
 - Groupe
 - Taille
 - Date de dernière modification
 - Nom
 - -a : affichage des fichiers commençant par un .

Systeme de Fichiers

Exemples

```
hamurabi% pwd  
/u/gorsen/HTML
```

```
hamurabi% cd  
hamurabi% pwd  
/u/gorsen
```

```
hamurabi% ls  
HTML  PhD  admin  archives  docs  local  loisirs  progs  tmp
```

```
hamurabi% cd PhD/enseignement/IFT-1214/
```

Systeme de Fichiers

Exemples

```
hamurabi% ls
Calendrier.pdf  FPNumbers.ppt  FSM.ppt
IntData.ppt    JavaScript     Unix   temp

hamurabi% ls -l
total 2032
-rwx-----  1 gorsen  lasso      34639 Oct  7 20:46 Calendrier.pdf
-rwx-----  1 gorsen  lasso     480256 Oct  1 14:44 FPNumbers.ppt
-rwx-----  1 gorsen  lasso     154112 Oct  1 14:44 FSM.ppt
-rwx-----  1 gorsen  lasso    1378816 Sep 29 07:25 IntData.ppt
drwx-----  4 gorsen  lasso      4096 Oct 18 11:21 JavaScript
drwx-----  2 gorsen  lasso      4096 Oct 19 13:07 Unix
drwx-----  4 gorsen  lasso      4096 Oct 16 10:05 temp
```

Systeme de Fichiers

Particularités

- Identification via la commande **ls -l**
 - Le signe **-** désigne un fichier
 - La lettre **d** désigne un répertoire
 - La lettre **l** désigne un lien
- Noms de fichiers et répertoires
 - Limités à 255 caractères
 - Tous les caractères spéciaux (*, (, %, etc.) sont permis, on utilisera alors les guillemets "" pour envelopper le nom du fichier ou le caractère \ devant chaque caractère spécial.

Manipulation des Fichiers

Introduction à UNIX



Manip. des Fichiers

Commandes

- Création
 - **mkdir rep1** : Création du répertoire vide **rep1**
 - **touch fic1** : Création du fichier vide **fic1**
- Suppression
 - **rmdir rep1** : Suppression du répertoire **rep1**
 - **rm fic1** : Suppression du fichier **fic1**
 - Options de rm
 - -i : Confirmation de suppression
 - -f : Retrait de tout genre de confirmation
 - -r : Suppression récursive (répertoires)
 - -rf : Combinaison : suppression **récursive sans confirmation**

Manip. des Fichiers

Exemples

```
hamurabi% ls -l
total 0

hamurabi% mkdir repl
hamurabi% touch ficl
hamurabi% ls -l
total 4
-rw-----      1 gorsen   lasso           0 Oct 19 13:56 ficl
drwx-----      2 gorsen   lasso        4096 Oct 19 13:56 repl

hamurabi% rmdir repl
hamurabi% ls -l
total 4
-rw-----      1 gorsen   lasso           0 Oct 19 13:56 ficl

hamurabi% rm -i ficl
rm: remove regular empty file `ficl'? y
```

Manip. des Fichiers

Commandes

- Copie
 - **cp aaa bbb** : Copie **aaa** en **bbb** si **bbb** n'est pas un répertoire, sinon, copie **aaa** dans **bbb**
 - Si **bbb** est un fichier déjà existant, il sera remplacé par la copie de **aaa**
 - Options de cp
 - -i : Confirmation de copie
 - -f : Retrait de tout genre de confirmation
 - -r : Copie récursive (répertoires)
 - -rf : Combinaison : copie **récursive sans confirmation**

Manip. des Fichiers

Exemples

```
hamurabi% mkdir repl
hamurabi% touch fic1

hamurabi% cp fic1 fic2
hamurabi% ls -l
total 4
-rw-----  1 gorsen  lasso           0 Oct 19 13:56 fic1
-rw-----  1 gorsen  lasso           0 Oct 19 13:56 fic2
drw-----  1 gorsen  lasso        4096 Oct 19 13:56 repl

hamurabi% cp fic2 repl
hamurabi% ls -l repl
total 0
-rw-----  1 gorsen  lasso           0 Oct 19 13:56 fic2
```

Manip. des Fichiers

Commandes

- Déplacement
 - **mv aaa bbb** : Renomme **aaa** en **bbb** si **bbb** n'est pas un répertoire, sinon, déplace **aaa** dans **bbb**
 - Si **bbb** est un fichier déjà existant, il sera remplacé par **aaa** lorsque celui-ci sera renommé
 - Options de mv
 - -i : confirmation de déplacement
 - -f : retrait de tout genre de confirmation
- Espace
 - **du aaa** : Donne l'espace occupé par **aaa**

Manip. des Fichiers

Exemples

```
hamurabi% mkdir repl
hamurabi% touch ficl

hamurabi% ls -l
total 4
-rw-----  1 gorsen  lasso           0 Oct 19 13:56 ficl
drwx-----  2 gorsen  lasso       4096 Oct 19 13:56 repl

hamurabi% mv ficl repl
hamurabi% ls -l
total 4
drwx-----  2 gorsen  lasso       4096 Oct 19 13:56 repl

hamurabi% ls -l repl
total 0
-rw-----  1 gorsen  lasso           0 Oct 19 13:56 ficl
```

Manip. des Fichiers

Commandes

- Recherche d'un fichier
 - **find <point de départ> -name <fichier>** : Recherche les occurrences du fichier dont le nom est donné en paramètre dans l'arborescence se trouvant sous le point de départ spécifié
- Lecture du contenu d'un fichier
 - **cat fic** : Affiche le contenu du fichier **fic**
- Lecture du contenu d'un fichier pas à pas
 - **more fic** : Affiche le contenu de **fic** page par page
 - Prochaine page : presser la barre d'espace
 - Prochaine ligne : presser la touche [enter]

Manip. des Fichiers

Exemples

```
hamurabi% mkdir rep1

hamurabi% cd rep1
hamurabi% mkdir rep2
hamurabi% touch fic

hamurabi% cd rep2
hamurabi% mkdir rep3
hamurabi% touch fic

hamurabi% cd ../../

hamurabi% find rep1 -name fic
rep1/rep2/fic
rep1/fic
```

Manip. des Fichiers

Exemples

```
hamurabi% ls -l
total 4
drwx-----  2 gorsen  lasso      4096 Oct 19 13:56 repl
-rw-----  1 gorsen  lasso       44 Oct 19 13:56 texte
```

```
hamurabi% cat texte
ceci est le contenu
du fichier
sur 3 lignes
```

```
hamurabi% cd repl
hamurabi% ls -l
total 4
drwx-----  2 gorsen  lasso      4096 Oct 19 13:56 rep2
-rw-----  1 gorsen  lasso       0 Oct 19 13:56 fic
```

```
hamurabi% cat fic
```

Manip. des Fichiers

Commandes

- Affichage d'un champ particulier
 - La commande **cut** permet de filtrer du texte pour n'afficher que certains champs désirés
 - Commande : **cut -d<délimiteur> -f<champ>**
 - Délimiteur : caractère délimitant les différents champs
 - Champ : numéro de champ désiré
 - Utilisation
 - **cut -d<delimiteur> -f<champ> fichier**
 - **cat fichier | cut -d<delimiteur> -f<champ>**
 - Champs
 - Un seul : un seul numéro
 - Plusieurs : liste de numéros séparés par une virgule

Manip. des Fichiers

Exemples

```
hamurabi% cat texte  
login:prenom:nom:departement:cours  
gorsen:nicolas:gorse:diro:ift1214  
aboulham:mostapha:aboulhamid:diro:ift1214
```

```
hamurabi% cat texte | cut -d:-f1  
login  
gorsen  
aboulham
```

```
hamurabi% cat texte | cut -d:-f1,5  
login:cours  
gorsen:ift1214  
aboulham:ift1214
```

Manip. des Fichiers

Commandes

- Affichage du début d'un fichier
 - La commande **head** affiche les 10 premières lignes d'un fichier texte donné en paramètre
 - La commande **head -n 3** affichera seulement les trois premières lignes
- Affichage de la fin d'un fichier
 - La commande **tail** affiche les 10 dernières lignes d'un fichier donné en paramètre
 - La commande **tail -n 3** affichera seulement les trois dernières lignes
- La combinaison des deux est parfois utile

Manip. des Fichiers

Exemples

```
hamurabi% cat texte
login:prenom:nom:departement:cours
gorsen:nicolas:gorse:diro:ift1214
aboulham:mostapha:aboulhamid:diro:ift1214

hamurabi% head -n 1 texte
login:prenom:nom:departement:cours

hamurabi% tail -n 1 texte
aboulham:mostapha:aboulhamid:diro:ift1214

hamurabi% tail -n 2 texte | head -n 1
gorsen:nicolas:gorse:diro:ift1214

hamurabi% tail -n 2 texte | head -n 1 | cut -d: -f2
nicolas
```

Manip. des Fichiers

Commandes

- Archivage
 - **tar -cvf bbb.tar aaa** : Archive le contenu du repertoire **aaa** dans le fichier **bbb.tar**
 - **tar -xvf bbb.tar** : Extrait le contenu de l'archive **bbb.tar** et le place dans le répertoire courant
- Compression
 - **gzip fic** : Comprime le fichier **fic** qui sera renommé en **fic.gz**
 - **gzip -d fic.gz** : Décompresse le fichier **fic.gz** qui sera renommé en **fic**

Manip. des Fichiers

Exemples

```
hamurabi% tar -cvf arch.tar repl  
repl/  
repl/rep2  
repl/rep2/rep3  
repl/rep2/fic  
repl/fic
```

```
hamurabi% ls -l  
total 20  
drwx----- 3 gorsen lasso 4096 Oct 19 13:56 repl  
-rw----- 1 gorsen lasso 10240 Oct 19 13:56 arch.tar
```

```
hamurabi% gzip arch.tar  
hamurabi% ls -l  
total 20  
drwx----- 3 gorsen lasso 4096 Oct 19 13:56 repl  
-rw----- 1 gorsen lasso 204 Oct 19 13:56 arch.tar.gz
```

Manip. des Fichiers

Commandes

- Filtres
 - **sort aaa** : Affiche le contenu du fichier **aaa** en triant les lignes par ordre alphabétique
 - **wc aaa** : Compte le nombre de lignes, de mots et de lettres contenus dans le fichier **aaa**
- Redirection d'E/S
 - **wc aaa > compte.txt** : Redirige la sortie de la commande **wc aaa** dans le fichier **compte.txt**
 - **cat aaa | more** : Redirige la sortie de la commande **cat aaa** dans l'entrée de la commande **more**
 - **wc < aaa** : Utilise **aaa** comme entrée pour **wc**

Manip. des Fichiers

Exemples

```
hamurabi% wc texte
      3      9     44 texte

hamurabi% wc texte > compte.txt
hamurabi% ls -l
total 20
drwx-----   3 gorsen  lasso      4096 Oct 19 13:56 repl
-rw-----   1 gorsen  lasso       44 Oct 19 13:56 texte
-rw-----   1 gorsen  lasso      30 Oct 19 13:56 compte.txt

hamurabi% cat compte.txt
      3      9     44 texte

hamurabi% wc < texte
      3      9     44
```

Manip. des Fichiers

? et *

- ?
 - Remplace un caractère dans un nom de fichier donné en argument à une commande
 - **ls fic?** : Donne la liste des fichiers dont le nom a exactement 4 lettres et commence par fic
- *
 - Signifie zéro, ou une infinité de caractères dans un nom de fichier donné en argument à une commande
 - **ls fic*** : Donne la liste des fichiers dont le nom a 3 lettres ou plus et commence par fic

Propriétés et Droits d'Accès

Introduction à UNIX



Prop. et Droits d'Accès

Propriété

- Appartenance d'un fichier
 - Usager
 - Groupe
 - **id** : Donne l'identificateur d'utilisateur (uid), le groupe principal (gid) et les groupes secondaires qui sont associés à un usager

```
hamurabi% id gorsen
uid=16849(gorsen) gid=309(lasso) groups=309(lasso),806(gift0012)

hamurabi% ls -l
total 4
-rw----- 1 gorsen  lasso          0 Oct 19 14:03 fic1
drwx----- 2 gorsen  lasso       4096 Oct 19 14:07 repl
```

Prop. et Droits d'Accès

Champs

- Quatre champs
 - Type
 - d : Répertoire
 - - : Fichier
 - l : Lien
 - Permissions du propriétaire
 - Permissions du groupe
 - Permissions des autres

```
hamurabi% ls -l
total 4
-rw- --- --- 1 gorsen  lasso          0 Oct 19 14:03 fic1
d rwx --- --- 2 gorsen  lasso        4096 Oct 19 14:07 repl
```

Prop. et Droits d'Accès

Droits

- Trois champs pour chaque catégorie
 - Lecture
 - r pour signifier l'autorisation, - pour signifier l'interdiction
 - Ecriture
 - w pour signifier l'autorisation, - pour signifier l'interdiction
 - Exécution
 - x pour signifier l'autorisation, - pour signifier l'interdiction

```

hamurabi% ls -l
total 4
- rw- r-- r--   1 gorsen  lasso           0 Oct 19 14:03 fic1
d rwx r-x --x   2 gorsen  lasso        4096 Oct 19 14:07 repl
  
```

Prop. et Droits d'Accès

Exemples

- **-rw-r--r--**
 - Fichier
 - Lecture et écriture pour le propriétaire
 - Lecture seulement pour le groupe et les autres
- **drw-r-xr-x**
 - Répertoire
 - Lecture et écriture pour le propriétaire
 - Lecture et exécution pour le groupe et les autres
 - Attention : Le propriétaire n'a pas le droit d'entrer dans le répertoire en question

Prop. et Droits d'Accès

chmod

- **chmod** : Modification des droits d'accès sur un fichier ou un répertoire
 - Utilisation **chmod <qui> <+/-> <droit> fichier**
 - qui : **u** – usager (propriétaire)
g – groupe
o – autres
a – all (tout le monde : u, g et o)
 - +/- : ajout (+) ou retrait (-) de la permission
 - droit : **r** – droit de lecture
w – droit d'écriture
x – droit d'exécution
 - Les usagers et permissions peuvent être combinés

Manip. des Fichiers

Exemples

```
hamurabi% ls -l
Total 2
d rwx --- -wx    3 gorsen  lasso    4096 Oct 19 13:56 repl

hamurabi% chmod g+x repl
hamurabi% ls -l
Total 2
d rwx --x -wx    3 gorsen  lasso    4096 Oct 19 13:56 repl

hamurabi% chmod uo-w repl
hamurabi% ls -l
Total 2
d r-x --x --x    3 gorsen  lasso    4096 Oct 19 13:56 repl
```

Prop. et Droits d'Accès

chmod

- Utilisation des combinaisons de chiffres
 - **chmod <ABC> fic**
 - **A, B et C** – Chiffres codant les droits
 - **A** – Droits pour l'utilisateur (propriétaire)
 - **B** – Droits pour le groupe
 - **C** – Droits pour les autres
 - Principe
 - Un bit par permission :
 - r = bit 2
 - w = bit 1
 - x = bit 0
 - Valeur : 1 si la permission est présente, 0 sinon
 - Le nombre binaire obtenu est ensuite traduit en décimal

Manip. des Fichiers

Exemples

```
hamurabi% ls -l
Total 2
d rwx --- -wx    3 gorsen  lasso    4096 Oct 19 13:56 repl

hamurabi% chmod 710 repl
hamurabi% ls -l
Total 2
d rwx --x ---    3 gorsen  lasso    4096 Oct 19 13:56 repl

hamurabi% chmod 511 repl
hamurabi% ls -l
Total 2
d r-x --x --x    3 gorsen  lasso    4096 Oct 19 13:56 repl
```

Prop. et Droits d'Accès

chmod

- Utilisation des droits relatifs
 - **chmod <qui> = <ref> <+/-> <droits> fichier**
 - <ref> représente le champ auquel on fait référence pour établir les droits sur les autres champs (<qui>)
- Exemple

```
hamurabi% ls -l
Total 2
d rwx --- -wx    3 gorsen  lasso          4096 Oct 19 13:56 repl

hamurabi% chmod go=u-rw repl
hamurabi% ls -l
Total 2
d rwx --x --x    3 gorsen  lasso          4096 Oct 19 13:56 repl
```

Prop. et Droits d'Accès

chmod

- Exemples permissions / code
 - permissions : **rw- r-- r--**
 - Code : **644**
 - permissions : **--x--x---**
 - Code : **110**
- Récursivité
 - **chmod -R go=u-w rep1**
 - Donne au groupe et aux autres les mêmes droits qu'au propriétaire (mis à part le droit en écriture) sur le répertoire **rep1** et tous les sous répertoires et fichiers de ce répertoire

Aspects Réseau

Introduction à UNIX



Aspects Réseaux

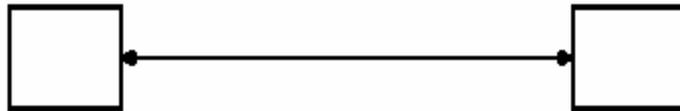
Applications

- Moyens d'échange
- Applications WEB – Commerce électronique
- Applications distribuées et client-serveur
- Accès à distance
 - Applications
 - Bases de données
- Systèmes industriels
- Télécommunications

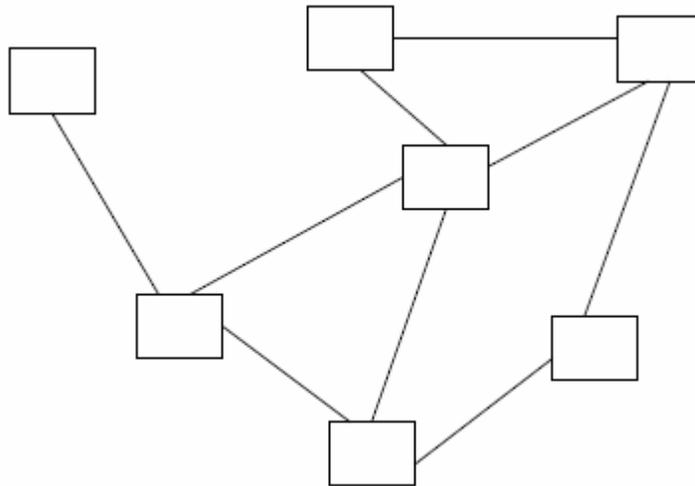
Aspects Réseau

Principe

- Communication de données



- Mise en réseau



Aspects Réseau

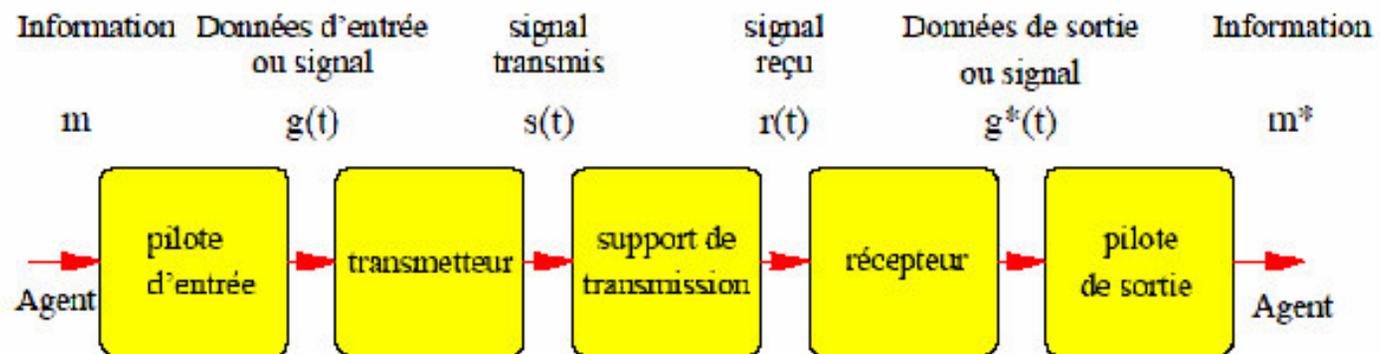
Couches

- Liaison
 - Media de communication (câbles réseau)
- Matériel
 - Contrôleur d'accès au media (carte réseau)
 - Adresse de la carte réseau
- Logiciel
 - Protocoles de communication
- Applications
 - Applications et programmes usagers

Aspects Réseau

Transmission

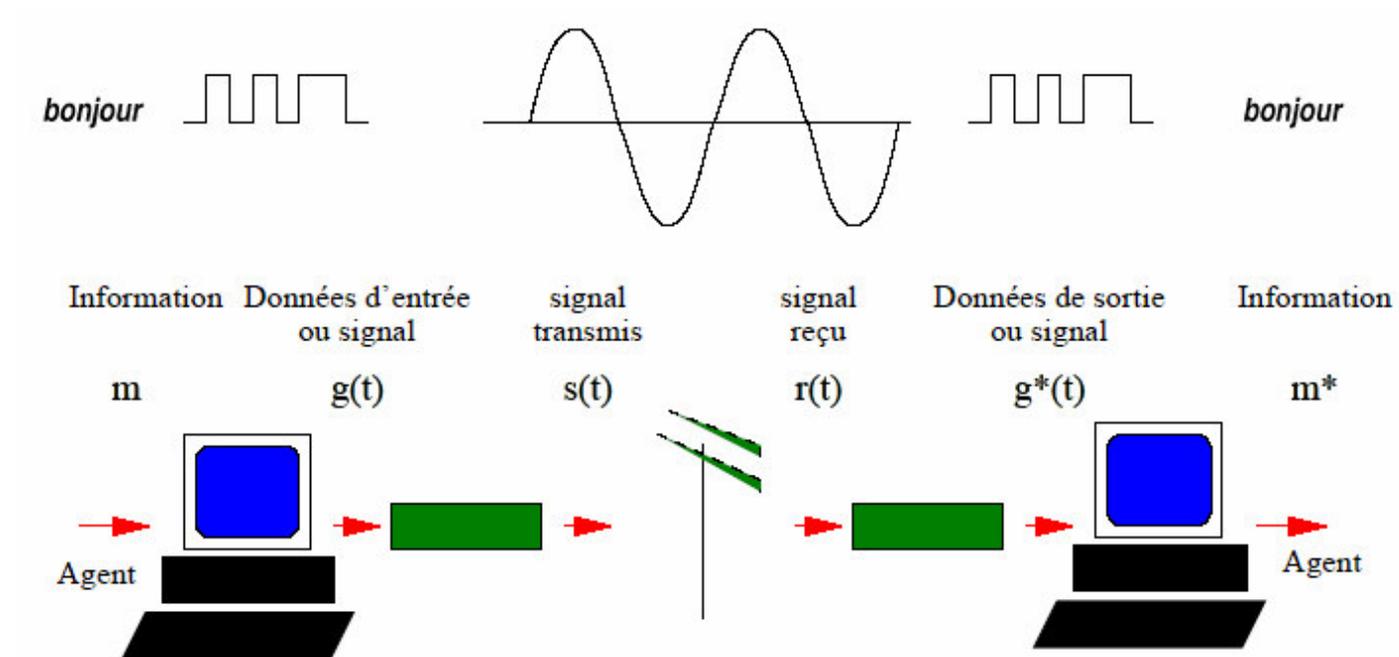
- Simple modèle de communication



Aspects Réseau

Transmission

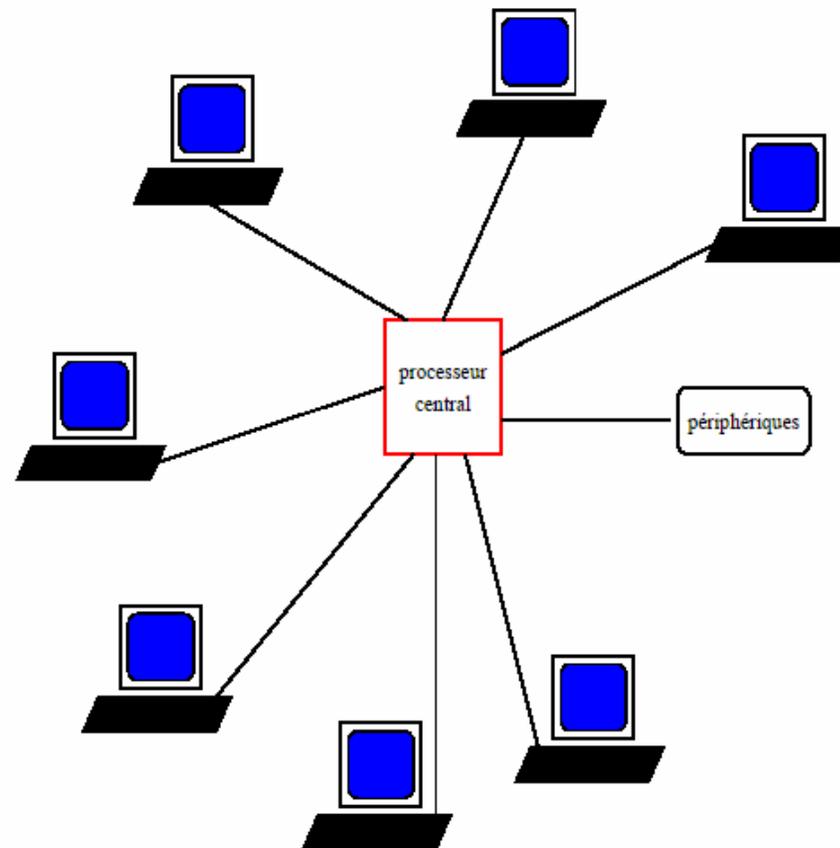
- Transmission par modem



Aspects Réseau

Types

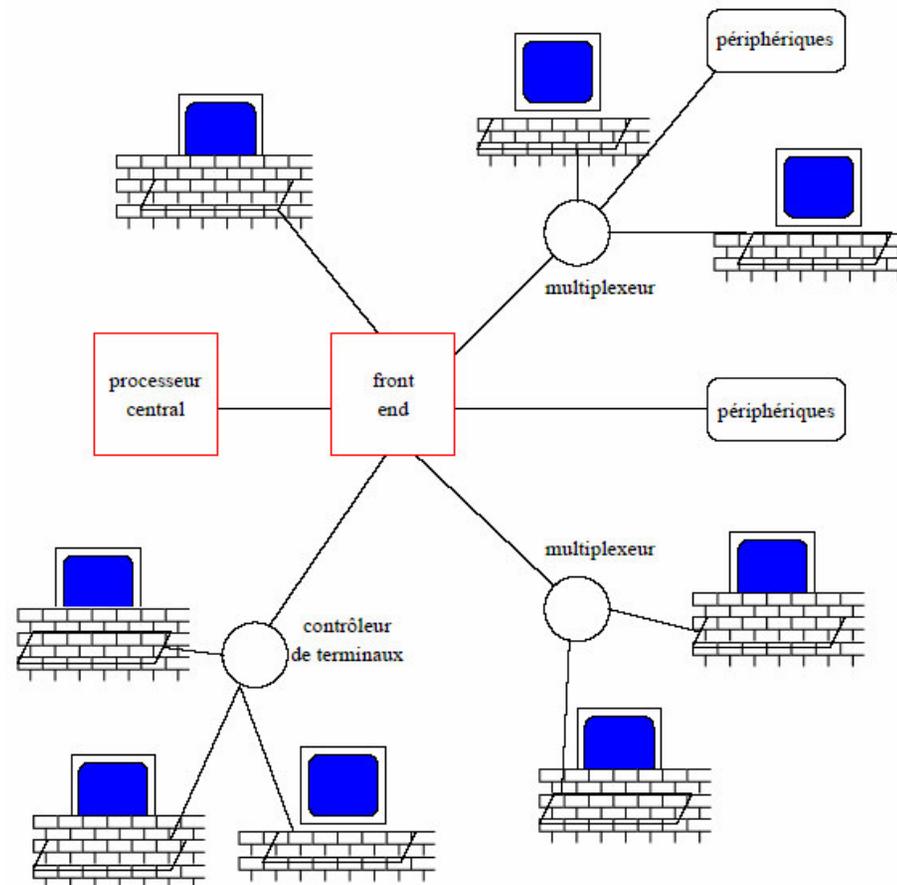
- Système avec unité centrale



Aspects Réseau

Types

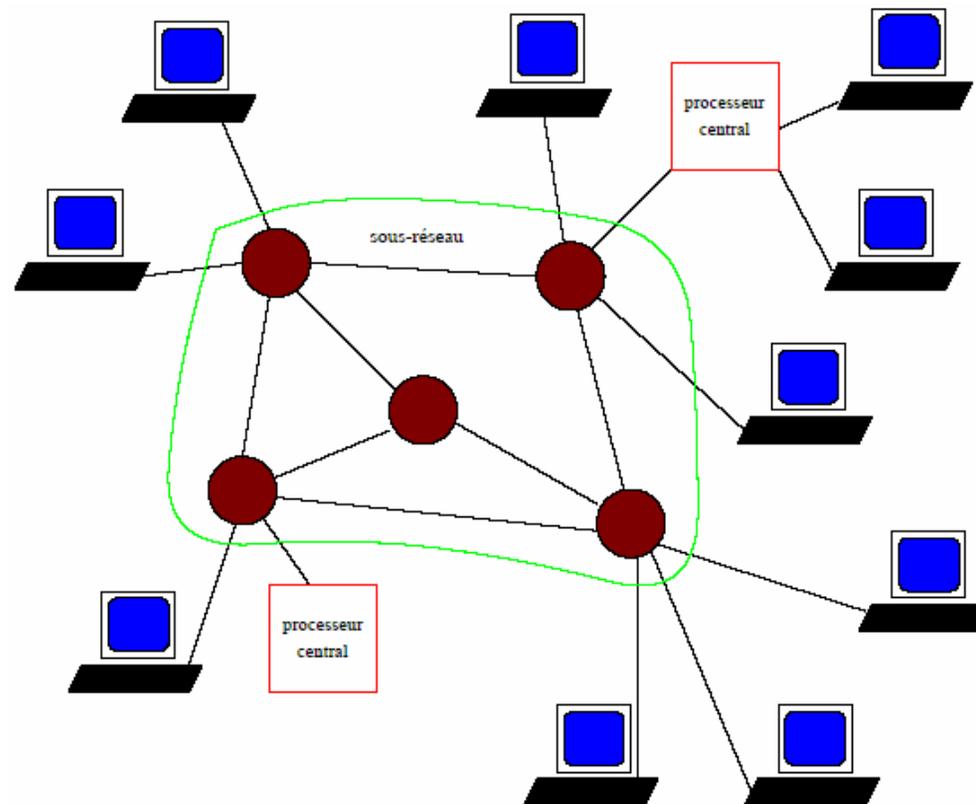
- Système avec contrôleurs de communication



Aspects Réseau

Types

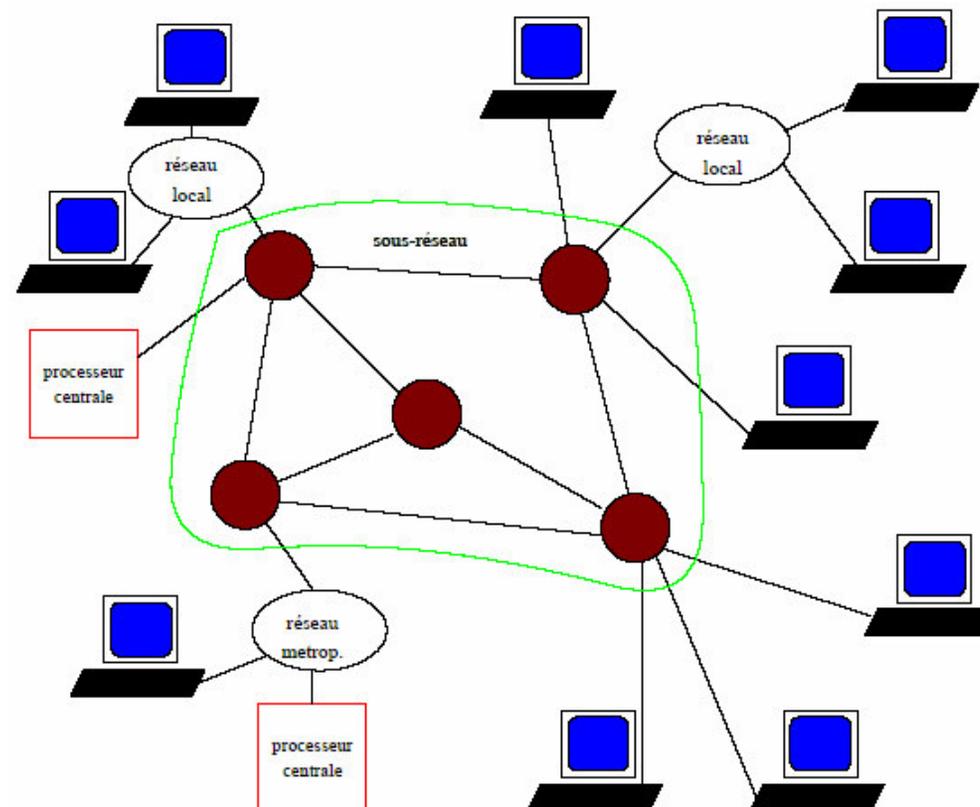
- Système central avec sous réseau de commutation



Aspects Réseau

Types

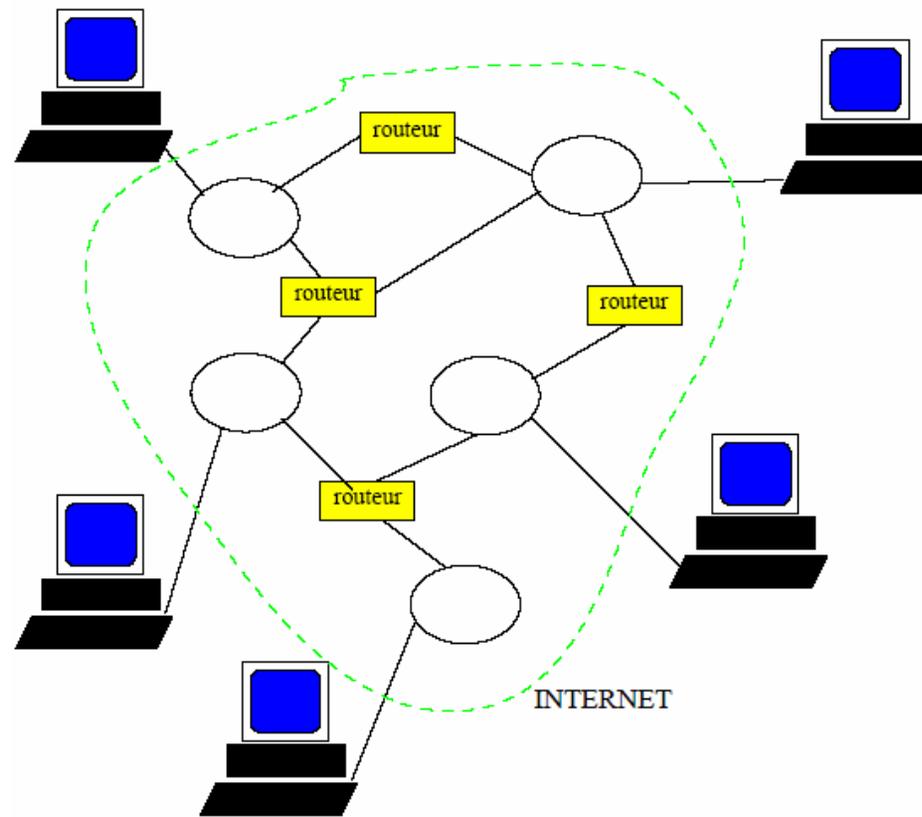
- Système hétérogène relié avec un réseau local



Aspects Réseau

Types

- Réseaux interconnectés



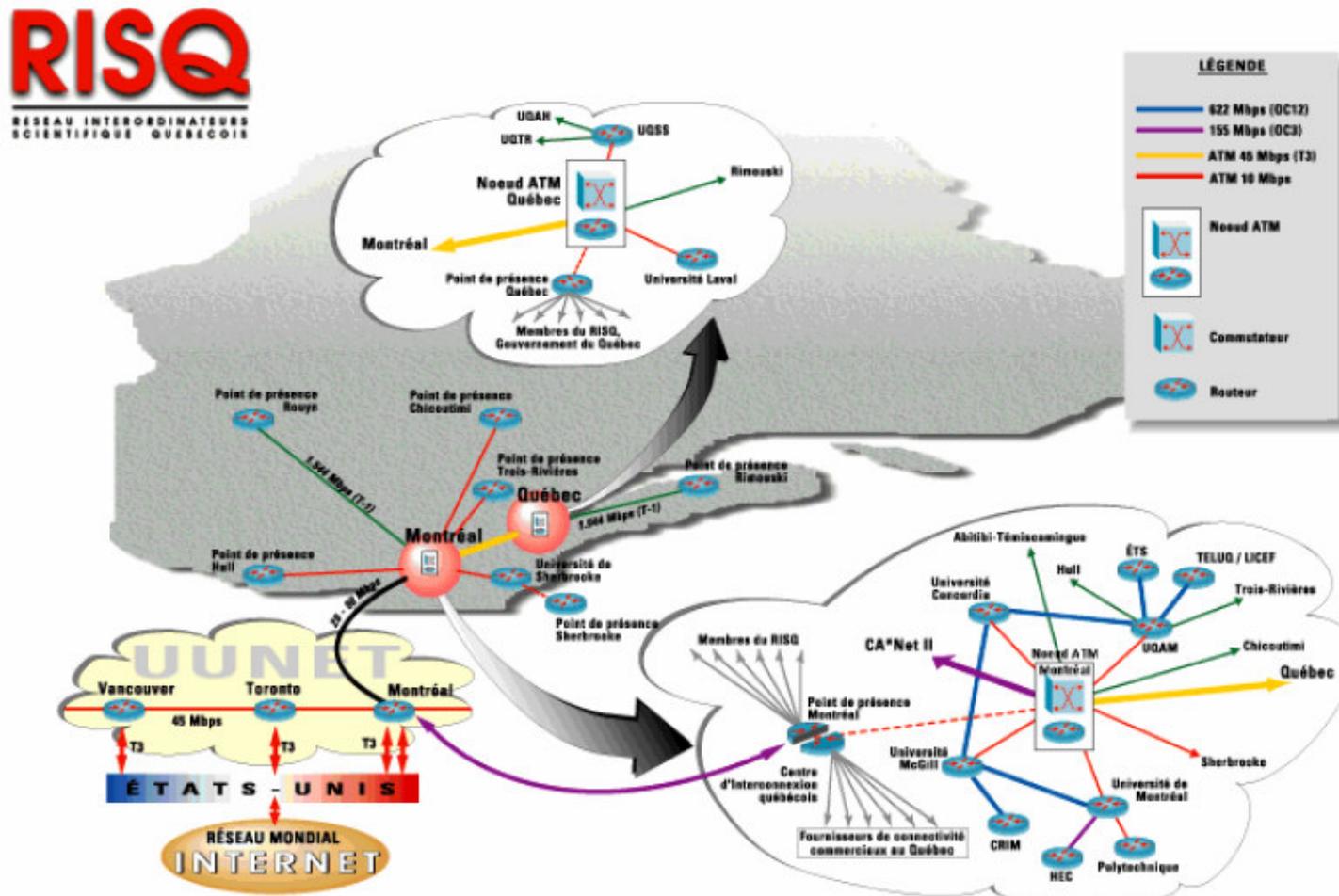
Aspects Réseau

Classification

- LAN – Local Area Network
 - Distance : 10 km
 - Performance : 10 Mbps – 1 Gbps
- MAN – Metropolitan Area Network
 - Distance : 100 km
 - Performance : 100 Mbps – 1 Gbps
- WAN – Wide Area Network
 - Distance : 1000 km
- GAN – Global Area Network
 - Interconnexion entre WAN et LAN

Aspects Réseaux

RISQ



Aspects Réseau

Identification

- Nom de machine
 - www.iro.umontreal.ca
- Adresse IP
 - 132.204.24.128
- Décomposition en domaines
- DNS – Domain Name Server
 - Base de données de correspondance noms/IP
- Commande UNIX : **nslookup**

Aspects Réseau

Identification

```
twister% nslookup www.iro.umontreal.ca
Server:   odin.iro.umontreal.ca
Address:  132.204.24.70

Name:     anex.IRO.UMontreal.CA
Address:  132.204.24.128
Aliases:  www.IRO.UMontreal.CA
```

- Nom de machine : anex.IRO.UMontreal.CA
- Alias : www.IRO.UMontreal.CA
- Adresse du DNS : 132.204.24.70
- Adresse IP : 132.204.24.128

Aspects Réseau

UNIX

- Sessions de travail à distance
 - telnet – session de travail non encryptée
 - ssh – session de travail encryptée

```
twister% telnet vor.iro.umontreal.ca
Trying 132.204.24.42...
Connected to vor.
Escape character is '^]'.
DIRO Linux 9.0 (2.4.20-20.9smp) 39
login: gorsen
Password:
Last login: Sun Oct 19 20:15:40 from hamurabi.IRO.UMontreal.CA
vor%
```

Aspects Réseau

UNIX

- Transfert de fichiers – ftp

```
twister% ftp vor.iro.umontreal.ca
Connected to vor.iro.umontreal.ca.
220 (vsFTPd 1.1.3)
Name (vor:gorsen): gorsen
331 Please specify the password.
Password:
230 Login successful. Have fun.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> get arch.tar.gz
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for arch.tar.gz (204 bytes).
226 File send OK.
local: arch.tar.gz remote: arch.tar.gz
204 bytes received in 0.0033 seconds (60.06 Kbytes/s)
```

Aspects Réseau

UNIX

- Email
 - Applications
 - Serveurs de courrier électronique

```
twister% telnet mail.iro.umontreal.ca 25
Trying 132.204.24.67...
Connected to mercure.iro.umontreal.ca.
Escape character is '^]'.
220 mercure.iro.umontreal.ca ESMTP Sendmail 8.12.9/8.12.9; Tue,
    21 Oct 2003 10:52:02 -0400
helo twister
250 mercure.iro.umontreal.ca Hello twister.iro.umontreal.ca
    [132.204.26.219], pleased to meet you
quit
221 2.0.0 mercure.iro.umontreal.ca closing connection
Connection to mercure.iro.umontreal.ca closed by foreign host.
```

Aspects Réseau

UNIX

- Temps de réponse d'une machine : ping

```
twister% ping hamurabi.iro.umontreal.ca
hamurabi.iro.umontreal.ca is alive
twister% ping -s hamurabi.
PING hamurabi: 56 data bytes
64 bytes from hamurabi.iro.umontreal.ca (132.204.26.198):
    icmp_seq=0. time=1. ms
64 bytes from hamurabi.iro.umontreal.ca (132.204.26.198):
    icmp_seq=1. time=0. ms
64 bytes from hamurabi.iro.umontreal.ca (132.204.26.198):
    icmp_seq=2. time=0. ms
64 bytes from hamurabi.iro.umontreal.ca (132.204.26.198):
    icmp_seq=3. time=0. ms
.
.
.
```

Gestion des Processus

Introduction à UNIX



Gestion des Processus Introduction

- Une tâche est effectuée par un processus
- Le lancement d'une nouvelle tâche engendre la création d'un processus
 - Commande tapée par l'utilisateur
 - Tâche lancée par un processus déjà existant
- UNIX est multitâche
 - Les tâches sont exécutées en parallèle
 - Chaque tâche reçoit l'usage du processeur durant un court laps de temps (env. 1/100 sec)
 - La distribution est gérée par un ordonnanceur

Gestion des Processus

Définitions

- Processus (job, tâche, programme)
 - Unité d'exécution
 - Toute action est exécutée par un processus
 - Caractéristiques
 - Identifié par un numéro unique : PID (Process ID)
 - Possède son propre environnement
 - Peut être dans différents états
- Thread (Fil)
 - Unité plus légère
 - Permet de traiter plusieurs sous tâches partageant le même espace mémoire

Gestion des Processus

Définitions

- Composition de l'image d'un processus
 - Code
 - Données associées (traitées par le code)
 - Fichiers ouverts et leur état
 - Répertoire courant
 - Informations du système sur le processus
 - Propriétaire
 - Terminal associé
 - Etat
 - Priorité
 - Etc.

Gestion des Processus

Création

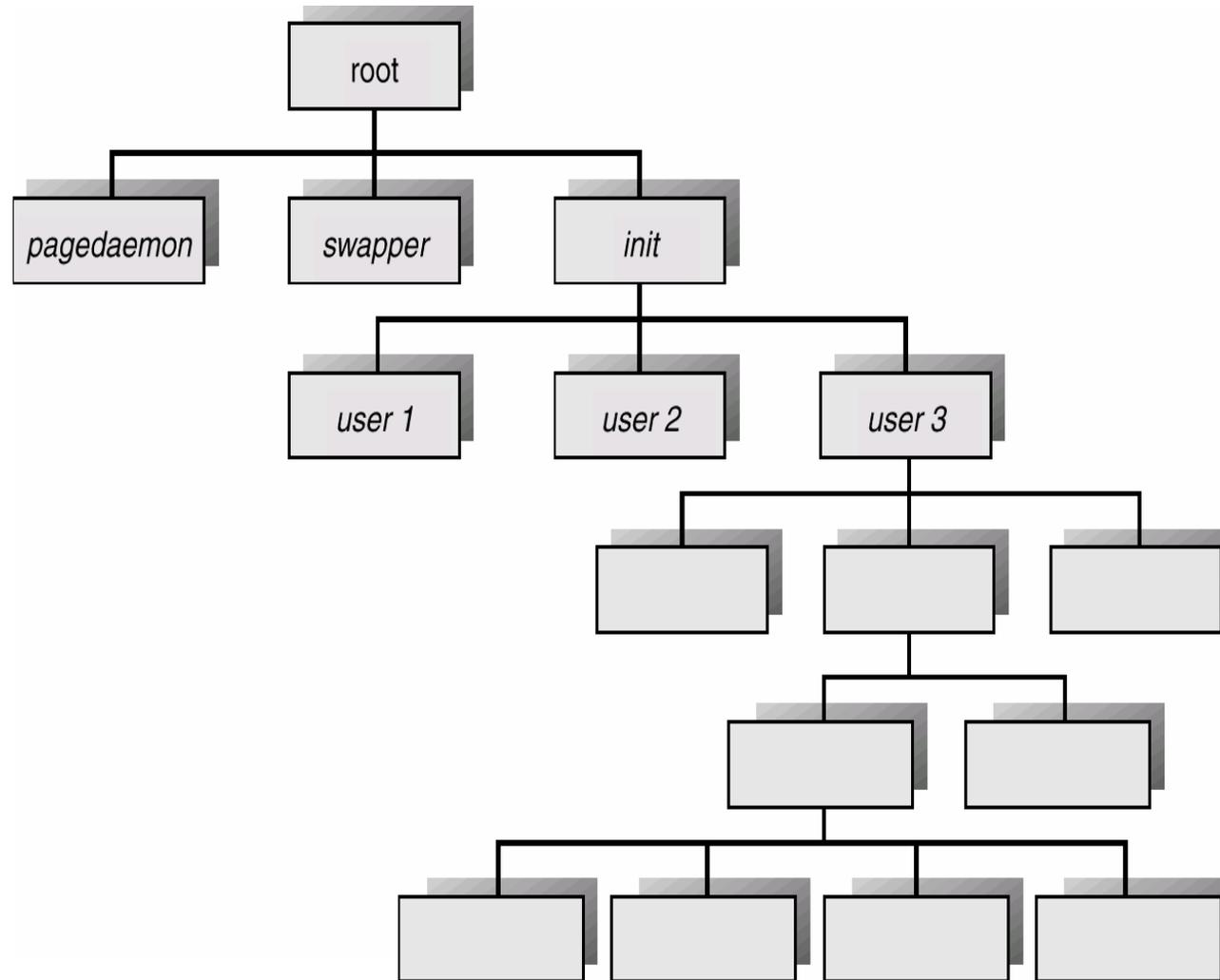
- Lancement d'une nouvelle tâche
- Création par un autre processus
 - Hiérarchie : Père, fils
 - Arbre de processus
- Question (différentes solutions suivant SE)
 - Un processus père doit continuer son exécution pendant l'existence des processus fils et doit les gérer de quelque façon
 - Père et fils sont indépendants et le père peut disparaître en laissant ses fils livrés à eux-mêmes

Gestion des Processus Terminaison

- Dernière instruction exécutée
- Réception d'un signal d'arrêt venant :
 - Du propriétaire
 - De l'administrateur
- Terminaison de l'exécution par son père
 - Différentes raisons possibles
 - Le fils a épuisé ses ressources
 - Le fils n'est plus requis
 - Etc.
- Note : Un fils peut tuer son père

Gestion des Processus

Arbre



Gestion des Processus

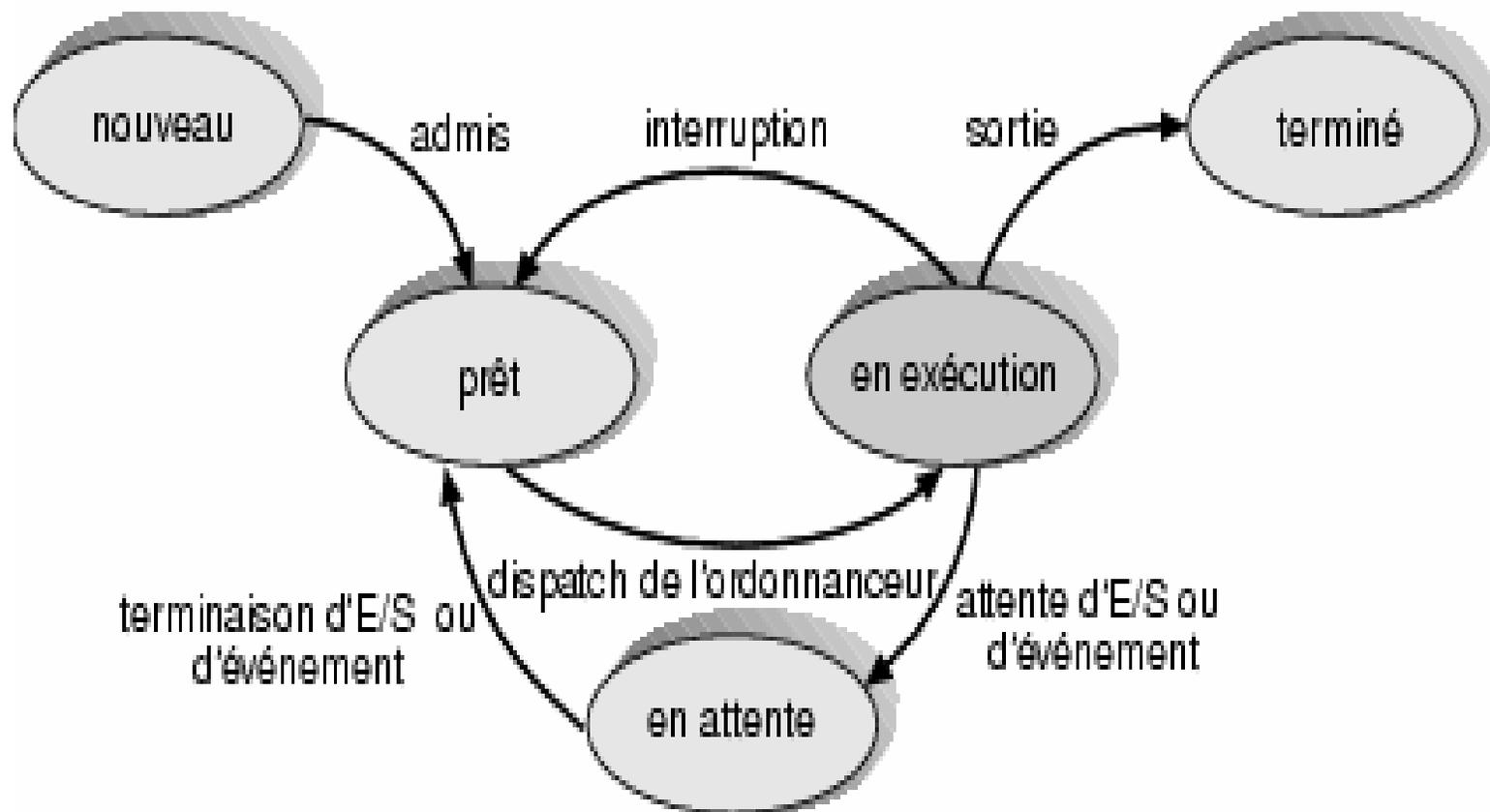
Etats

- Nouveau / New
 - Le processus vient d'être créé
- Exécutant / Running
 - Le processus est en train d'être exécuté
- Attente / Waiting
 - Le processus est en train d'attendre un événement
- Prêt / Ready
 - Le processus est en attente d'être exécuté
- Terminé / Terminated
 - Fin d'exécution

Gestion des Processus

Etats

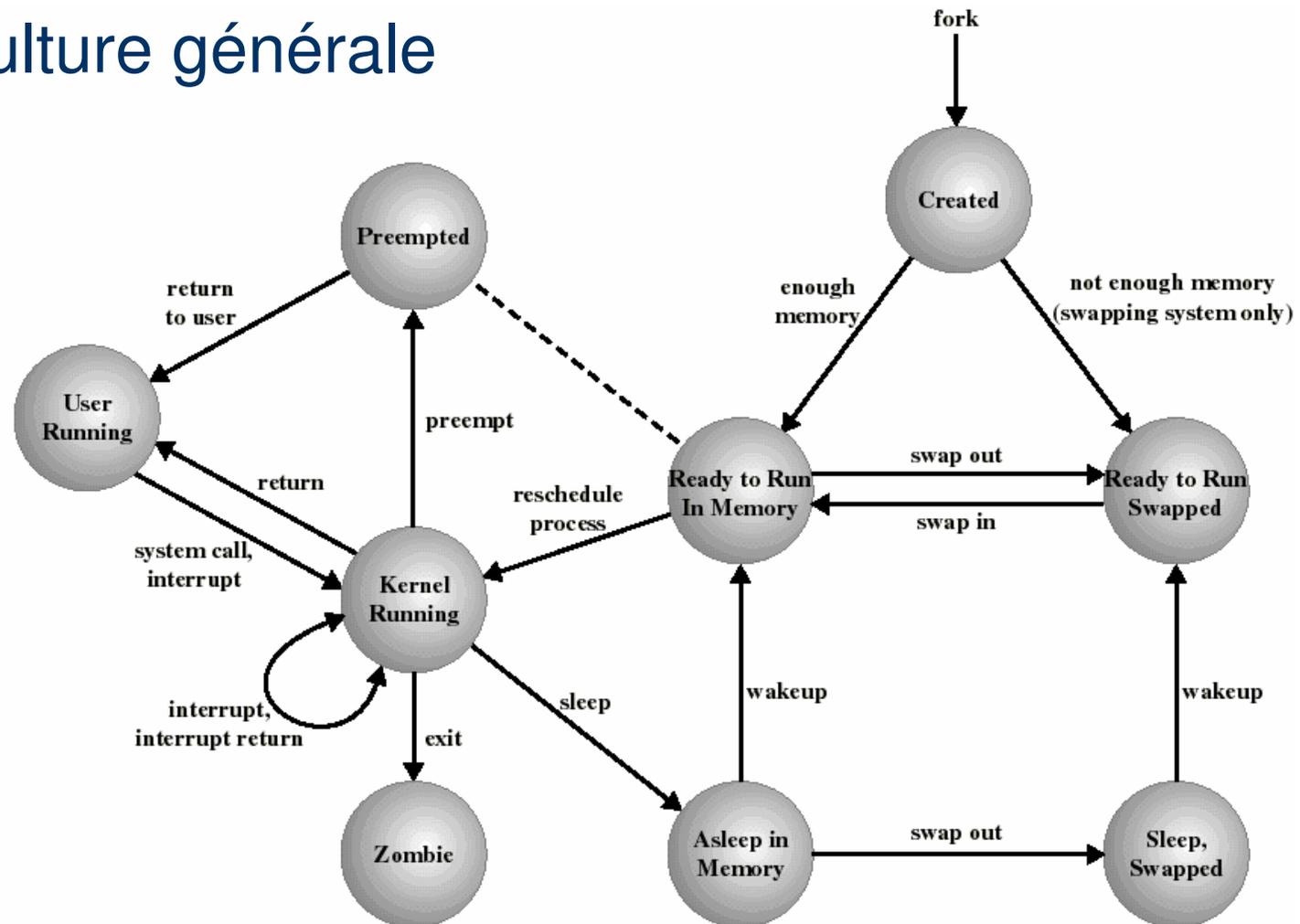
- Illustration



Gestion des Processus

Etats – SVr4

- Culture générale

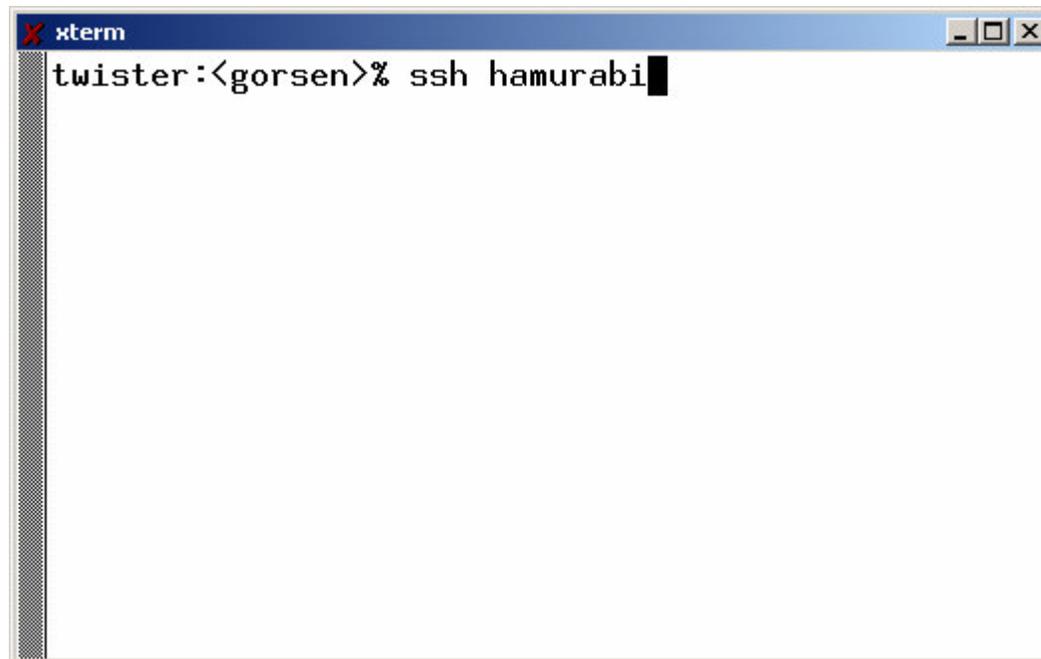


Gestion des Processus Commandes

- Lancement
 - Premier plan : **commande**
 - Arrière plan : **commande &**
 - Passage en premier plan : **fg**
- Premier plan
 - Arrêt : **Crtl-C**
 - Suspension : **Crtl-Z**
 - Redémarrage
 - Premier plan : **fg**
 - Arrière plan : **bg**

Gestion des Processus

Exemples



```
xterm  
twister:<gorsen>% ssh hamurabi
```

Gestion des Processus

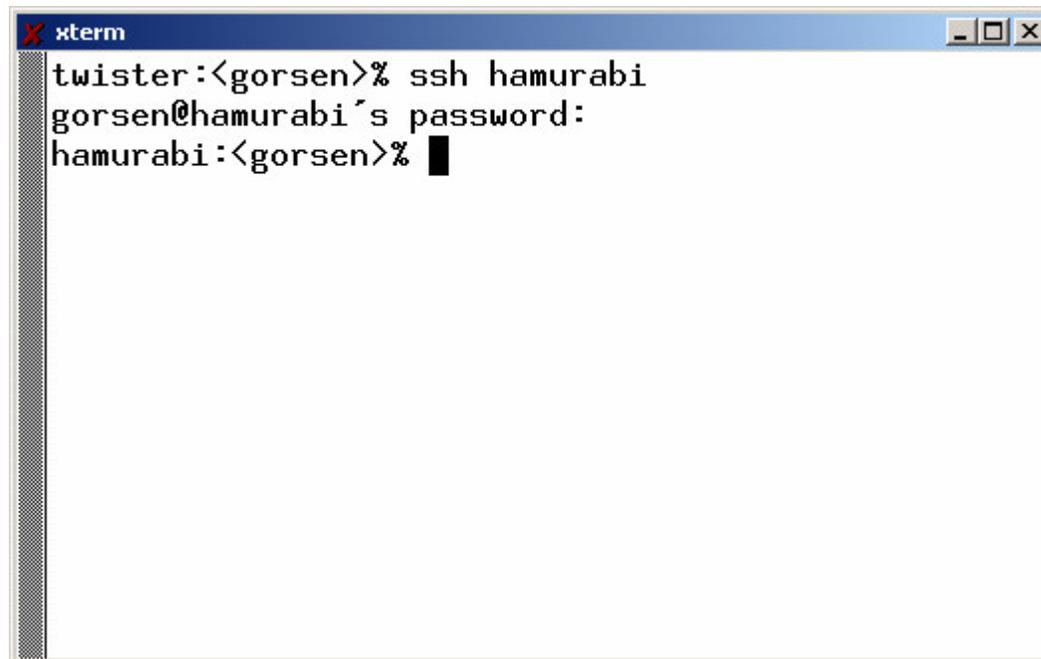
Exemples



```
xterm  
twister:<gorsen>% ssh hamurabi  
gorsen@hamurabi's password: █
```

Gestion des Processus

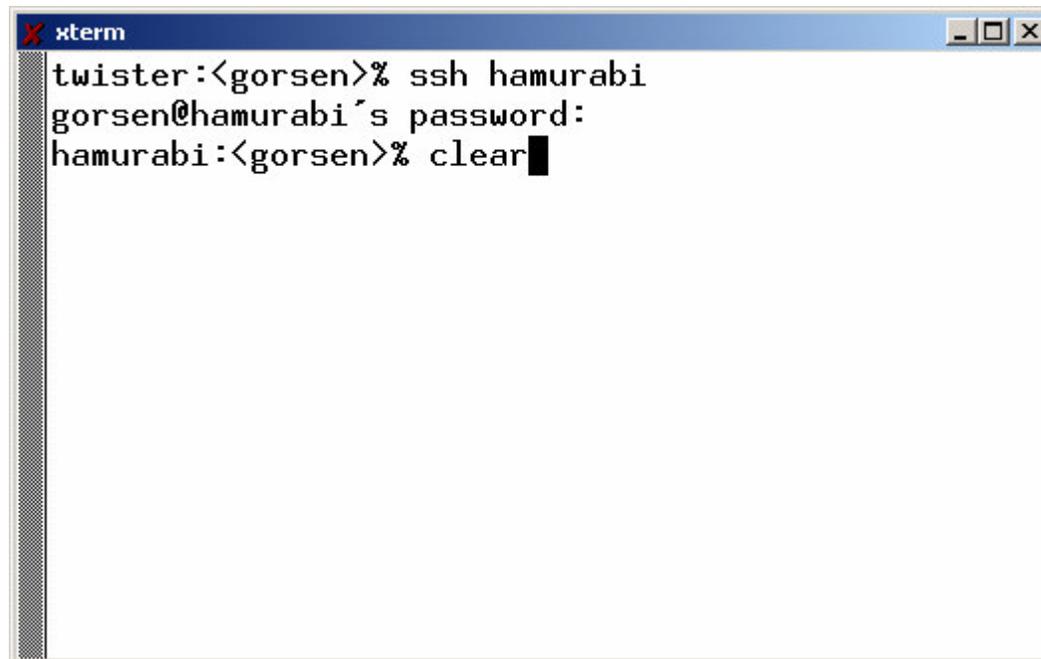
Exemples



```
xterm
twister:<gorsen>% ssh hamurabi
gorsen@hamurabi's password:
hamurabi:<gorsen>% █
```

Gestion des Processus

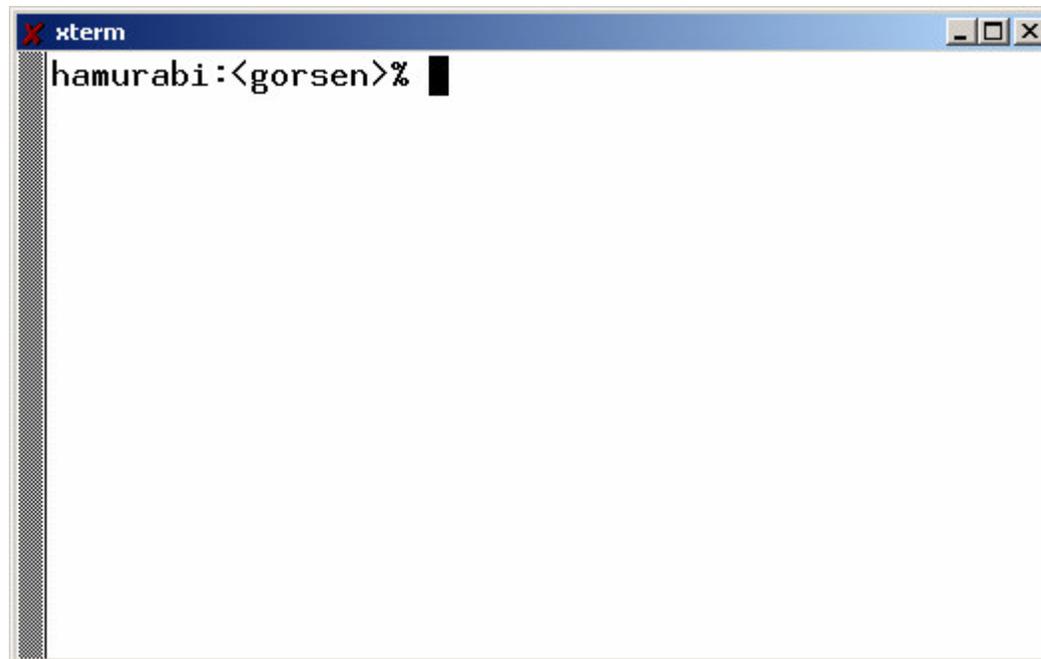
Exemples



```
xterm  
twister:<gorsen>% ssh hamurabi  
gorsen@hamurabi's password:  
hamurabi:<gorsen>% clear
```

Gestion des Processus

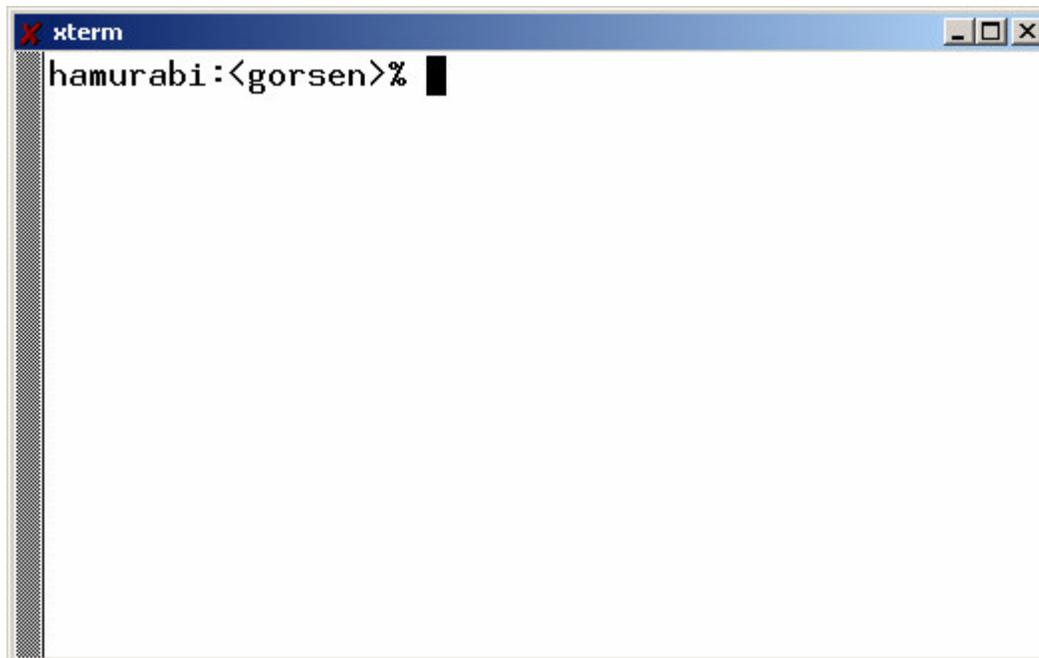
Exemples



A screenshot of an xterm window. The title bar reads "xterm" and includes standard window control buttons (minimize, maximize, close). The main content area shows a shell prompt: "hamurabi:<gorsen>%". A black cursor is positioned at the end of the prompt.

Gestion des Processus

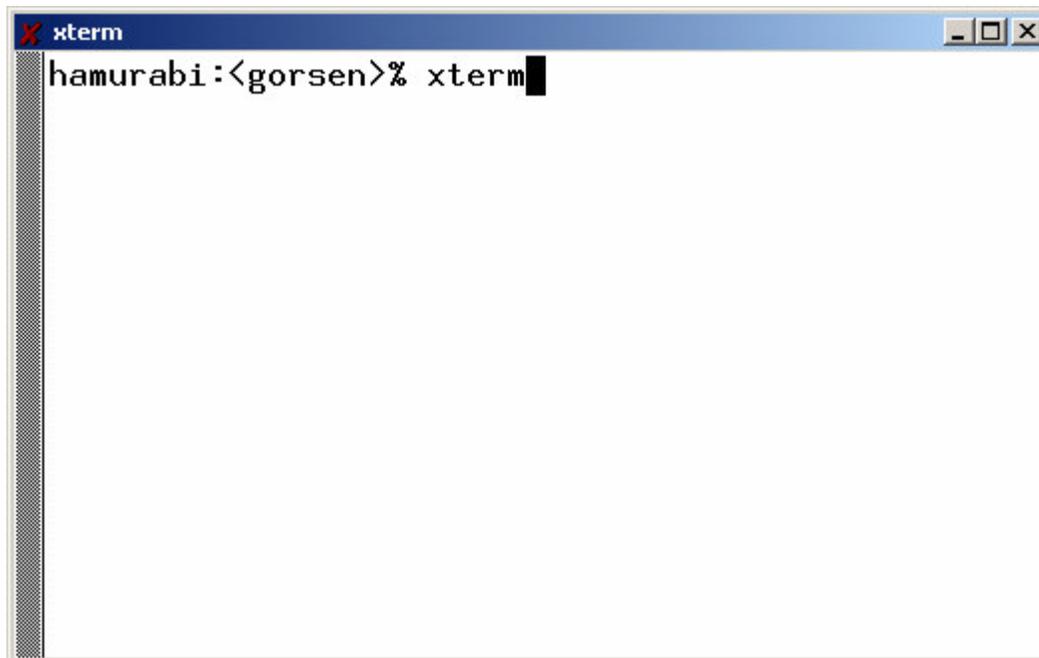
Exemples



```
xterm  
hamurabi:<gorsen>% █
```

Gestion des Processus

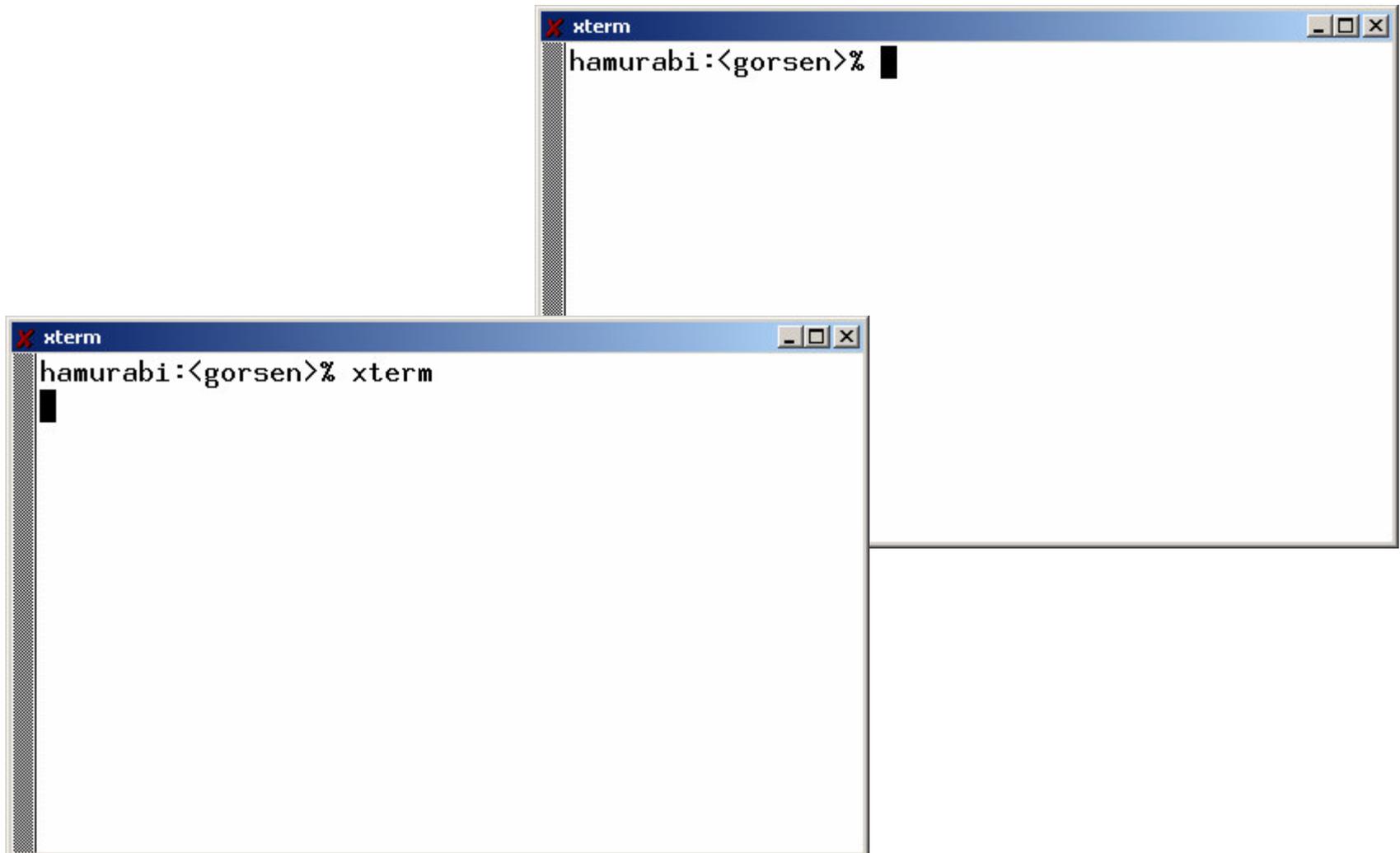
Exemples



```
xterm  
hamurabi:<gorsen>% xterm
```

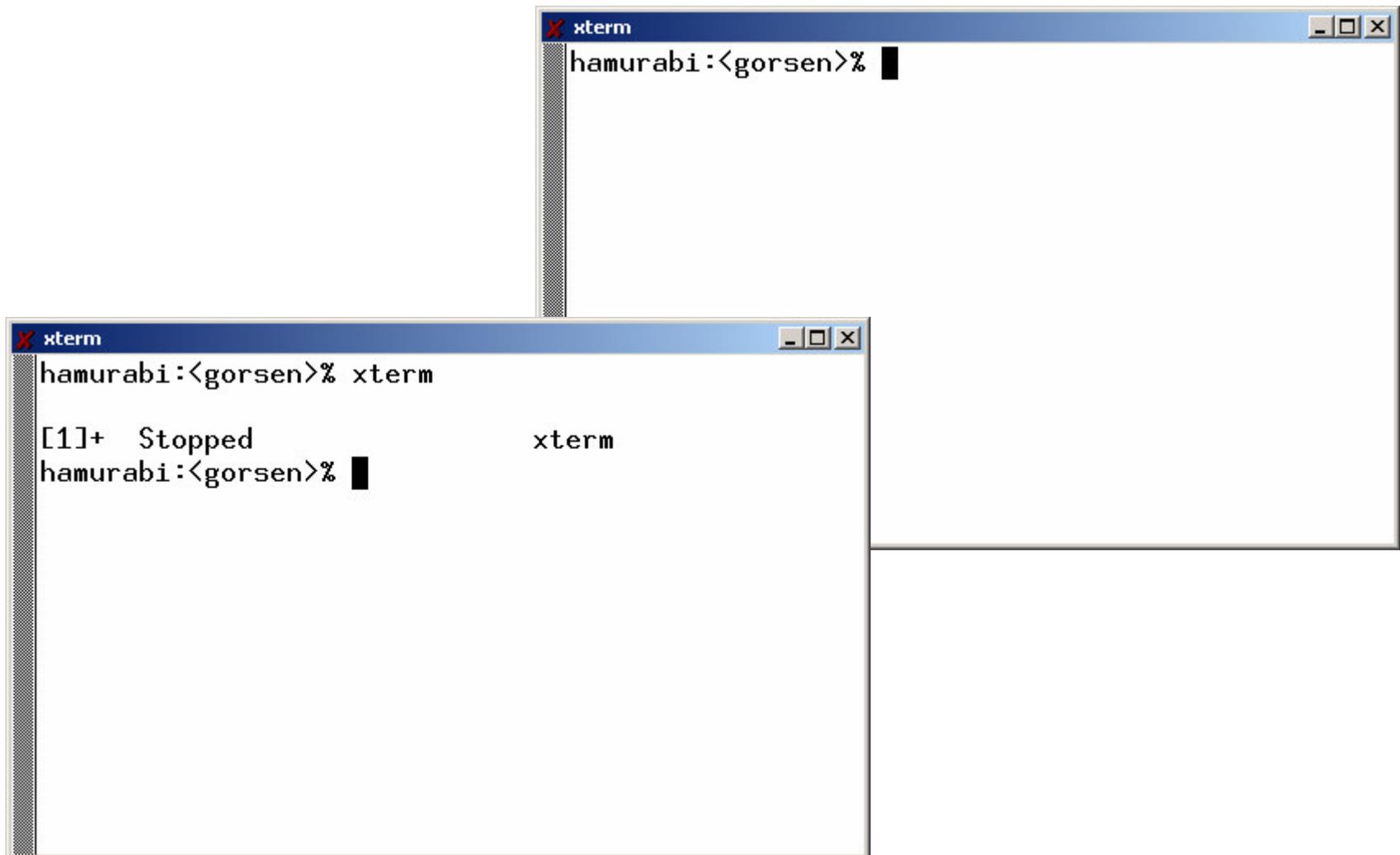
Gestion des Processus

Exemples



Gestion des Processus

Exemples



```
xterm
hamurabi:<gorsen>%

xterm
hamurabi:<gorsen>% xterm
[1]+  Stopped                  xterm
hamurabi:<gorsen>%
```

Gestion des Processus

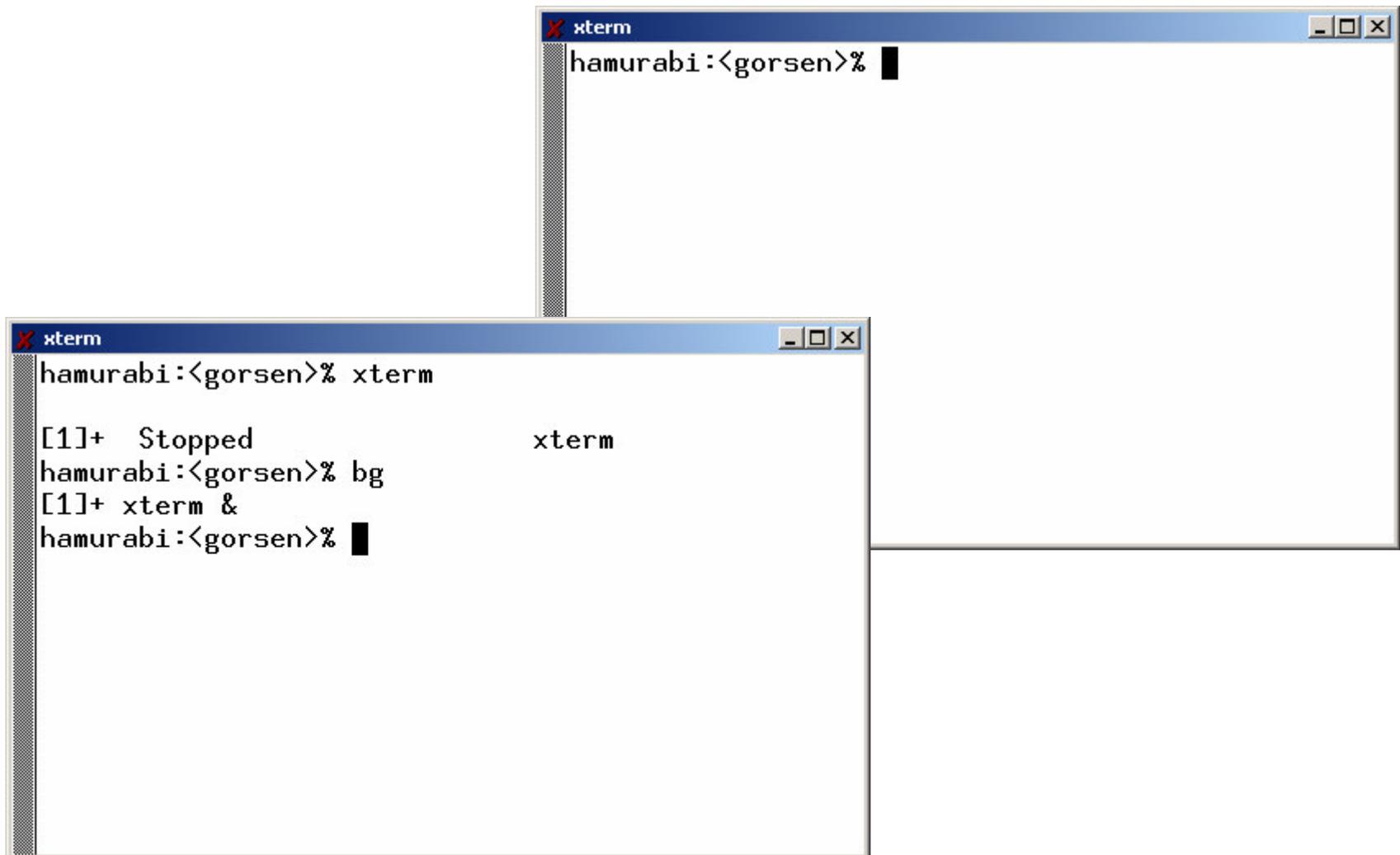
Exemples



```
hamurabi:<gorsen>%  
  
hamurabi:<gorsen>% xterm  
[1]+  Stopped                  xterm  
hamurabi:<gorsen>% bg
```

Gestion des Processus

Exemples



```
hamurabi:<gorsen>%  
  
hamurabi:<gorsen>% xterm  
[1]+  Stopped                  xterm  
hamurabi:<gorsen>% bg  
[1]+  xterm &  
hamurabi:<gorsen>%
```

Gestion des Processus

Exemples



```
xterm
hamurabi:<gorsen>%

xterm
hamurabi:<gorsen>% xterm
[1]+  Stopped                  xterm
hamurabi:<gorsen>% bg
[1]+  xterm &
hamurabi:<gorsen>% xterm &
```

Gestion des Processus

Exemples

PID de la 2^{ème} tache

Identificateur

```
xterm
hamurabi:<gorsen>%

xterm
hamurabi:<gorsen>%

xterm
hamurabi:<gorsen>% xterm
[1]+  Stopped
hamurabi:<gorsen>% bg
[1]+  xterm &
hamurabi:<gorsen>% xterm &
[2] 4027
hamurabi:<gorsen>%
```

xterm

Gestion des Processus Commandes

- Liste des processus : **ps [options]**
 - **ps -u <nom>** : Processus de l'utilisateur <nom>
 - **ps -a** : Processus attachés à un terminal
 - **ps -x** : Processus détachés de tout terminal
- Terminaison : **kill [options] <pid>**
 - **kill 2232** : Terminaison "propre" du processus
 - **kill -9 2232** : Terminaison non interceptable
- Priorité d'une tâche : **nice**
 - **nice -n <priorité> <commande>**
 - Plus haute : -20 et plus basse : 19

Gestion des Processus

Exemples

PID de la 2^{ème} tache

Identificateur

```
xterm
hamurabi:<gorsen>%

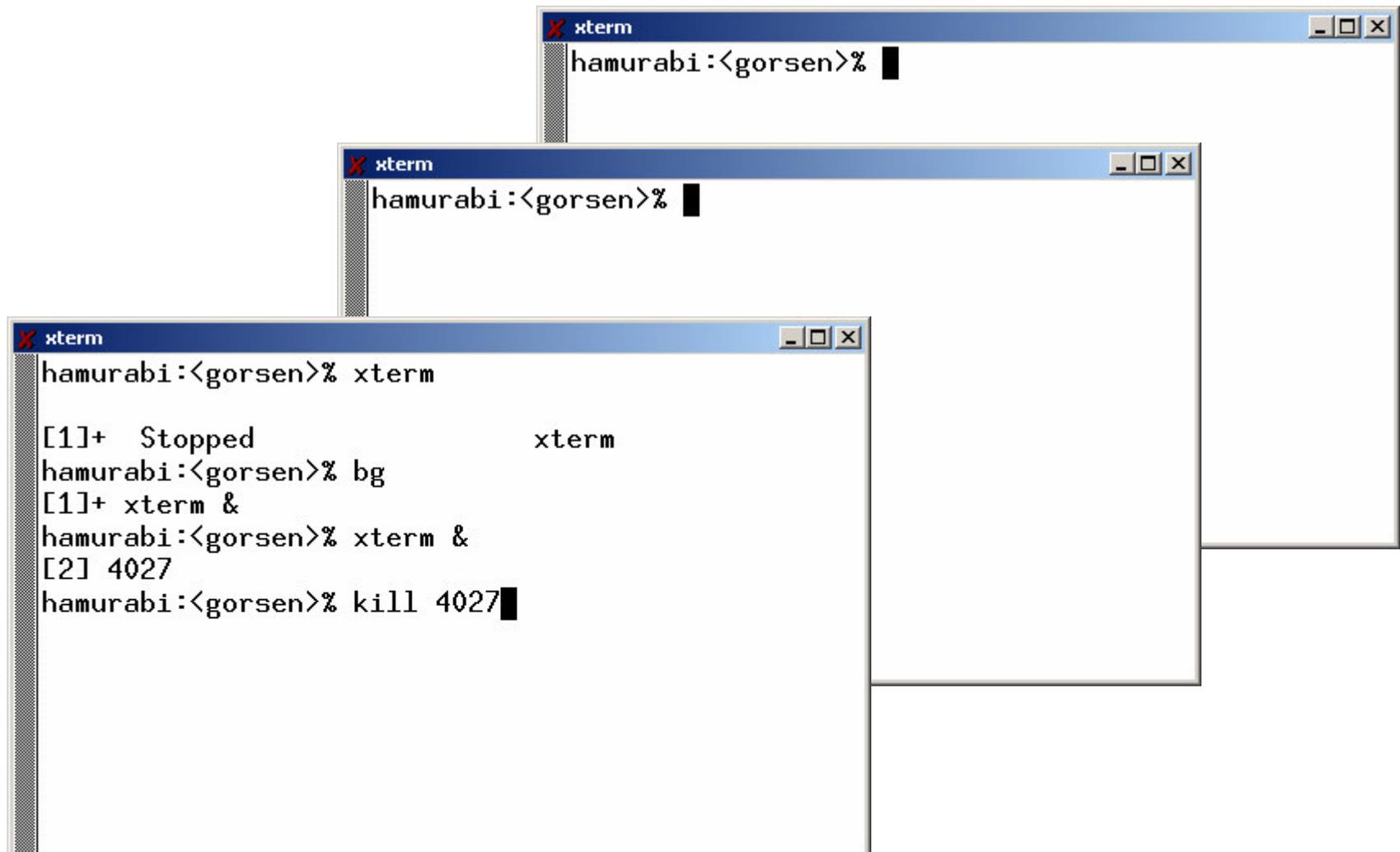
xterm
hamurabi:<gorsen>%

xterm
hamurabi:<gorsen>% xterm
[1]+  Stopped
hamurabi:<gorsen>% bg
[1]+  xterm &
hamurabi:<gorsen>% xterm &
[2] 4027
hamurabi:<gorsen>%
```

xterm

Gestion des Processus

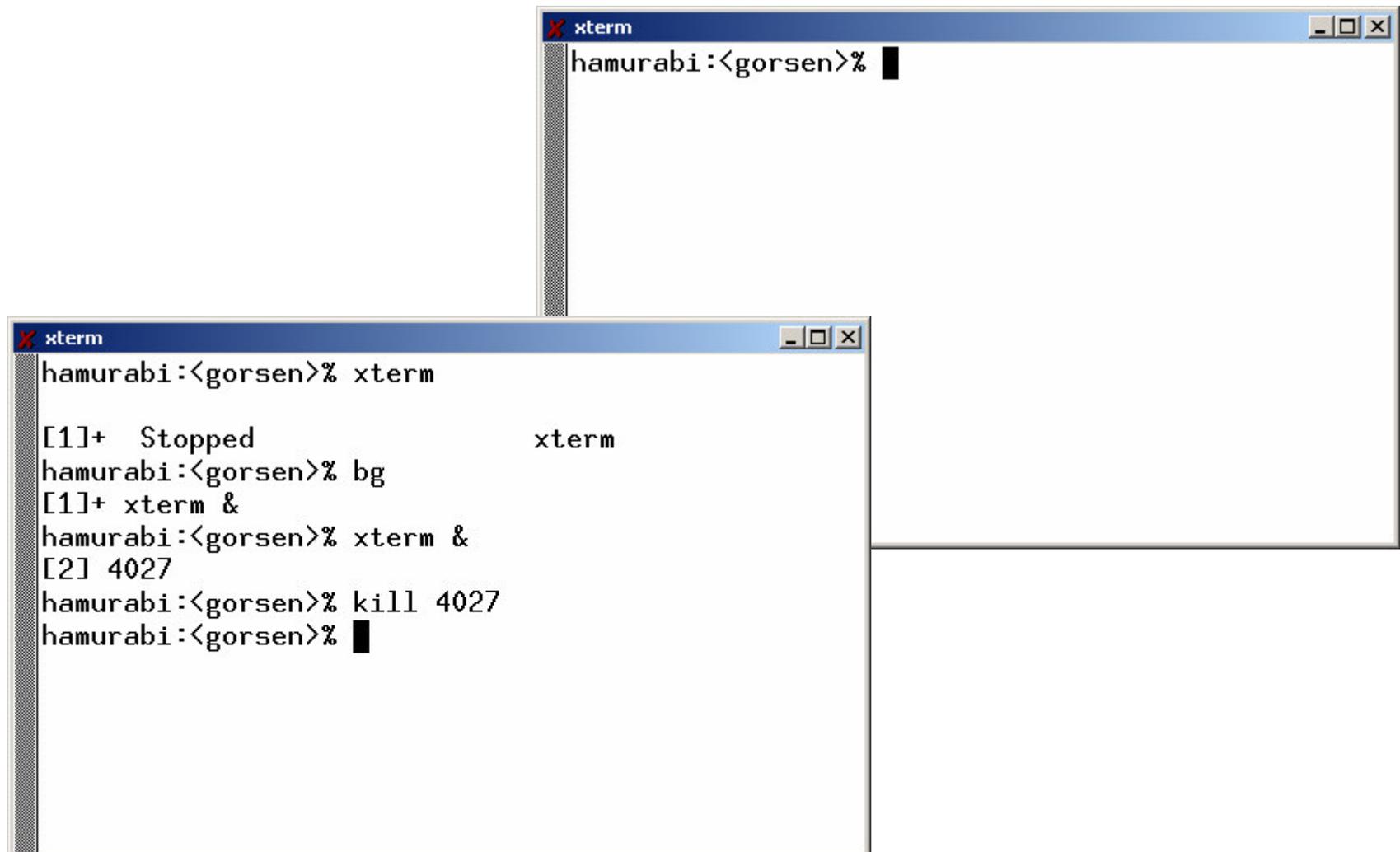
Exemples



```
hamurabi:<gorsen>% xterm
[1]+  Stopped                  xterm
hamurabi:<gorsen>% bg
[1]+  xterm &
hamurabi:<gorsen>% xterm &
[2] 4027
hamurabi:<gorsen>% kill 4027
```

Gestion des Processus

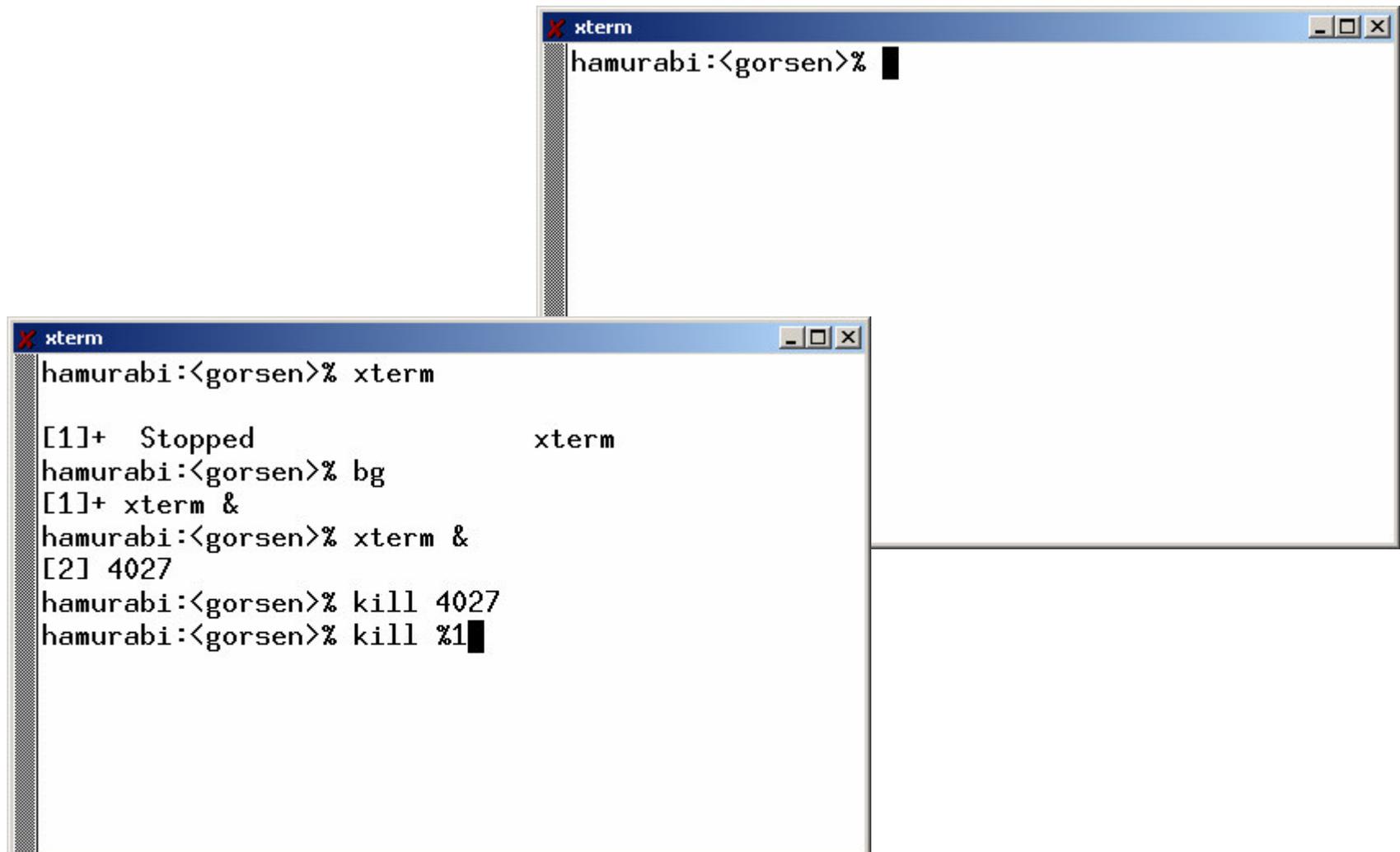
Exemples



```
hamurabi:<gorsen>%  
  
hamurabi:<gorsen>% xterm  
[1]+  Stopped                  xterm  
hamurabi:<gorsen>% bg  
[1]+  xterm &  
hamurabi:<gorsen>% xterm &  
[2] 4027  
hamurabi:<gorsen>% kill 4027  
hamurabi:<gorsen>%
```

Gestion des Processus

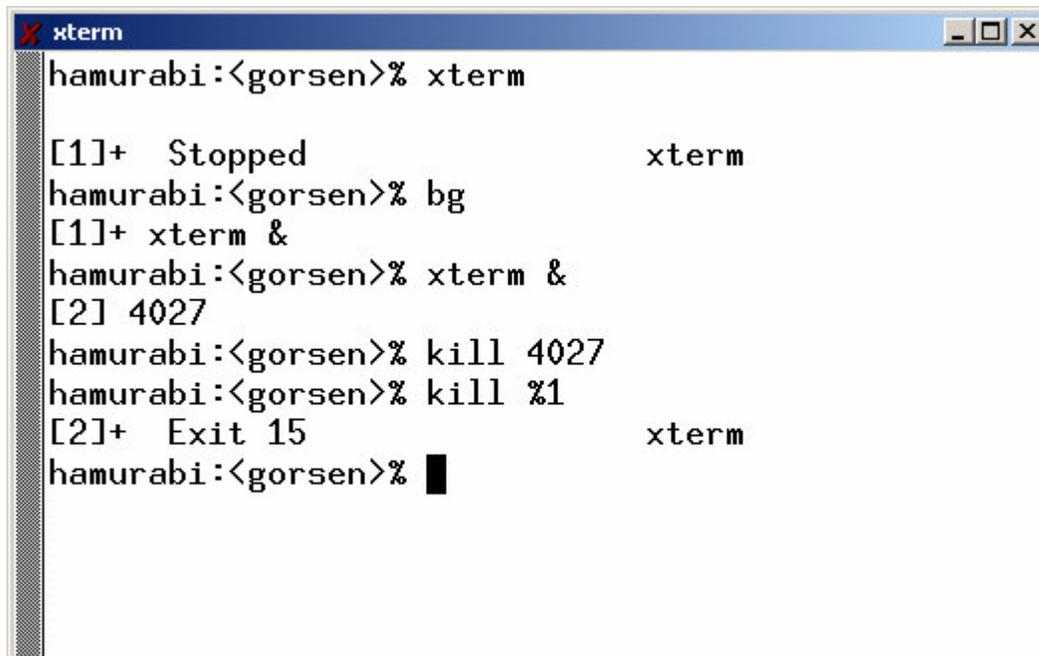
Exemples



```
hamurabi:<gorsen>%  
  
hamurabi:<gorsen>% xterm  
[1]+  Stopped                xterm  
hamurabi:<gorsen>% bg  
[1]+  xterm &  
hamurabi:<gorsen>% xterm &  
[2] 4027  
hamurabi:<gorsen>% kill 4027  
hamurabi:<gorsen>% kill %1
```

Gestion des Processus

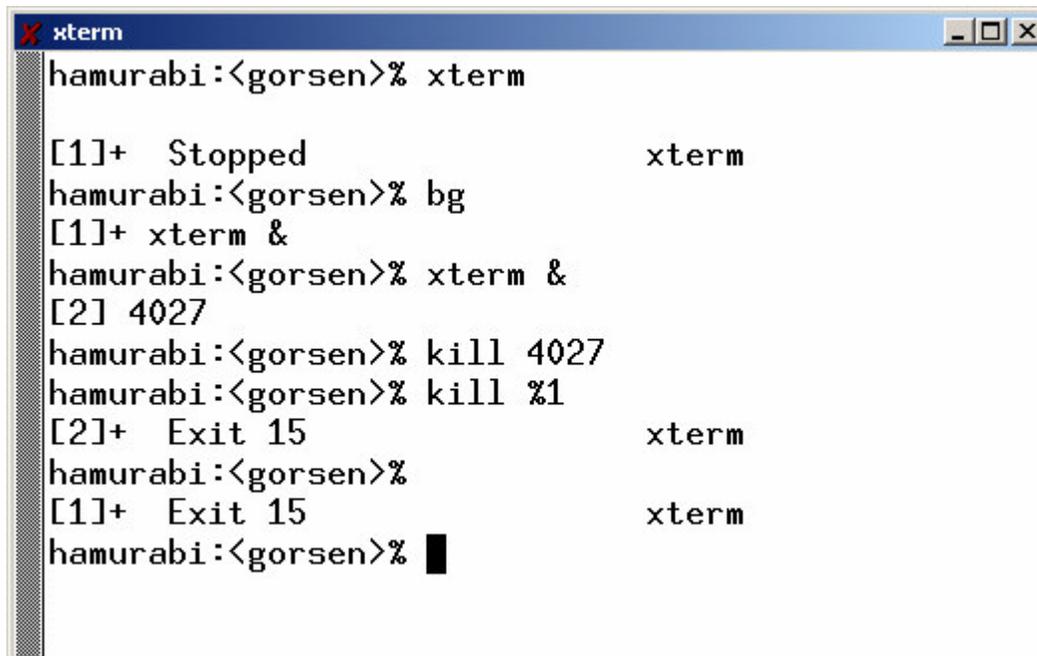
Exemples



```
xterm
hamurabi:<gorsen>% xterm
[1]+  Stopped                  xterm
hamurabi:<gorsen>% bg
[1]+  xterm &
hamurabi:<gorsen>% xterm &
[2] 4027
hamurabi:<gorsen>% kill 4027
hamurabi:<gorsen>% kill %1
[2]+  Exit 15                  xterm
hamurabi:<gorsen>% █
```

Gestion des Processus

Exemples



```
xterm
hamurabi:<gorsen>% xterm
[1]+  Stopped                  xterm
hamurabi:<gorsen>% bg
[1]+  xterm &
hamurabi:<gorsen>% xterm &
[2] 4027
hamurabi:<gorsen>% kill 4027
hamurabi:<gorsen>% kill %1
[2]+  Exit 15                  xterm
hamurabi:<gorsen>%
[1]+  Exit 15                  xterm
hamurabi:<gorsen>% █
```

Gestion des Processus

Exemples



A screenshot of an xterm window. The title bar reads "xterm" and includes standard window control buttons (minimize, maximize, close). The main area of the window displays a shell prompt: "hamurabi:<gorsen>%".

Gestion des Processus

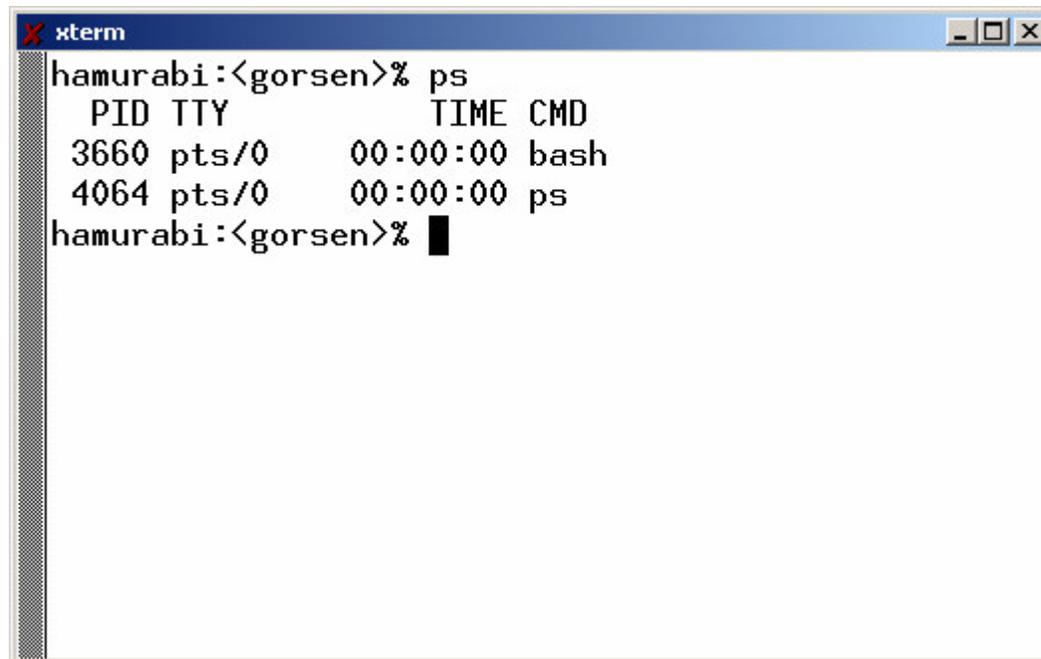
Exemples

A terminal window titled 'xterm' with standard window controls (minimize, maximize, close). The prompt is 'hamurabi:<gorsen>% ps' followed by a black cursor block.

```
xterm  
hamurabi:<gorsen>% ps
```

Gestion des Processus

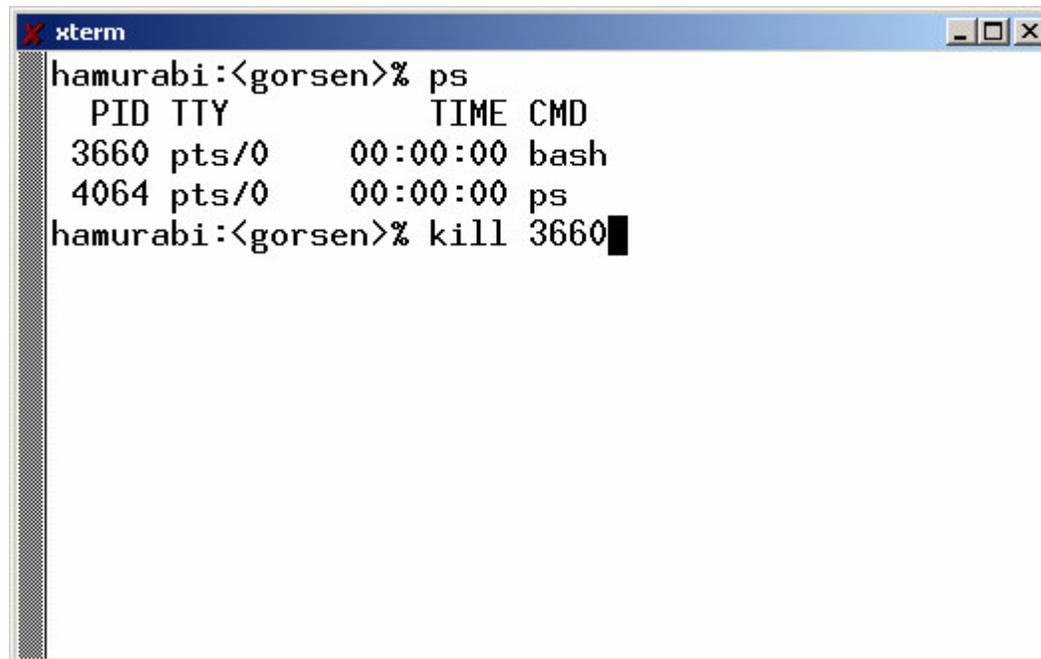
Exemples



```
xterm
hamurabi:<gorsen>% ps
  PID TTY          TIME CMD
 3660 pts/0    00:00:00 bash
 4064 pts/0    00:00:00 ps
hamurabi:<gorsen>% █
```

Gestion des Processus

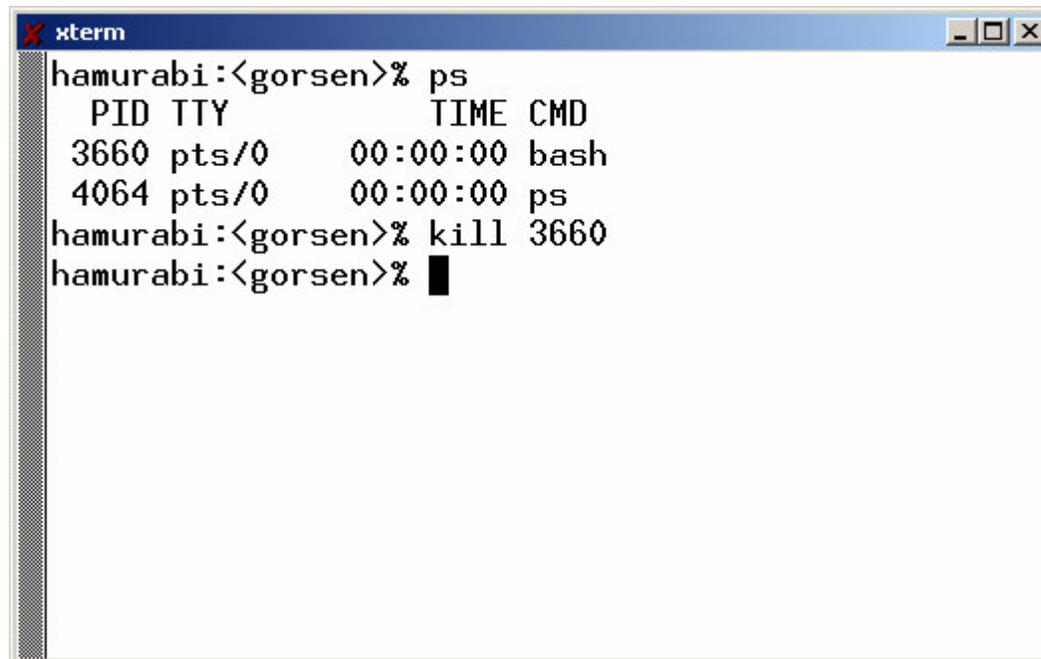
Exemples



```
xterm
hamurabi:<gorsen>% ps
  PID TTY          TIME CMD
 3660 pts/0        00:00:00 bash
 4064 pts/0        00:00:00 ps
hamurabi:<gorsen>% kill 3660
```

Gestion des Processus

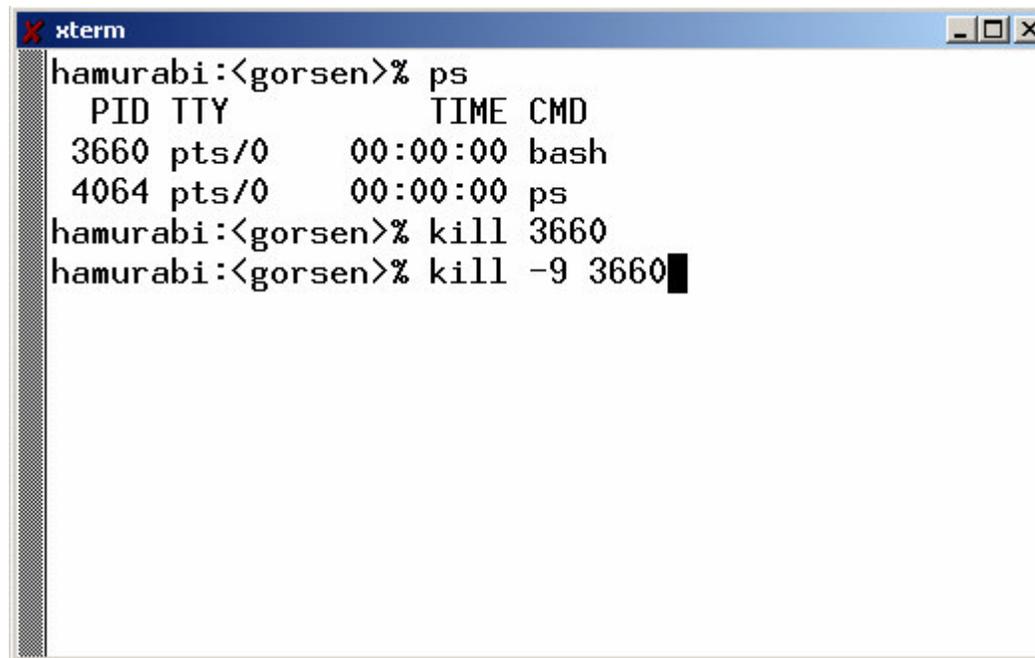
Exemples



```
xterm
hamurabi:<gorsen>% ps
  PID TTY          TIME CMD
 3660 pts/0    00:00:00 bash
 4064 pts/0    00:00:00 ps
hamurabi:<gorsen>% kill 3660
hamurabi:<gorsen>% █
```

Gestion des Processus

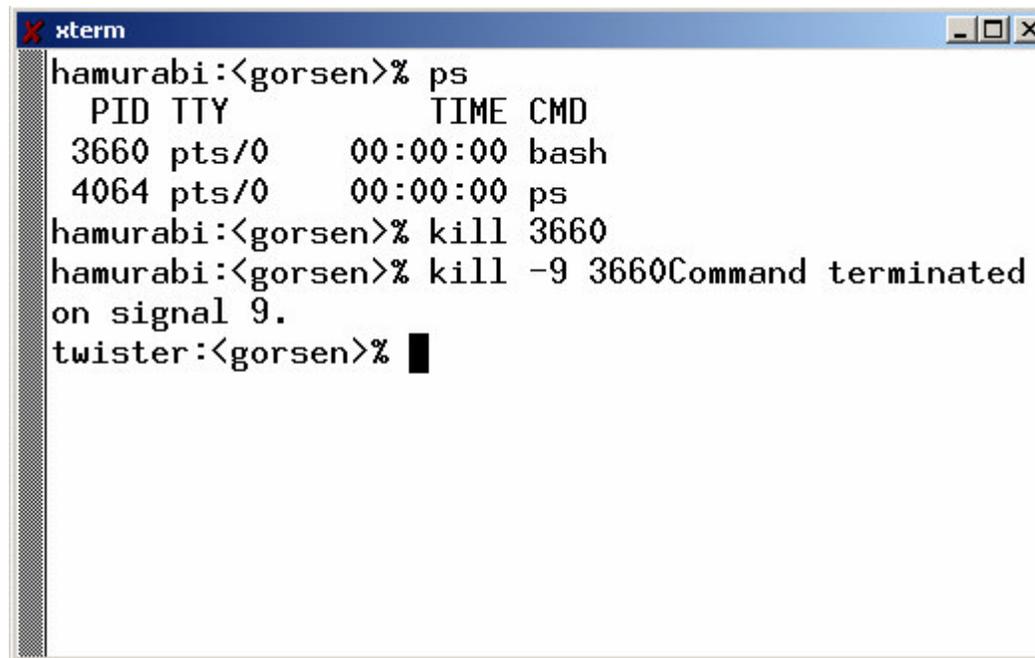
Exemples



```
xterm
hamurabi:<gorsen>% ps
  PID TTY          TIME CMD
 3660 pts/0    00:00:00 bash
 4064 pts/0    00:00:00 ps
hamurabi:<gorsen>% kill 3660
hamurabi:<gorsen>% kill -9 3660
```

Gestion des Processus

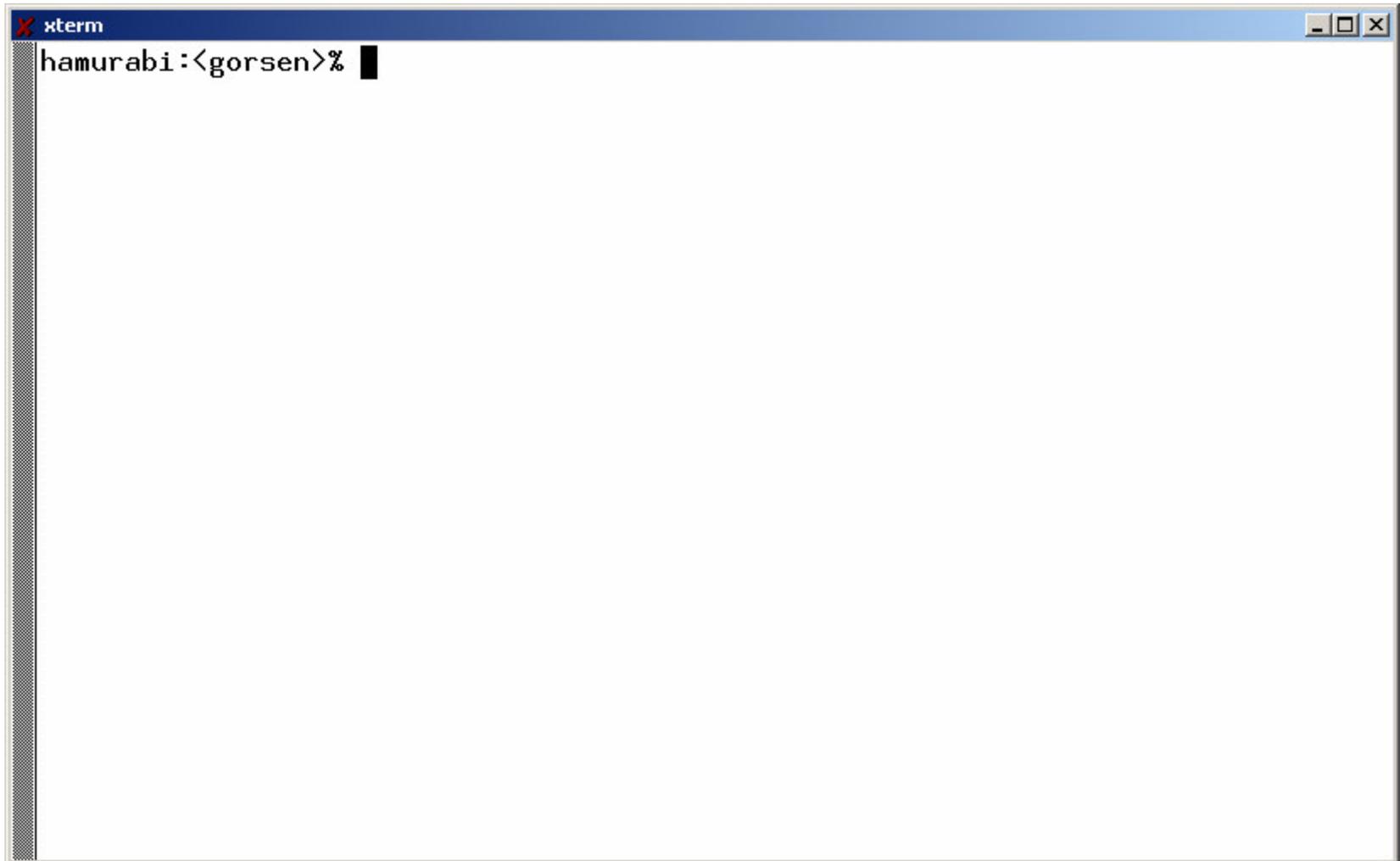
Exemples



```
xterm
hamurabi:<gorsen>% ps
  PID TTY          TIME CMD
 3660 pts/0        00:00:00 bash
 4064 pts/0        00:00:00 ps
hamurabi:<gorsen>% kill 3660
hamurabi:<gorsen>% kill -9 3660Command terminated
on signal 9.
twister:<gorsen>% █
```

Gestion des Processus

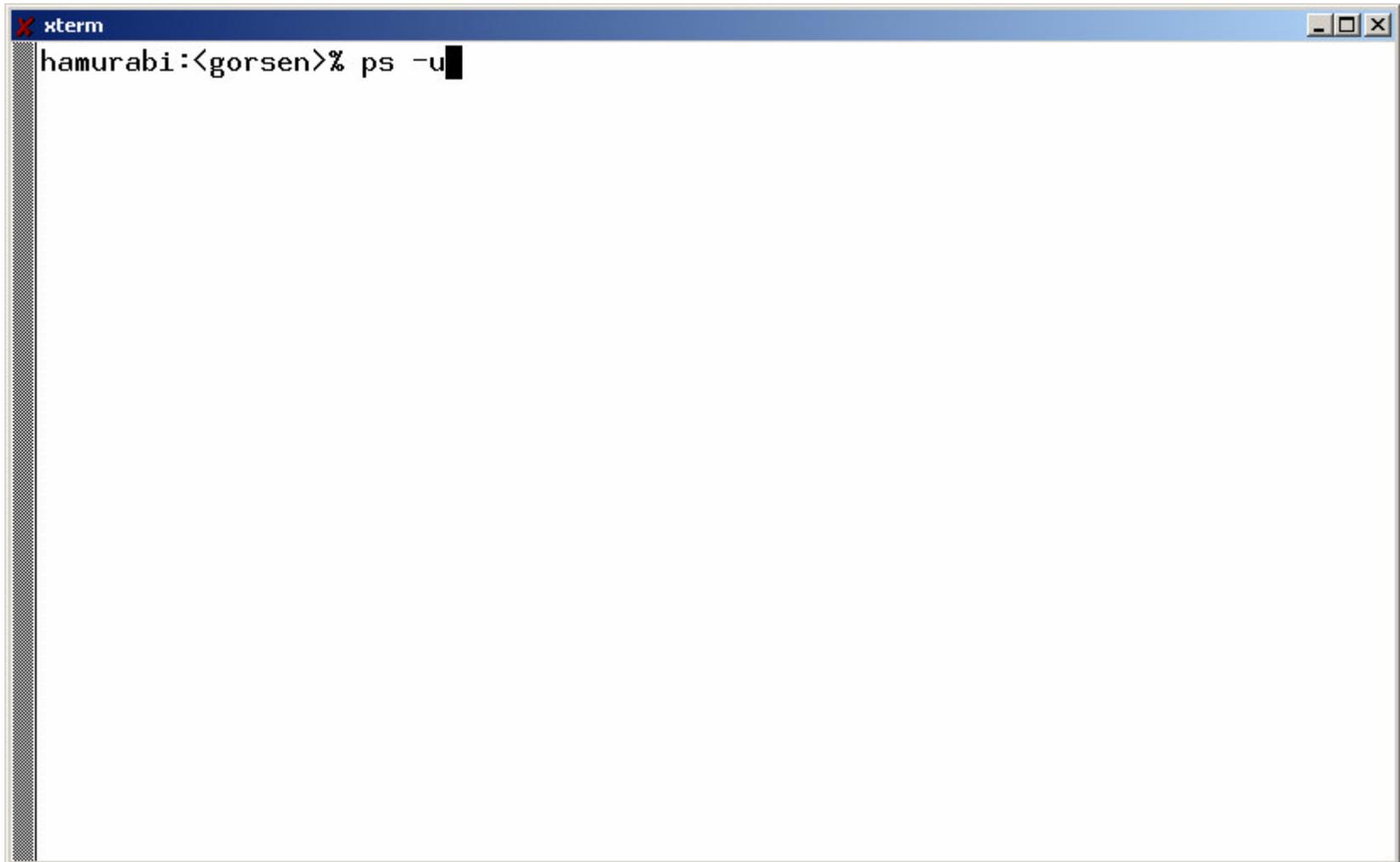
Exemples



```
xterm  
hamurabi :<gorsen>% █
```

Gestion des Processus

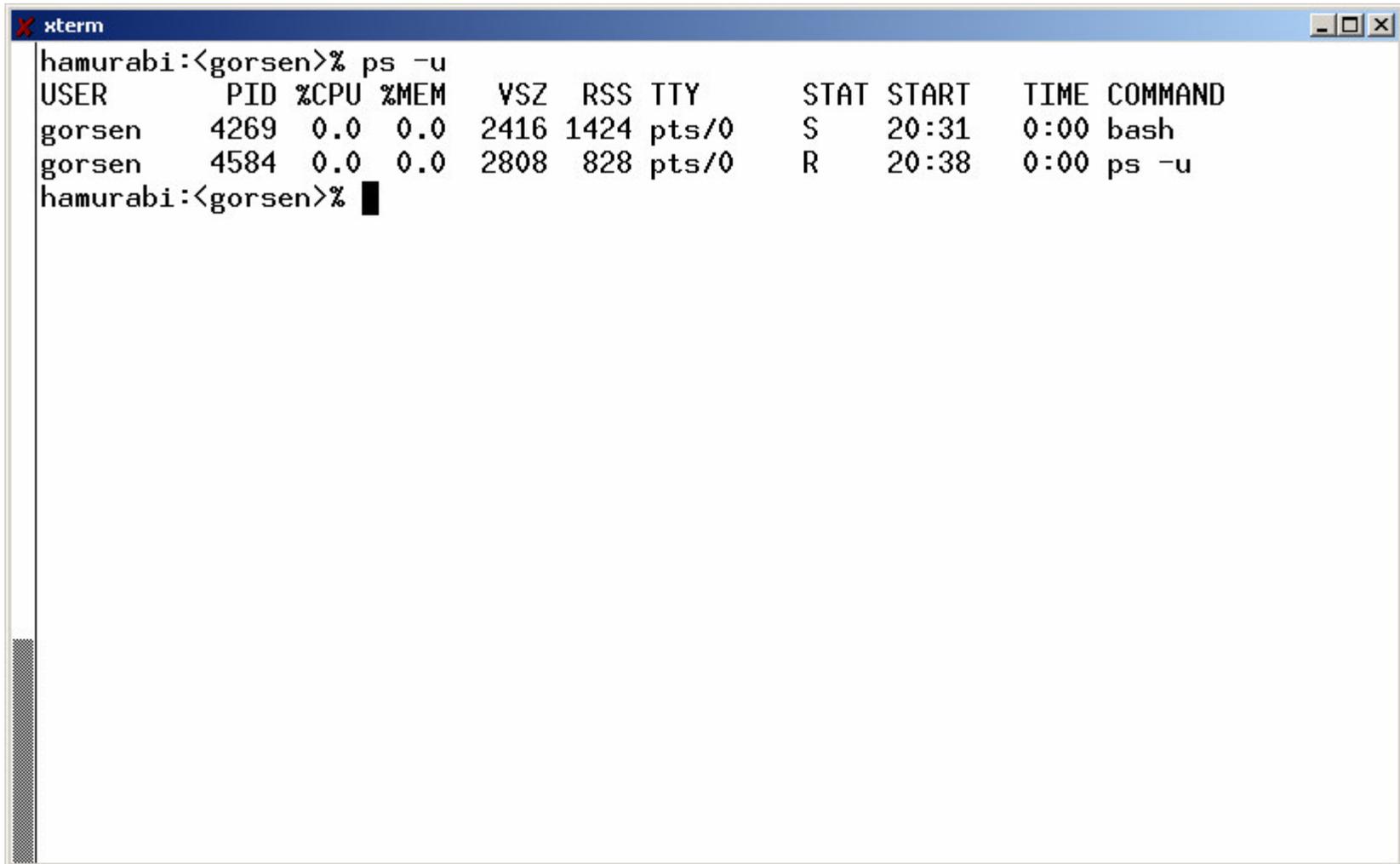
Exemples



```
xterm  
hamurabi:<gorsen>% ps -u
```

Gestion des Processus

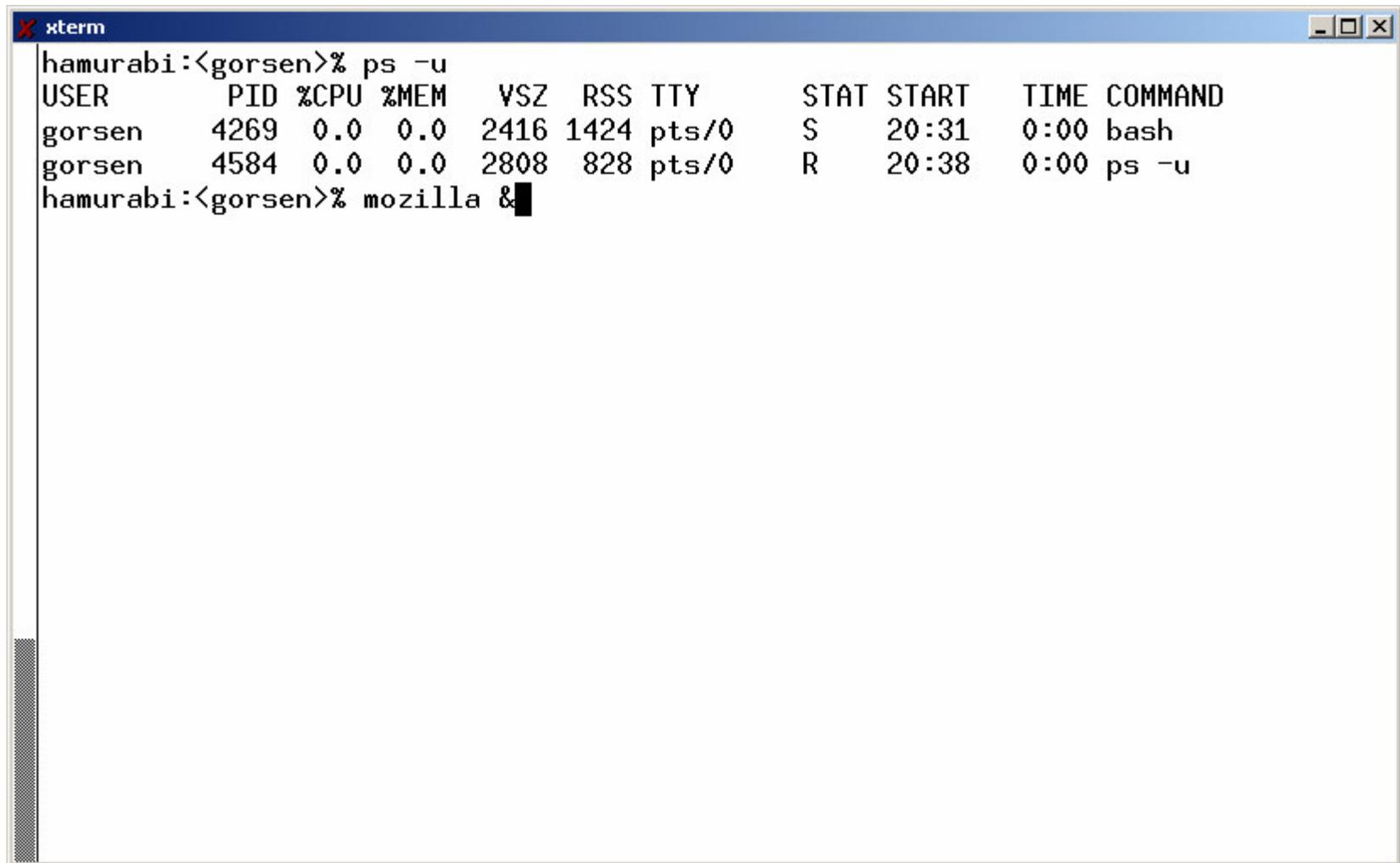
Exemples



```
hamurabi:<gorsen>% ps -u
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
gorsen    4269  0.0  0.0   2416  1424 pts/0    S     20:31   0:00 bash
gorsen    4584  0.0  0.0   2808   828 pts/0    R     20:38   0:00 ps -u
hamurabi:<gorsen>%
```

Gestion des Processus

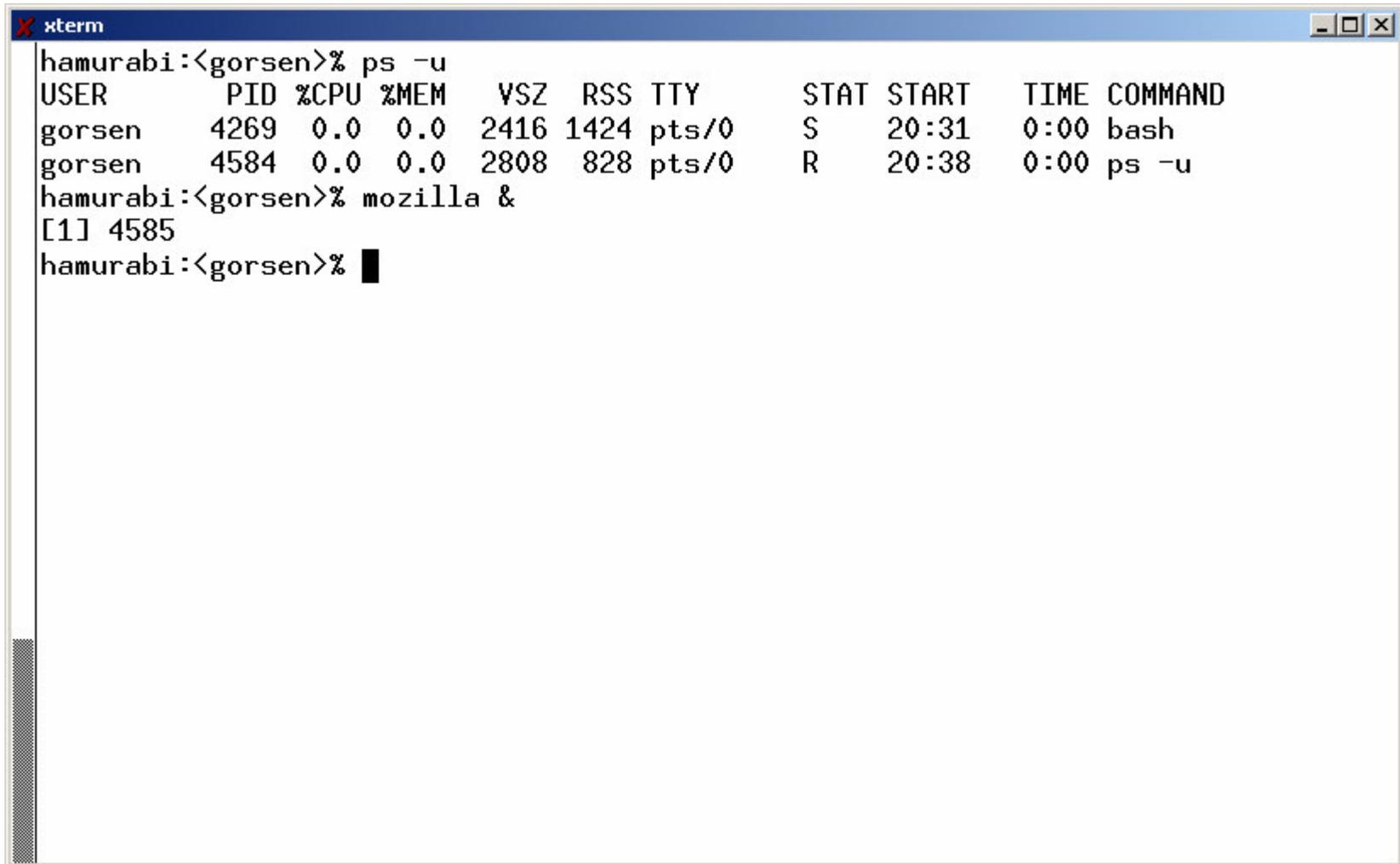
Exemples



```
xterm
hamurabi:<gorsen>% ps -u
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
gorsen    4269  0.0  0.0   2416  1424 pts/0    S    20:31   0:00 bash
gorsen    4584  0.0  0.0   2808   828 pts/0    R    20:38   0:00 ps -u
hamurabi:<gorsen>% mozilla &
```

Gestion des Processus

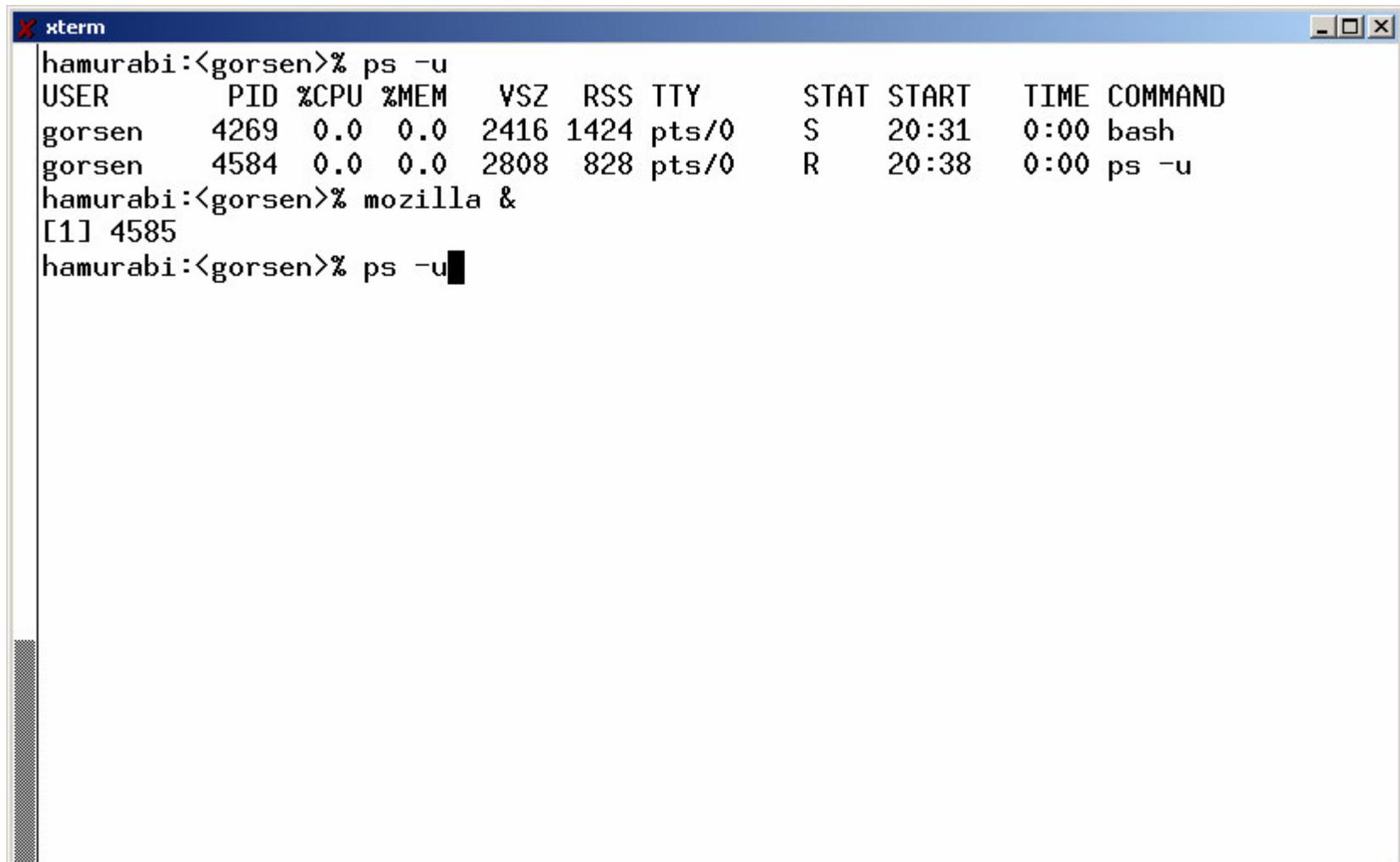
Exemples



```
xterm
hamurabi:<gorsen>% ps -u
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
gorsen    4269  0.0  0.0  2416  1424 pts/0    S    20:31   0:00 bash
gorsen    4584  0.0  0.0  2808   828 pts/0    R    20:38   0:00 ps -u
hamurabi:<gorsen>% mozilla &
[1] 4585
hamurabi:<gorsen>%
```

Gestion des Processus

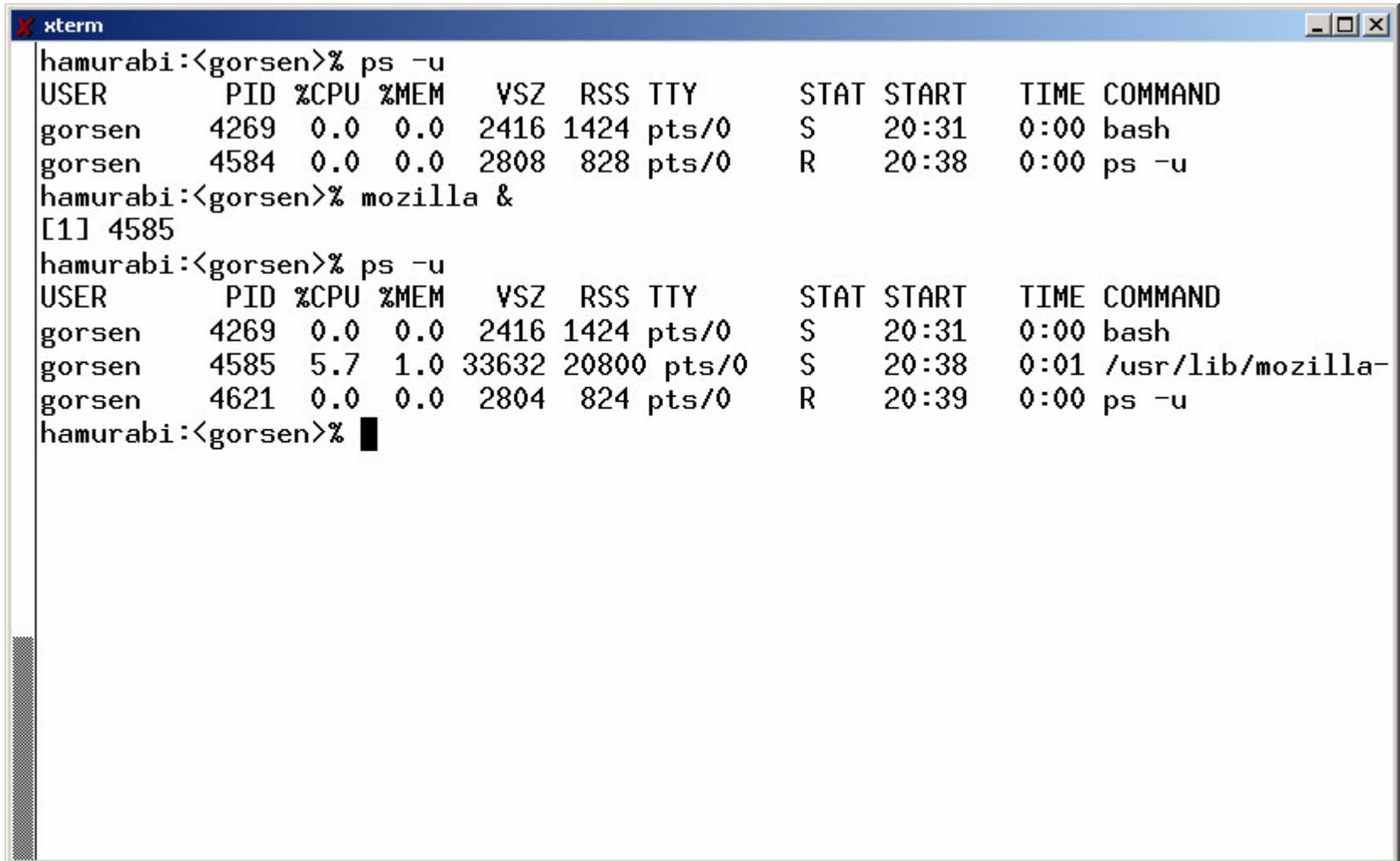
Exemples



```
xterm
hamurabi:<gorsen>% ps -u
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
gorsen    4269  0.0  0.0  2416  1424 pts/0    S    20:31   0:00  bash
gorsen    4584  0.0  0.0  2808   828 pts/0    R    20:38   0:00  ps -u
hamurabi:<gorsen>% mozilla &
[1] 4585
hamurabi:<gorsen>% ps -u
```

Gestion des Processus

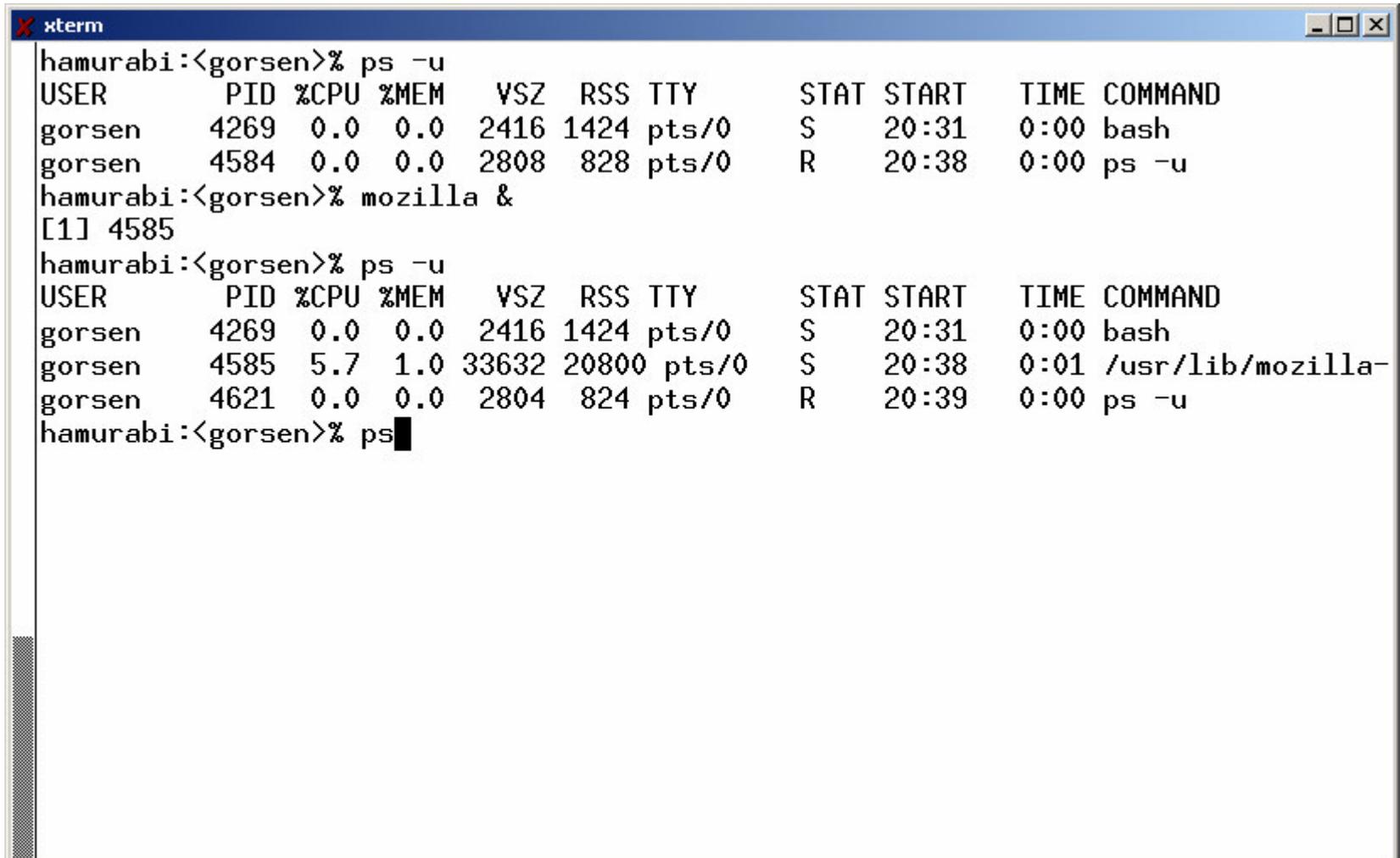
Exemples



```
xterm
hamurabi:<gorsen>% ps -u
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
gorsen    4269  0.0  0.0   2416  1424 pts/0    S    20:31   0:00  bash
gorsen    4584  0.0  0.0   2808   828 pts/0    R    20:38   0:00  ps -u
hamurabi:<gorsen>% mozilla &
[1] 4585
hamurabi:<gorsen>% ps -u
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
gorsen    4269  0.0  0.0   2416  1424 pts/0    S    20:31   0:00  bash
gorsen    4585  5.7  1.0  33632  20800 pts/0    S    20:38   0:01  /usr/lib/mozilla-
gorsen    4621  0.0  0.0   2804   824 pts/0    R    20:39   0:00  ps -u
hamurabi:<gorsen>%
```

Gestion des Processus

Exemples



```
xterm
hamurabi:<gorsen>% ps -u
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
gorsen    4269  0.0  0.0   2416  1424 pts/0    S    20:31   0:00  bash
gorsen    4584  0.0  0.0   2808   828 pts/0    R    20:38   0:00  ps -u
hamurabi:<gorsen>% mozilla &
[1] 4585
hamurabi:<gorsen>% ps -u
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
gorsen    4269  0.0  0.0   2416  1424 pts/0    S    20:31   0:00  bash
gorsen    4585  5.7  1.0  33632  20800 pts/0    S    20:38   0:01  /usr/lib/mozilla-
gorsen    4621  0.0  0.0   2804   824 pts/0    R    20:39   0:00  ps -u
hamurabi:<gorsen>% ps
```

Gestion des Processus

Exemples

```
xterm
hamurabi:<gorsen>% ps -u
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
gorsen    4269  0.0  0.0   2416  1424 pts/0    S    20:31   0:00 bash
gorsen    4584  0.0  0.0   2808   828 pts/0    R    20:38   0:00 ps -u
hamurabi:<gorsen>% mozilla &
[1] 4585
hamurabi:<gorsen>% ps -u
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
gorsen    4269  0.0  0.0   2416  1424 pts/0    S    20:31   0:00 bash
gorsen    4585  5.7  1.0  33632  20800 pts/0    S    20:38   0:01 /usr/lib/mozilla-
gorsen    4621  0.0  0.0   2804   824 pts/0    R    20:39   0:00 ps -u
hamurabi:<gorsen>% ps
  PID TTY          TIME CMD
 4269 pts/0    00:00:00 bash
 4585 pts/0    00:00:01 mozilla-bin
 4622 pts/0    00:00:00 ps
hamurabi:<gorsen>%
```

Gestion des Processus

Exemples

```
xterm
hamurabi:<gorsen>% ps -u
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
gorsen    4269  0.0  0.0   2416  1424 pts/0    S    20:31   0:00  bash
gorsen    4584  0.0  0.0   2808   828 pts/0    R    20:38   0:00  ps -u
hamurabi:<gorsen>% mozilla &
[1] 4585
hamurabi:<gorsen>% ps -u
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
gorsen    4269  0.0  0.0   2416  1424 pts/0    S    20:31   0:00  bash
gorsen    4585  5.7  1.0  33632  20800 pts/0    S    20:38   0:01  /usr/lib/mozilla-
gorsen    4621  0.0  0.0   2804   824 pts/0    R    20:39   0:00  ps -u
hamurabi:<gorsen>% ps
  PID TTY          TIME CMD
 4269 pts/0    00:00:00 bash
 4585 pts/0    00:00:01 mozilla-bin
 4622 pts/0    00:00:00 ps
hamurabi:<gorsen>% top
```

Gestion des Processus

Exemples

```

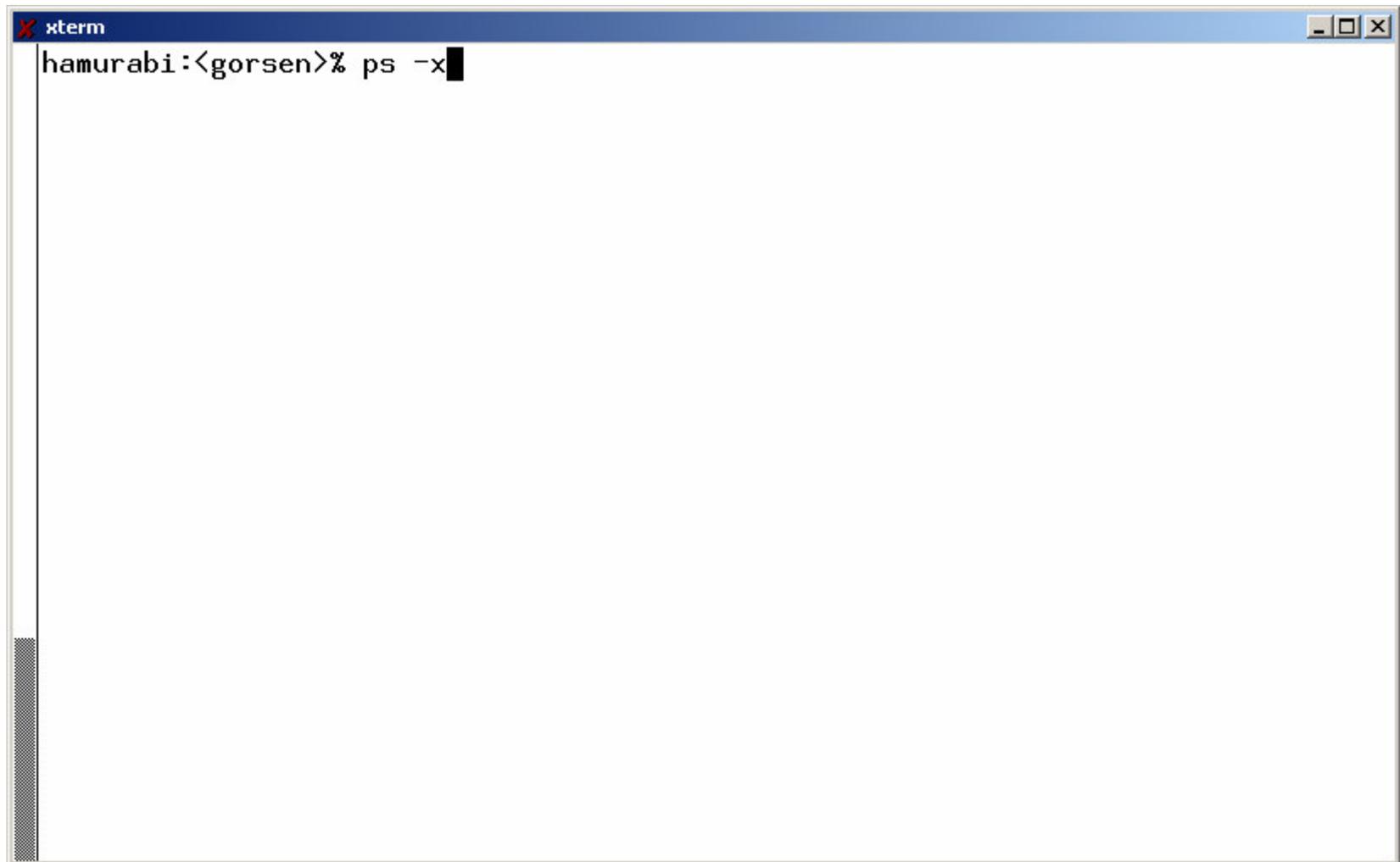
xterm
8:40pm up 16 days, 14:35, 1 user, load average: 0.93, 0.91, 0.84
74 processes: 71 sleeping, 3 running, 0 zombie, 0 stopped
CPU0 states: 7.0% user, 25.0% system, 0.0% nice, 66.0% idle
CPU1 states: 25.0% user, 55.0% system, 0.0% nice, 18.0% idle
Mem: 2064760K av, 2053120K used, 11640K free, 0K shrd, 176016K buff
Swap: 2104496K av, 17588K used, 2086908K free 1721840K cached

  PID USER   PRI  NI  SIZE  RSS  SHARE STAT  %CPU  %MEM  TIME COMMAND
 2151 david   16   0   6352   788   756 R    99.9   0.0 22207m emacs
    6 root    9   0     0     0     0 SW   11.8   0.0 602:43 kscand
    5 root    9   0     0     0     0 SW   1.9   0.0 172:48 kswapd
   13 root    9   0     0     0     0 SW   0.9   0.0  1:51 kjournald
4626 gorsen  10   0   1124  1124   864 R    0.9   0.0  0:00 top
    1 root    8   0    484   452   428 S    0.0   0.0  0:18 init
    2 root    9   0     0     0     0 SW   0.0   0.0  0:31 keventd
    3 root   19  19     0     0     0 SWN  0.0   0.0  0:00 ksoftirqd_CPU0
    4 root   19  19     0     0     0 SWN  0.0   0.0  0:00 ksoftirqd_CPU1
    7 root    9   0     0     0     0 SW   0.0   0.0  0:00 bdflush
    8 root    9   0     0     0     0 SW   0.0   0.0  0:33 kupdated
    9 root   -1 -20     0     0     0 SW<  0.0   0.0  0:00 mdrecoveryd
   63 root    9   0     0     0     0 SW   0.0   0.0  0:00 khubd
  168 root    9   0     0     0     0 SW   0.0   0.0  0:31 kjournald
  169 root    9   0     0     0     0 SW   0.0   0.0  0:55 kjournald
  170 root    9   0     0     0     0 SW   0.0   0.0  0:45 kjournald

```

Gestion des Processus

Exemples

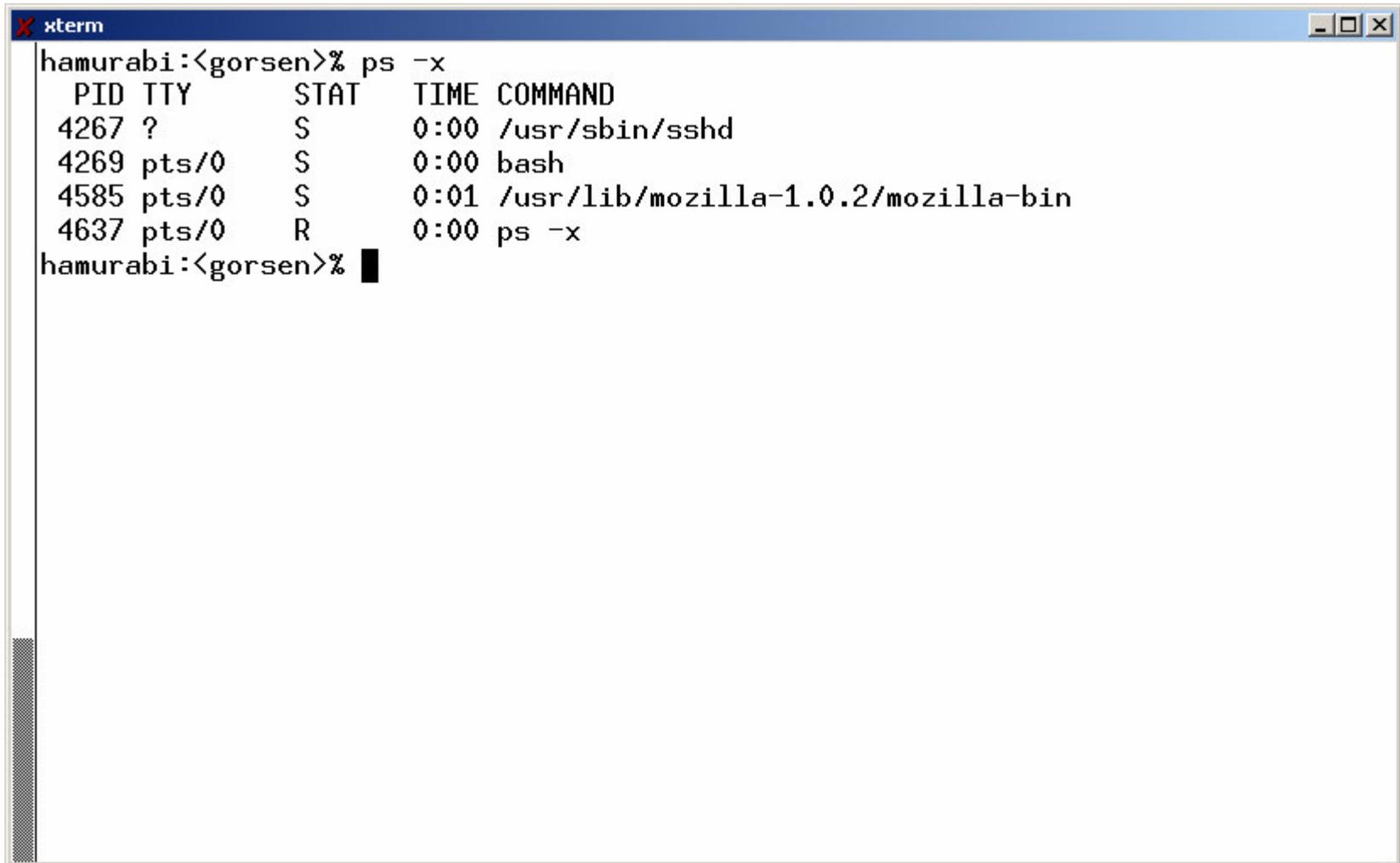


A screenshot of an xterm terminal window. The title bar shows 'xterm' with a red 'X' icon on the left and standard window control buttons (minimize, maximize, close) on the right. The terminal content shows a prompt 'hamurabi:<gorsen>% ' followed by the command 'ps -x' and a black cursor block at the end of the line.

```
hamurabi:<gorsen>% ps -x
```

Gestion des Processus

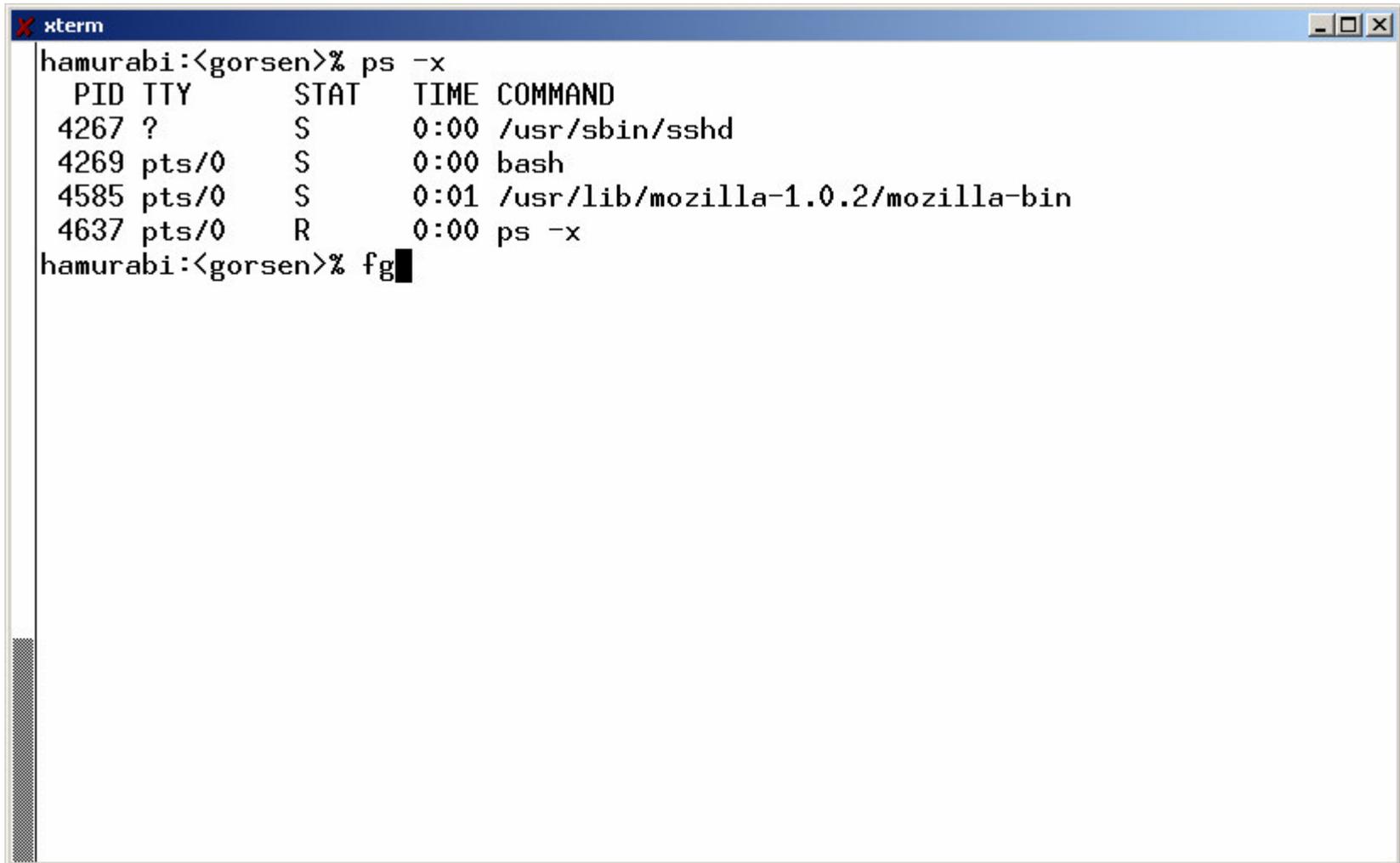
Exemples



```
xterm
hamurabi:<gorsen>% ps -x
  PID TTY          STAT       TIME COMMAND
 4267 ?            S           0:00 /usr/sbin/sshd
 4269 pts/0        S           0:00 bash
 4585 pts/0        S           0:01 /usr/lib/mozilla-1.0.2/mozilla-bin
 4637 pts/0        R           0:00 ps -x
hamurabi:<gorsen>%
```

Gestion des Processus

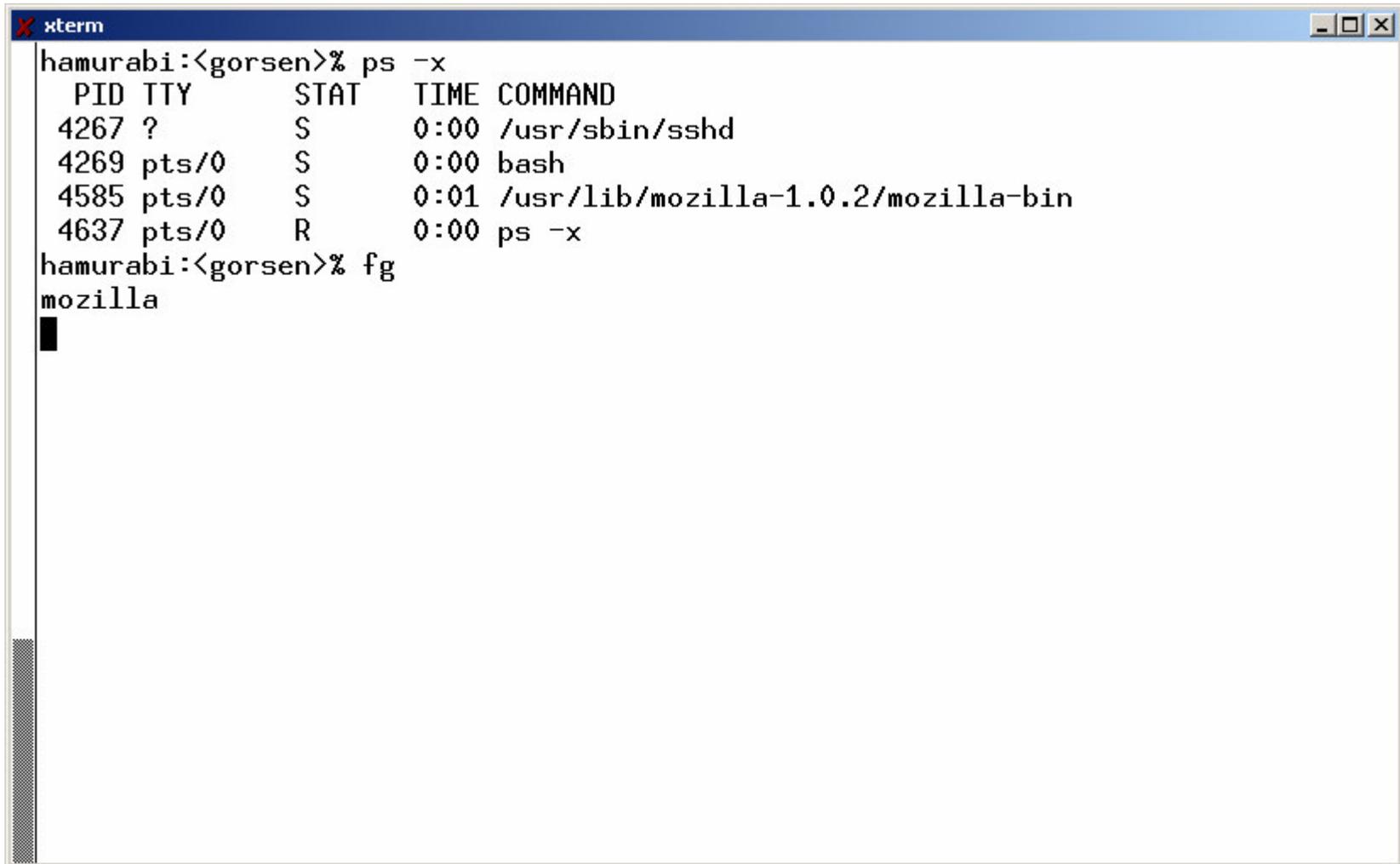
Exemples



```
xterm
hamurabi:<gorsen>% ps -x
  PID TTY          STAT       TIME COMMAND
 4267 ?            S            0:00 /usr/sbin/sshd
 4269 pts/0        S            0:00 bash
 4585 pts/0        S            0:01 /usr/lib/mozilla-1.0.2/mozilla-bin
 4637 pts/0        R            0:00 ps -x
hamurabi:<gorsen>% fg
```

Gestion des Processus

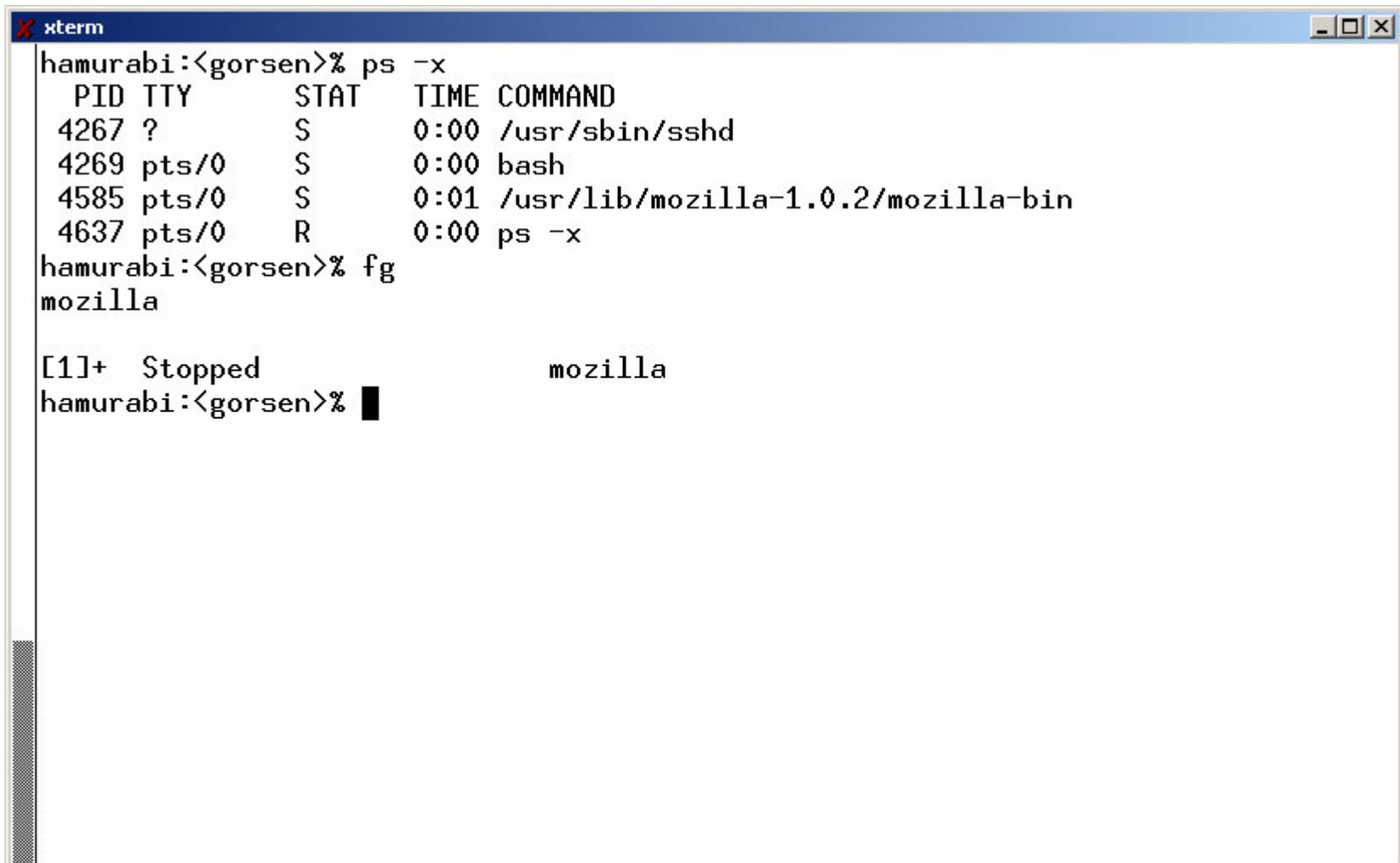
Exemples



```
xterm
hamurabi:<gorsen>% ps -x
  PID TTY          STAT       TIME COMMAND
  4267 ?            S           0:00 /usr/sbin/sshd
  4269 pts/0        S           0:00 bash
  4585 pts/0        S           0:01 /usr/lib/mozilla-1.0.2/mozilla-bin
  4637 pts/0        R           0:00 ps -x
hamurabi:<gorsen>% fg
mozilla
█
```

Gestion des Processus

Exemples

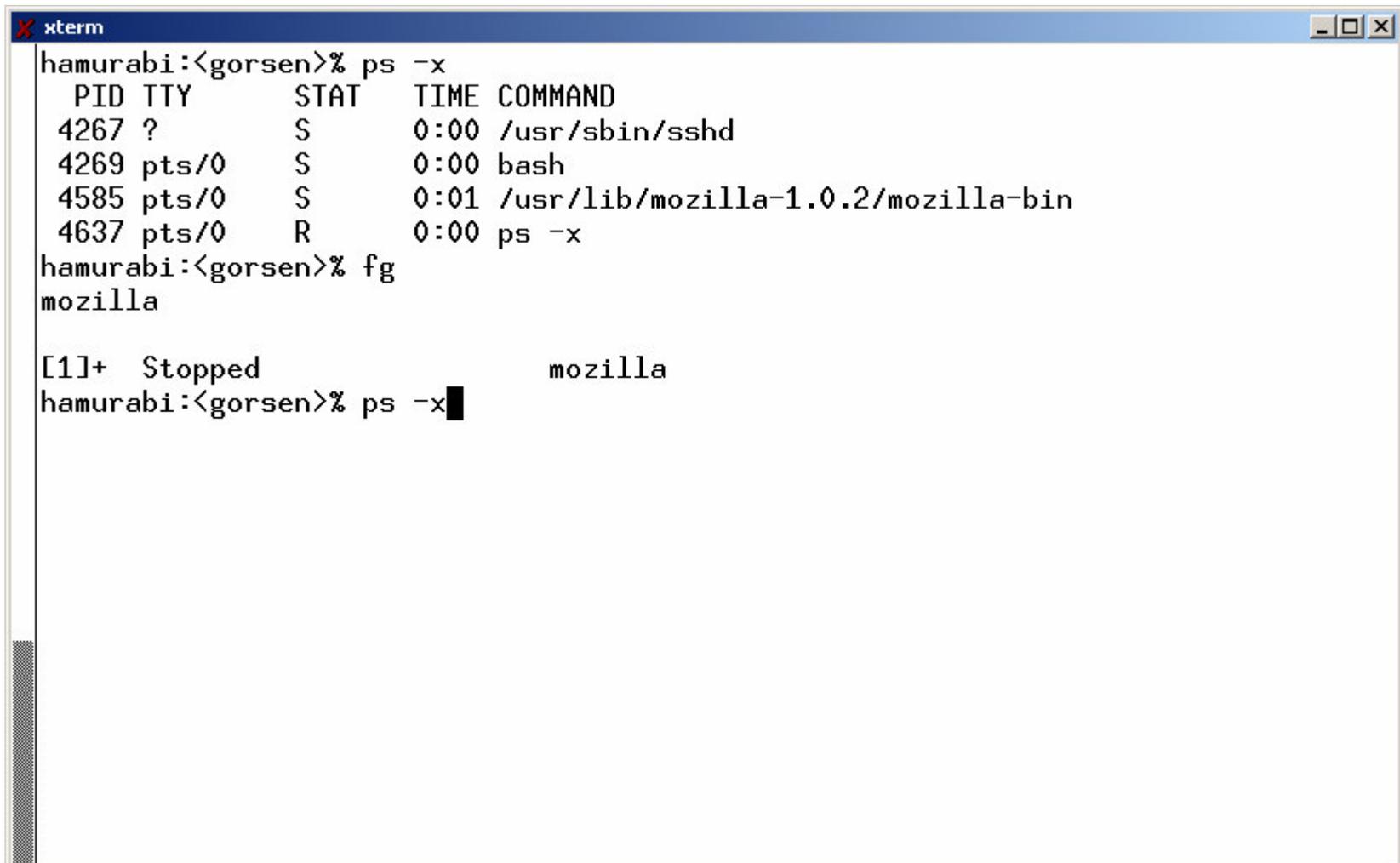


```
xterm
hamurabi:<gorsen>% ps -x
  PID TTY          STAT       TIME COMMAND
  4267 ?            S           0:00 /usr/sbin/sshd
  4269 pts/0        S           0:00 bash
  4585 pts/0        S           0:01 /usr/lib/mozilla-1.0.2/mozilla-bin
  4637 pts/0        R           0:00 ps -x
hamurabi:<gorsen>% fg
mozilla

[1]+  Stopped                  mozilla
hamurabi:<gorsen>%
```

Gestion des Processus

Exemples

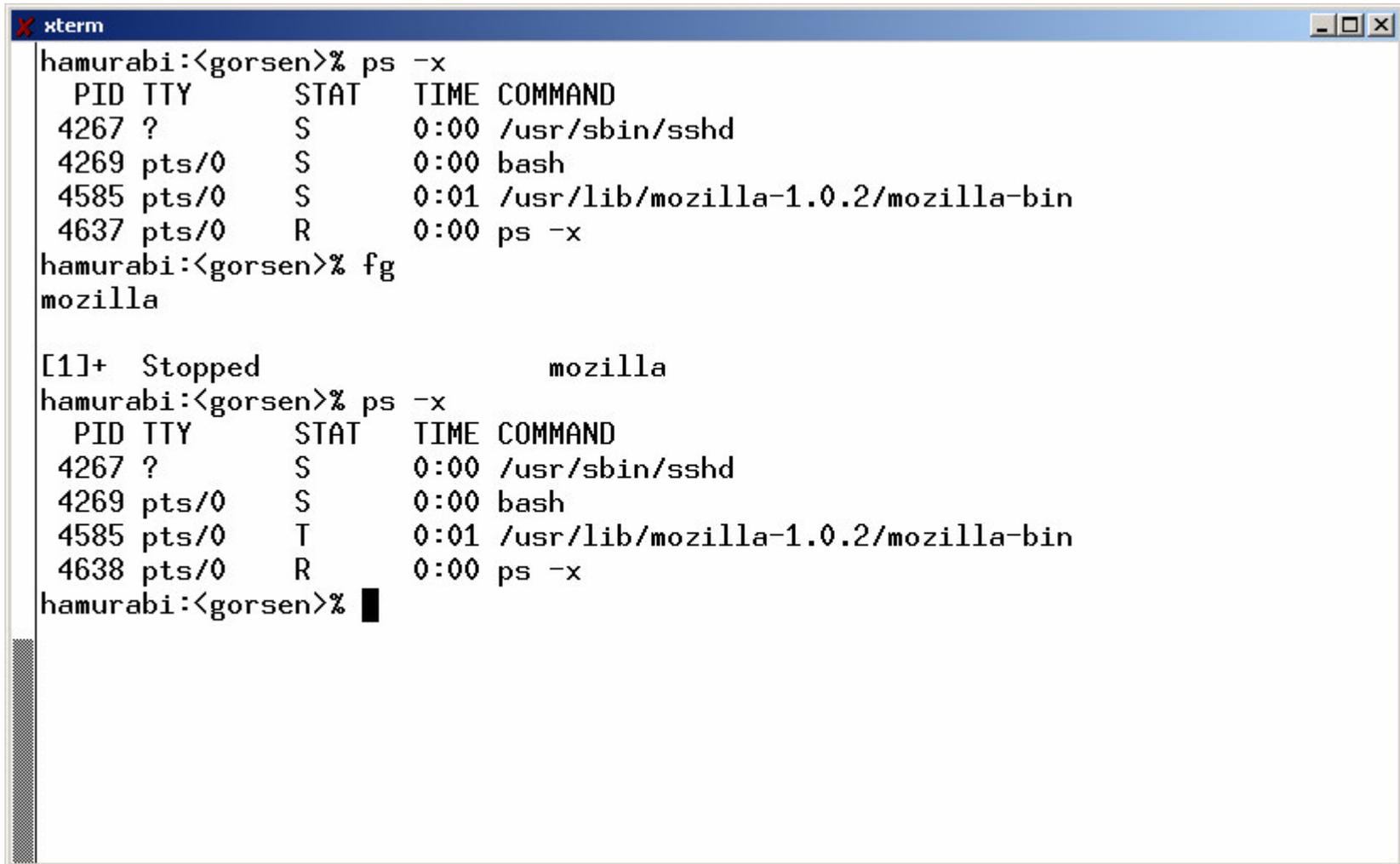


```
xterm
hamurabi:<gorsen>% ps -x
  PID TTY          STAT       TIME COMMAND
  4267 ?            S           0:00 /usr/sbin/sshd
  4269 pts/0        S           0:00 bash
  4585 pts/0        S           0:01 /usr/lib/mozilla-1.0.2/mozilla-bin
  4637 pts/0        R           0:00 ps -x
hamurabi:<gorsen>% fg
mozilla

[1]+  Stopped                  mozilla
hamurabi:<gorsen>% ps -x
```

Gestion des Processus

Exemples

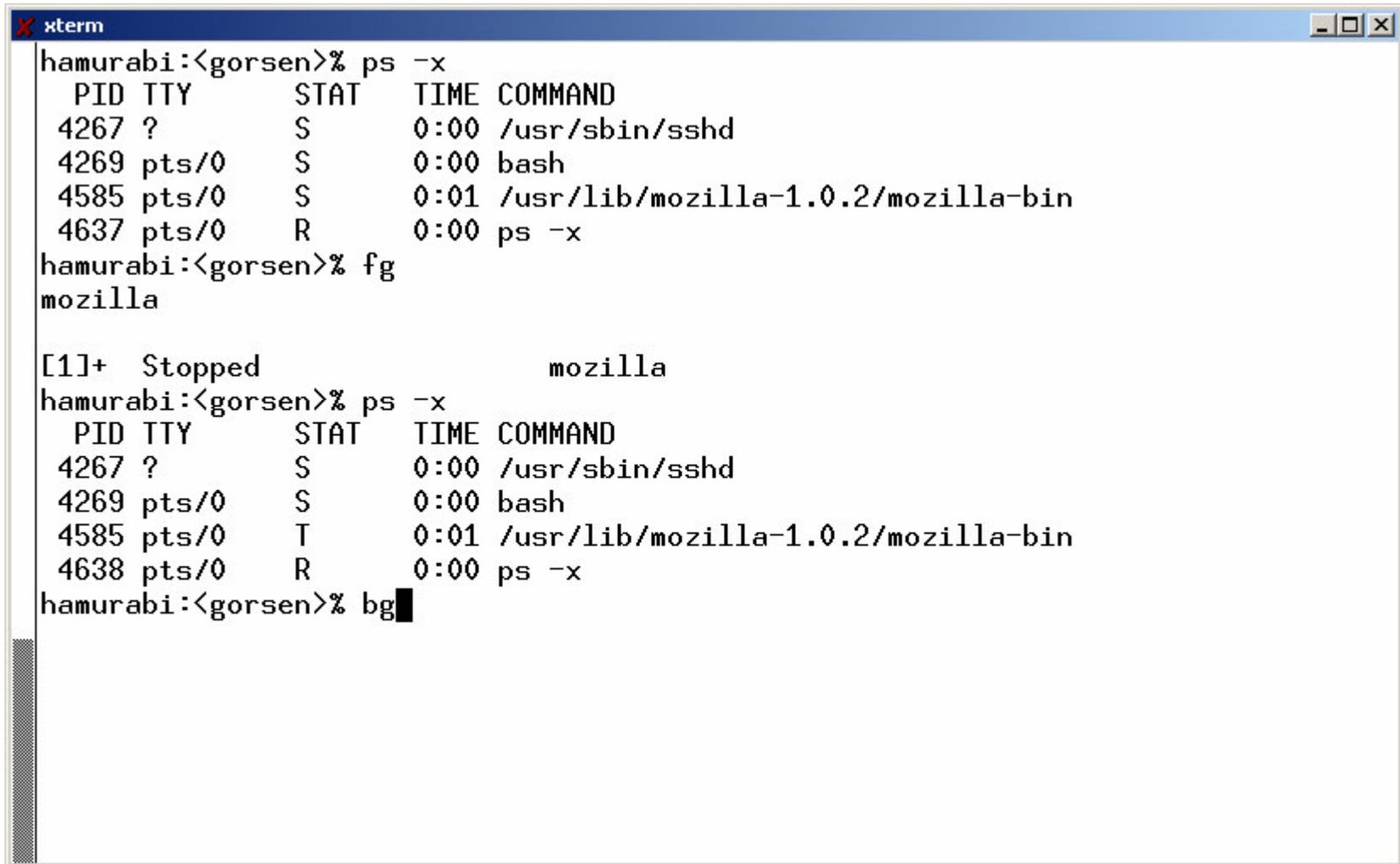


```
xterm
hamurabi:<gorsen>% ps -x
  PID TTY          STAT       TIME COMMAND
 4267 ?            S            0:00 /usr/sbin/sshd
 4269 pts/0        S            0:00 bash
 4585 pts/0        S            0:01 /usr/lib/mozilla-1.0.2/mozilla-bin
 4637 pts/0        R            0:00 ps -x
hamurabi:<gorsen>% fg
mozilla

[1]+  Stopped                  mozilla
hamurabi:<gorsen>% ps -x
  PID TTY          STAT       TIME COMMAND
 4267 ?            S            0:00 /usr/sbin/sshd
 4269 pts/0        S            0:00 bash
 4585 pts/0        T            0:01 /usr/lib/mozilla-1.0.2/mozilla-bin
 4638 pts/0        R            0:00 ps -x
hamurabi:<gorsen>%
```

Gestion des Processus

Exemples

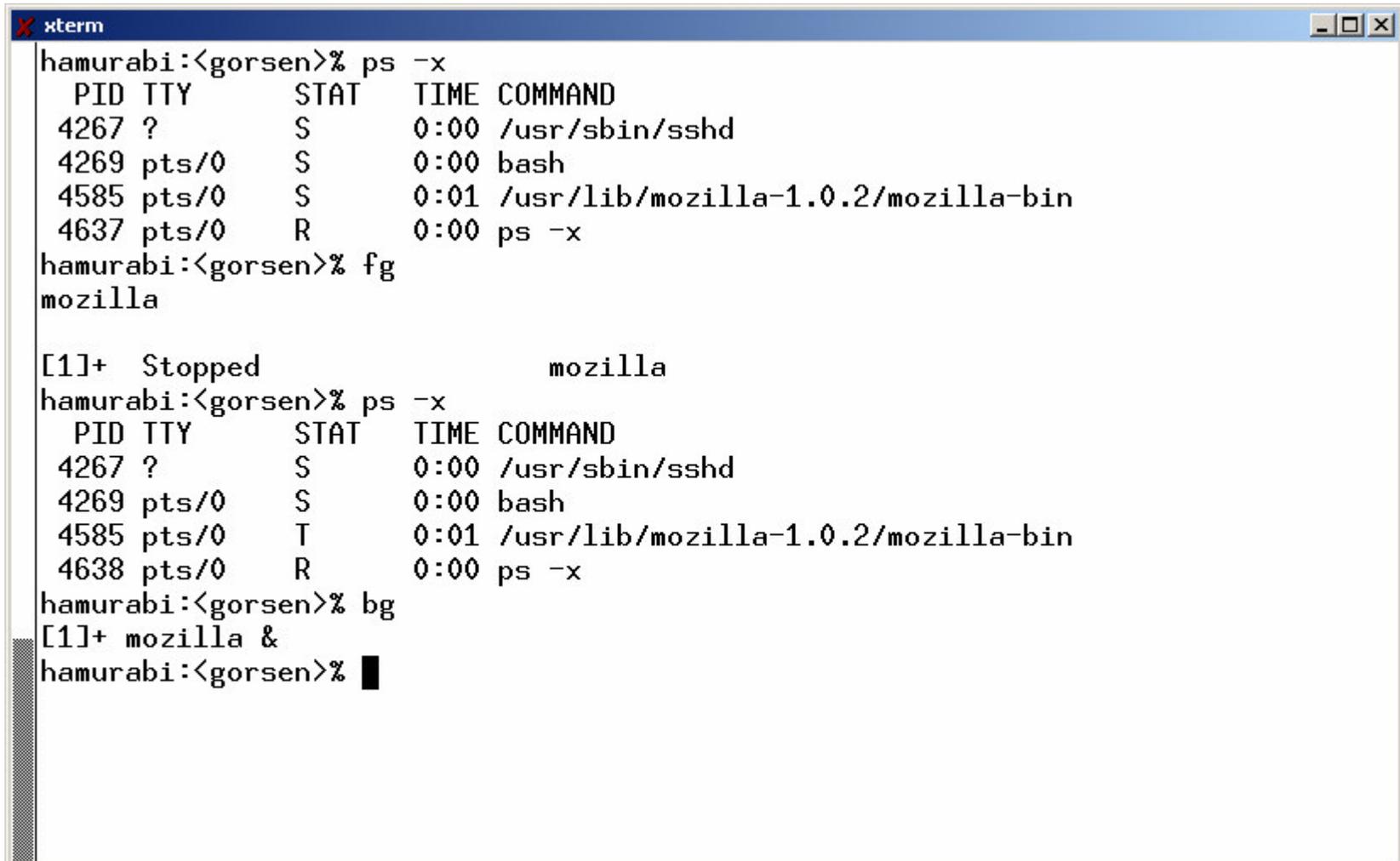


```
xterm
hamurabi:<gorsen>% ps -x
  PID TTY          STAT       TIME COMMAND
 4267 ?            S           0:00 /usr/sbin/sshd
 4269 pts/0        S           0:00 bash
 4585 pts/0        S           0:01 /usr/lib/mozilla-1.0.2/mozilla-bin
 4637 pts/0        R           0:00 ps -x
hamurabi:<gorsen>% fg
mozilla

[1]+  Stopped                  mozilla
hamurabi:<gorsen>% ps -x
  PID TTY          STAT       TIME COMMAND
 4267 ?            S           0:00 /usr/sbin/sshd
 4269 pts/0        S           0:00 bash
 4585 pts/0        T           0:01 /usr/lib/mozilla-1.0.2/mozilla-bin
 4638 pts/0        R           0:00 ps -x
hamurabi:<gorsen>% bg
```

Gestion des Processus

Exemples



```
xterm
hamurabi:<gorsen>% ps -x
  PID TTY          STAT       TIME COMMAND
 4267 ?            S           0:00 /usr/sbin/sshd
 4269 pts/0        S           0:00 bash
 4585 pts/0        S           0:01 /usr/lib/mozilla-1.0.2/mozilla-bin
 4637 pts/0        R           0:00 ps -x
hamurabi:<gorsen>% fg
mozilla

[1]+  Stopped                  mozilla
hamurabi:<gorsen>% ps -x
  PID TTY          STAT       TIME COMMAND
 4267 ?            S           0:00 /usr/sbin/sshd
 4269 pts/0        S           0:00 bash
 4585 pts/0        T           0:01 /usr/lib/mozilla-1.0.2/mozilla-bin
 4638 pts/0        R           0:00 ps -x
hamurabi:<gorsen>% bg
[1]+  mozilla &
hamurabi:<gorsen>%
```

Gestion des Processus

Exemples

```
xterm
hamurabi:<gorsen>% ps -x
  PID TTY          STAT       TIME COMMAND
 4267 ?            S            0:00 /usr/sbin/sshd
 4269 pts/0        S            0:00 bash
 4585 pts/0        S            0:01 /usr/lib/mozilla-1.0.2/mozilla-bin
 4637 pts/0        R            0:00 ps -x
hamurabi:<gorsen>% fg
mozilla

[1]+  Stopped                  mozilla
hamurabi:<gorsen>% ps -x
  PID TTY          STAT       TIME COMMAND
 4267 ?            S            0:00 /usr/sbin/sshd
 4269 pts/0        S            0:00 bash
 4585 pts/0        T            0:01 /usr/lib/mozilla-1.0.2/mozilla-bin
 4638 pts/0        R            0:00 ps -x
hamurabi:<gorsen>% bg
[1]+  mozilla &
hamurabi:<gorsen>% ps -x
```

Gestion des Processus

Exemples

```
xterm
PID TTY      STAT   TIME COMMAND
4267 ?        S      0:00 /usr/sbin/sshd
4269 pts/0     S      0:00 bash
4585 pts/0     S      0:01 /usr/lib/mozilla-1.0.2/mozilla-bin
4637 pts/0     R      0:00 ps -x
hamurabi:<gorsen>% fg
mozilla

[1]+  Stopped                  mozilla
hamurabi:<gorsen>% ps -x
  PID TTY      STAT   TIME COMMAND
 4267 ?        S      0:00 /usr/sbin/sshd
 4269 pts/0     S      0:00 bash
 4585 pts/0     T      0:01 /usr/lib/mozilla-1.0.2/mozilla-bin
 4638 pts/0     R      0:00 ps -x
hamurabi:<gorsen>% bg
[1]+  mozilla &
hamurabi:<gorsen>% ps -x
  PID TTY      STAT   TIME COMMAND
 4267 ?        S      0:00 /usr/sbin/sshd
 4269 pts/0     S      0:00 bash
 4585 pts/0     S      0:01 /usr/lib/mozilla-1.0.2/mozilla-bin
 4639 pts/0     R      0:00 ps -x
hamurabi:<gorsen>%
```

Gestion des Processus

Exemples



A screenshot of an xterm terminal window. The title bar shows 'xterm' with a red 'X' icon on the left and standard window control buttons (minimize, maximize, close) on the right. The terminal content shows a shell prompt 'hamurabi:<gorsen>% ' followed by the command 'pstree -u gorsen' and a black cursor at the end of the line.

```
hamurabi:<gorsen>% pstree -u gorsen
```

Gestion des Processus

Exemples



```
xterm
hamurabi:<gorsen>% pstree -u gorsen
sshd---bash+-mozilla-bin---mozilla-bin---3*[mozilla-bin]
      ^-pstree
hamurabi:<gorsen>% █
```

Gestion des Processus

Exemples



```
xterm
hamurabi:<gorsen>% pstree -u gorsen
sshd---bash-+-mozilla-bin---mozilla-bin---3*[mozilla-bin]
      ^-pstree
hamurabi:<gorsen>% xterm &
```

Gestion des Processus

Exemples



```
xterm
hamurabi:<gorsen>% pstree -u gorsen
sshd---bash-+-mozilla-bin---mozilla-bin---3*[mozilla-bin]
      ^-pstree
hamurabi:<gorsen>% xterm &
[2] 4875
hamurabi:<gorsen>% xterm &
```

Gestion des Processus

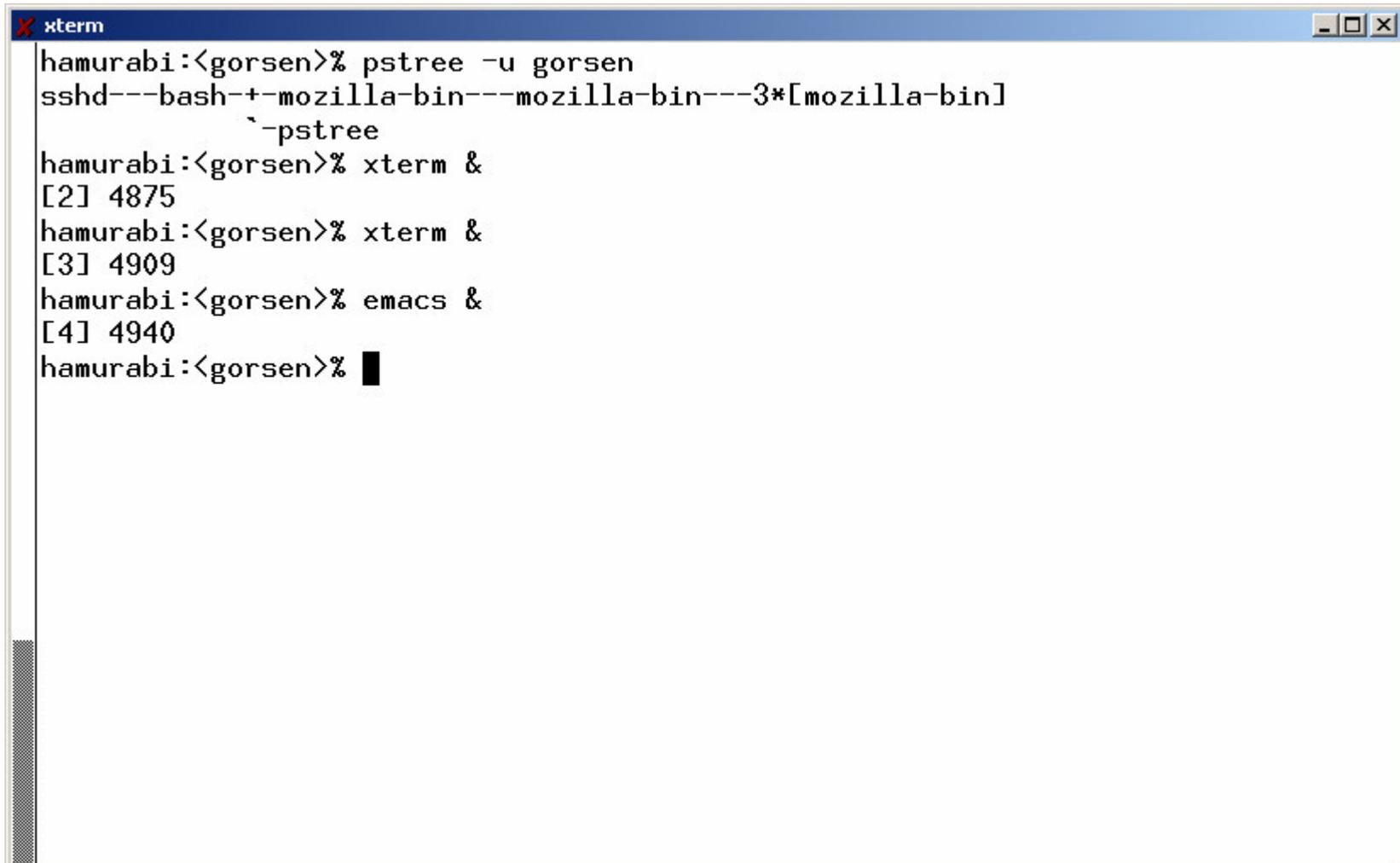
Exemples



```
xterm
hamurabi:<gorsen>% pstree -u gorsen
sshd---bash-+-mozilla-bin---mozilla-bin---3*[mozilla-bin]
      ^-pstree
hamurabi:<gorsen>% xterm &
[2] 4875
hamurabi:<gorsen>% xterm &
[3] 4909
hamurabi:<gorsen>% emacs &
```

Gestion des Processus

Exemples



```
xterm
hamurabi:<gorsen>% pstree -u gorsen
sshd---bash+-mozilla-bin---mozilla-bin---3*[mozilla-bin]
      ^-pstree
hamurabi:<gorsen>% xterm &
[2] 4875
hamurabi:<gorsen>% xterm &
[3] 4909
hamurabi:<gorsen>% emacs &
[4] 4940
hamurabi:<gorsen>% █
```

Gestion des Processus

Exemples



```
xterm
hamurabi:<gorsen>% pstree -u gorsen
sshd---bash-+-mozilla-bin---mozilla-bin---3*[mozilla-bin]
      ^-pstree
hamurabi:<gorsen>% xterm &
[2] 4875
hamurabi:<gorsen>% xterm &
[3] 4909
hamurabi:<gorsen>% emacs &
[4] 4940
hamurabi:<gorsen>% pstree -u gorsen
```

Gestion des Processus

Exemples



```
xterm
hamurabi:<gorsen>% pstree -u gorsen
sshd---bash-+-mozilla-bin---mozilla-bin---3*[mozilla-bin]
      ^-pstree
hamurabi:<gorsen>% xterm &
[2] 4875
hamurabi:<gorsen>% xterm &
[3] 4909
hamurabi:<gorsen>% emacs &
[4] 4940
hamurabi:<gorsen>% pstree -u gorsen
sshd---bash-+-emacs
          |-mozilla-bin---mozilla-bin---3*[mozilla-bin]
          |-pstree
          ^-2*[xterm---bash]
hamurabi:<gorsen>% █
```

Gestion des Processus

Exemples



```
xterm
hamurabi:<gorsen>% pstree -u gorsen
sshd---bash-+-mozilla-bin---mozilla-bin---3*[mozilla-bin]
      ^-pstree
hamurabi:<gorsen>% xterm &
[2] 4875
hamurabi:<gorsen>% xterm &
[3] 4909
hamurabi:<gorsen>% emacs &
[4] 4940
hamurabi:<gorsen>% pstree -u gorsen
sshd---bash-+-emacs
          |-mozilla-bin---mozilla-bin---3*[mozilla-bin]
          |-pstree
          ^-2*[xterm---bash]
hamurabi:<gorsen>% pstree -u gorsen
```

Gestion des Processus

Exemples

```

xterm
hamurabi:<gorsen>% pstree -u gorsen
sshd---bash-+-mozilla-bin---mozilla-bin---3*[mozilla-bin]
      ^-pstree
hamurabi:<gorsen>% xterm &
[2] 4875
hamurabi:<gorsen>% xterm &
[3] 4909
hamurabi:<gorsen>% emacs &
[4] 4940
hamurabi:<gorsen>% pstree -u gorsen
sshd---bash-+-emacs
          |-mozilla-bin---mozilla-bin---3*[mozilla-bin]
          |-pstree
          ^-2*[xterm---bash]
hamurabi:<gorsen>% pstree -u gorsen
sshd---bash-+-emacs
          |-mozilla-bin---mozilla-bin---3*[mozilla-bin]
          |-pstree
          |-xterm---bash---pine
          ^-xterm---bash
hamurabi:<gorsen>% █

```

Gestion des Processus

A savoir

- Ne JAMAIS éteindre une station UNIX
- Si une application bloque l'interface graphique de la station sur laquelle vous travaillez :
 - Ouvrir une session sur une autre station
 - Etablir une session à distance sur la station bloquée (telnet, ssh, à vous de choisir)
 - Identifier le PID de l'application fautive avec **ps**
 - Tuer l'application fautive
 - Avec **kill**
 - Avec **kill -9** si **kill** ne suffit pas
 - Le tour est joué

Egrep et Expressions Régulières

Introduction à UNIX



Egrep

Introduction

- Recherche dans des fichiers
 - D'un chaîne ou d'une sous chaîne de caractères
 - Simplement d'un mot
 - D'une chaîne *formalisée* par une expression régulière
- Utilisation et informations
 - **egrep [options] <chaîne recherchée> <fichier>**
 - **man egrep**
- Résultat
 - Lignes du fichier contenant ce qui est recherché
 - Ou autre résultat, suivant les options utilisées

Egrep

Introduction

- Il existe en réalité deux versions :
 - **grep** qui supporte simplement les expressions régulières de base
 - **egrep** qui supporte les expressions régulières dites étendues et qui offre donc plus de puissance
- La commande **egrep** supporte toutes les options de la commande **grep**
- Nous utiliserons donc dans ce cours **egrep**

Egrep

Recherche d'une Chaîne

- `egrep <chaîne> fichier`

Juchée à l'extrémité **nord**-est de la table qui me sert de bureau, une lampe imbibe de sa lumière tamisée les quelques papiers privilégiés qui n'ont pas encore leur place dans le fatras des classeurs habitant l'étagère à laquelle je tourne le dos avec une feinte obstination.

```
biotope% egrep nord lettre.txt
```

```
Juchée à l'extrémité nord-est de la table qui me sert de bureau,
```

Egrep

Recherche d'une Chaîne

- `egrep <chaîne> fichier`

Juchée à l'extrémité nord-est de la table qui me sert de bureau, une lampe imbibe de sa lumière tamisée les quelques papiers privilégiés qui n'ont pas encore leur place dans le fatras des classeurs habitant l'étagère à laquelle je tourne le dos avec une feinte obstination.

```
biotope% egrep or lettre.txt
```

```
Juchée à l'extrémité nord-est de la table qui me sert de bureau,  
privilégiés qui n'ont pas encore leur place parmi le fatras des
```

Egrep

Recherche *Inversée*

- `egrep -v <chaîne> fichier`

Juchée à l'extrémité nord-est de la table qui me sert de bureau, une lampe imbibe de sa lumière tamisée les **quelques** papiers privilégiés qui n'ont pas encore leur place parmi le fatras des classeurs habitant l'étagère à **laquelle** je tourne le dos avec une feinte obstination.

```
biotope% egrep -v que lettre.txt
```

```
Juchée à l'extrémité nord-est de la table qui me sert de bureau,  
privilégiés qui n'ont pas encore leur place dans le fatras des  
feinte obstination.
```

Egrep

Recherche d'un Mot Exact

- `egrep -w <chaîne> fichier`

Juchée à l'extrémité nord-est de la table qui me sert de bureau, une lampe imbibe de sa lumière tamisée les quelques papiers privilégiés qui n'ont pas encore leur place parmi le fatras des classeurs habitant l'étagère à laquelle je tourne le dos avec une feinte obstination.

```
biotope% egrep -w le lettre.txt  
privilégiés qui n'ont pas encore leur place dans le fatras des  
classeurs habitant l'étagère à laquelle je tourne le dos avec une
```

Egrep

Lignes de Contexte

- `egrep -<nombre de lignes> <chaîne> fichier`

Juchée à l'extrémité nord-est de la table qui me sert de bureau,
→ une lampe imbibe de sa lumière tamisée les quelques papiers
privilégiés qui n'ont pas encore leur place dans le fatras des
→ classeurs habitant l'étagère à laquelle je tourne le dos avec une
feinte obstination.

```
biotope% egrep -1 leur lettre.txt
```

→ une lampe imbibe de sa lumière tamisée les quelques papiers
privilégiés qui n'ont pas encore leur place parmi le fatras des
→ classeurs habitant l'étagère à laquelle je tourne le dos avec une

Egrep

Numéro de Ligne

- `egrep -n <chaîne> fichier`

Juchée à l'extrémité nord-est de la table qui me sert de bureau, une lampe imbibe de sa lumière tamisée les quelques papiers privilégiés qui n'ont pas encore **leur** place dans le fatras des classeurs habitant l'étagère à laquelle je tourne le dos avec une feinte obstination.

```
biotope% egrep -n leur lettre.txt
```

```
3:privilégiés qui n'ont pas encore leur place parmi le fatras des
```

Egrep

Combinaison d'Options

- `egrep -n <nombre de lignes> <chaîne> fichier`

Juchée à l'extrémité nord-est de la table qui me sert de bureau,
→ une lampe imbibe de sa lumière tamisée les quelques papiers
privilégiés qui n'ont pas encore leur place parmi le fatras des
→ classeurs habitant l'étagère à laquelle je tourne le dos avec une
feinte obstination.

```
biotope% egrep -n1 leur lettre.txt
```

→ 2-une lampe imbibe de sa lumière tamisée les quelques papiers
3:privilégiés qui n'ont pas encore leur place parmi le fatras des
→ 4-classeurs habitant l'étagère à laquelle je tourne le dos avec une

Egrep

Respect de la Casse

- `egrep -i <chaîne> fichier`

Juchée à l'extrémité nord-est de la table qui me sert de bureau, une lampe imbibe de sa lumière tamisée les quelques papiers privilégiés qui n'ont pas encore leur place parmi le fatras des classeurs habitant l'étagère à laquelle **j**e tourne le dos avec une feinte obstination.

```
biotope% egrep j lettre.txt
```

```
classeurs habitant l'étagère à laquelle je tourne le dos avec une
```

```
biotope% egrep -i j lettre.txt
```

```
Juchée à l'extrémité nord-est de la table qui me sert de bureau,
```

```
classeurs habitant l'étagère à laquelle je tourne le dos avec une
```

Egrep

Compte des occurrences

- `egrep -c <chaîne> fichier`

Juchée à l'extrémité nord-est de la table qui me sert de bureau, une lampe imbibe de sa lumière tamisée les **quelques** papiers privilégiés qui n'ont pas encore leur place parmi le fatras des classeurs habitant l'étagère à **laquelle** je tourne le dos avec une feinte obstination.

```
biotope% egrep -c que lettre.txt  
2
```

Egrep

Exemple Concret

- Imaginons le fichier de notes suivant

```
crepetna:Crepet, Nathalie:CREN1807750:92:87:88:54:70
yosnheat:Yos Nhean, Trakal:YOST19087603:84:73:70:50:73
benelaur:Benel, Aurelien:BENA80207700:84:73:89:45:100
soucypas:Soucy, Pascal:SOUP14067502:95:90:89:87:99
```

- On peut extraire les lignes qui contiennent une note comprise entre 90 et 99

```
biotope% egrep :9 notes.txt
crepetna:Crepet, Nathalie:CREN1807750:92:87:88:54:70
soucypas:Soucy, Pascal:SOUP14067502:95:90:89:87:99
```

Egrep

Exemple Concret

```
biotope% egrep :9 notes.txt  
crepetna:Crepet, Nathalie:CREN1807750:92:87:88:54:70  
soucypas:Soucy, Pascal:SOUP14067502:95:90:89:87:99
```

- En combinant la commande avec **cut**, on peut extraire les noms des personnes concernées

```
biotope% egrep :9 notes.txt | cut -d: -f2  
Crepet, Nathalie  
Soucy, Pascal
```

Egrep

Limite ?

```
crepetna:Crepet, Nathalie:CREN1807750:92:87:88:54:70  
yosnheat:Yos Nhean, Trakal:YOST19087603:84:73:70:50:73  
benelaur:Benel, Aurelien:BENA80207700:84:73:89:45:100  
soucypas:Soucy, Pascal:SOUP14067502:95:90:89:87:99
```

- Comment faire pour extraire les lignes où la dernière note est comprise entre 90 et 99 ?
- Solution : Utiliser une expression régulière
- Mais, qu'est ce qu'une expression régulière ?

Expressions Régulières Introduction

- Définition
 - Formule qui représente une chaîne de caractères
 - Composée de caractères et d'opérateurs
- Utilisation
 - On recherche alors non pas un mot ou une simple chaîne de caractères mais une suite de caractères qui correspondent aux critères énoncés par la formule
 - Certains opérateurs doivent être précédés d'un \ pour ne pas entrer en conflit avec le shell, ainsi : {, }, <, >, (,) et | seront écrits \{, \}, \<, \>, \(, \) et \| . C'est aussi le cas de l'espace
- Culture Générale
 - Expression régulière, "regular expression" en anglais, se traduit en bon français par "expression rationnelle" mais l'usage est de dire "régulière"

Expressions Régulières

Opérateurs

- Comment représenter
 - Un caractère quelconque : `.`
 - Une ou une infinité d'occurrences : `+`
 - Zéro ou une infinité d'occurrences : `*`
 - Un choix parmi un ensemble : `[<liste>]`
 - Tout sauf un certain caractère : `[^<caractère>]`
- Evidemment, on peut combiner les expressions
 - `.*` : Zéro ou une infinité de caractères quelconques
 - `a+b*` : Au moins un 'a' suivi de 0 ou une infinité de 'b'
 - `[ab]+` : Au moins un 'a' ou 'b' ou une infinité
 - etc.

Expressions Régulières

Utilisation

- Exemples

- **v.+** : Les chaînes contenant un v suivi de n'importe quelle suite de caractères
 - vandale
 - vestale
 - lavage
- **[vs].+** : Les chaînes contenant un 'v' ou un 's' suivi de n'importe quelle suite de caractères
 - vandale
 - voiture
 - descendre
 - sandale

Expressions Régulières

Utilisation

- Exemples

- **a.*a** : Les chaînes contenant deux 'a'
 - palabres
 - sandale
 - pascalle
 - cascade
- **[ps].*a.*a** : Les chaînes contenant un 'p' ou un 's' suivi d'une sous chaîne contenant deux 'a'
 - sandale
 - pascalle
 - apprentissage linéaire

Egrep

Exemple

- Reprenons le fichier de notes précédent

```
crepetna:Crepet, Nathalie:CREN1807750:92:87:88:54:70
yosnheat:Yos Nhean, Trakal:YOST19087603:84:73:70:50:73
benelaur:Benel, Aurelien:BENA80207700:84:73:89:45:100
soucypas:Soucy, Pascal:SOUP14067502:95:90:89:87:99
```

- On peut maintenant extraire la ligne où la dernière note est comprise entre 90 et 99
 - Une chaîne avec 7 ':' entre lesquels on peut avoir n'importe quoi et dont le dernier ':' est suivi d'un 9

```
biotope% egrep :.*:.*:.*:.*:.*:.*:9 notes.txt
soucypas:Soucy, Pascal:SOUP14067502:95:90:89:87:99
```

Expressions Régulières

Opérateurs

- Comment représenter
 - Un caractère compris entre 'a' et 'z' : **[a-z]**
 - Un caractère compris entre '0' et '9' : **[0-9]**
 - Le début d'une ligne : **^**
 - La fin d'une ligne : **\$**
 - Un choix entre deux chaînes : **|**
 - Et en combinant avec le choix
 - Caractère compris entre 'a' et 'z' ou 'A' et 'Z' : **[a-zA-Z]**
- Répétitions d'une occurrence
 - Exactement **2** répétitions de 'x' : **x\{2\}**
 - Entre **2** et **5** répétitions de 'x' : **x\{2, 5\}**
 - Au moins **2** répétitions de 'x' : **x\{2, \}**

Egrep

Exemple

- Encore le fameux fichier de notes

```
crepetna:Crepet, Nathalie:CREN1807750:92:87:88:54:70  
yosnheat:Yos Nhean, Trakal:YOST19087603:84:73:70:50:73  
benelaur:Benel, Aurelien:BENA80207700:84:73:89:45:100  
soucypas:Soucy, Pascal:SOUP14067502:95:90:89:87:99
```

- Aviez vous remarqué que le code permanent de Nathalie ne contient que 7 chiffres au lieu des 8 habituels ?
- On peut facilement identifier cela en utilisant une expression régulière

Egrep

Exemple

- Il suffit de *formaliser* le code permanent

```
crepetna:Crepet, Nathalie:CREN1807750:92:87:88:54:70
yosnheat:Yos Nhean, Trakal:YOST19087603:84:73:70:50:73
benelaur:Benel, Aurelien:BENA80207700:84:73:89:45:100
soucypas:Soucy, Pascal:SOUP14067502:95:90:89:87:99
```

- Un code permanent contient **4 majuscules** suivies de **8 chiffres** : `[A-Z]{4}[0-9]{8}`

```
biotope% egrep [A-Z]{4}[0-9]{8} notes.txt
yosnheat:Yos Nhean, Trakal:YOST19087603:84:73:70:50:73
benelaur:Benel, Aurelien:BENA80207700:84:73:89:45:100
soucypas:Soucy, Pascal:SOUP14067502:95:90:89:87:99
```

Egrep

Exemple

- On peut aussi ne faire apparaître que la ligne qui est fautive

```
crepetna:Crepet, Nathalie:CREN1807750:92:87:88:54:70  
yosnheat:Yos Nhean, Trakal:YOST19087603:84:73:70:50:73  
benelaur:Benel, Aurelien:BENA80207700:84:73:89:45:100  
soucypas:Soucy, Pascal:SOUP14067502:95:90:89:87:99
```

- Simplement en utilisant l'option **-v**

```
biotope% egrep -v [A-Z]\{4\} [0-9]\{8\} notes.txt  
crepetna:Crepet, Nathalie:CREN1807750:92:87:88:54:70
```

Egrep

Exemple

- Autre solution possible

```
crepetna:Crepet, Nathalie:CREN1807750:92:87:88:54:70
yosnheat:Yos Nhean, Trakal:YOST19087603:84:73:70:50:73
benelaur:Benel, Aurelien:BENA80207700:84:73:89:45:100
soucypas:Soucy, Pascal:SOUP14067502:95:90:89:87:99
```

- Finalement, un code permanent contient **12** caractères **lettres ou chiffres** : `[A-Z0-9]{12}`

```
biotope% egrep [A-Z0-9]{12} notes.txt
yosnheat:Yos Nhean, Trakal:YOST19087603:84:73:70:50:73
benelaur:Benel, Aurelien:BENA80207700:84:73:89:45:100
soucypas:Soucy, Pascal:SOUP14067502:95:90:89:87:99
```

Egrep

Exemple

- Autre solution pour la dernière note

```
crepetna:Crepet, Nathalie:CREN1807750:92:87:88:54:70
yosnheat:Yos Nhean, Trakal:YOST19087603:84:73:70:50:73
benelaur:Benel, Aurelien:BENA80207700:84:73:89:45:100
soucypas:Soucy, Pascal:SOUP14067502:95:90:89:87:99
```

- Une chaîne composée de 6 fois ':' suivi d'une chaîne quelconque puis d'un :9. Donc :

```
\(:.*\) \{6\}:9
```

```
biotope% egrep \(:.*\) \{6\}:9 notes.txt
soucypas:Soucy, Pascal:SOUP14067502:95:90:89:87:99
```

Egrep

Exemple

- Lignes commençant par un 'p' ou un 'c'

Juchée à l'extrémité nord-est de la table qui me sert de bureau, une lampe imbibe de sa lumière tamisée les quelques papiers privilégiés qui n'ont pas encore leur place parmi le fatras des classeurs habitant l'étagère à laquelle je tourne le dos avec une feinte obstination.

```
biotope% egrep ^[pc] lettre.txt  
p  
priviliégiés qui n'ont pas encore leur place dans le fatras des  
c  
classeurs habitant l'étagère à laquelle je tourne le dos avec une
```

Egrep

Exemple

- Lignes finissant par un 's'

Juchée à l'extrémité nord-est de la table qui me sert de bureau, une lampe imbibe de sa lumière tamisée les quelques papiers^s privilégiés qui n'ont pas encore leur place parmi le fatras de^s classeurs habitant l'étagère à laquelle je tourne le dos avec une feinte obstination.

```
biotope% egrep s$ lettre.txt  
une lampe imbibe de sa lumiere tamisee les quelques papierss  
privilégiés qui n'ont pas encore leur place dans le fatras des
```

Egrep

Exemple

- Recherche des lignes contenant 'leur' ou 'qui'

Juchée à l'extrémité nord-est de la table **qui** me sert de bureau, une lampe imbibe de sa lumière tamisée les quelques papiers privilégiés **qui** n'ont pas encore **leur** place parmi le fatras des classeurs habitant l'étagère à laquelle je tourne le dos avec une feinte obstination.

```
biotope% egrep \(\bleur\b|\bqui\b\) lettre.txt
```

```
Juchée à l'extrémité nord-est de la table qui me sert de bureau,  
privilégiés qui n'ont pas encore leur place parmi le fatras des
```

Egrep

Exemple Complexe

- Repérer les lignes contenant une date qui est exprimée par JJ/MM/AA ou bien JJ/MM/AAAA
- Formalisation des jours
 - Les jours sont compris entre 1 et 31
 - Le premier chiffre est compris entre 0 et 3
 - Le deuxième chiffre est compris entre 0 et 9
sauf dans les cas où le premier est un 0 ou bien un 3
 - On obtient alors : $\backslash(0[1-9]\backslash|[1-2][0-9]\backslash|3[0-1]\backslash)$
 - Jours 01 à 09 : $0[1-9]$
 - Jours 10 à 29 : $[1-2][0-9]$
 - Jours 30 et 31 : $3[0-1]$

Egrep

Exemple Complexe

- Formalisation des mois
 - Similairement, les mois sont compris entre 1 et 12
 - Le premier chiffre est compris entre 0 et 1
 - Le deuxième chiffre est compris entre 0 et 9
sauf dans les cas où le premier est un 0
 - On obtient alors : `\(0[1-9]\|1[0-2]\)`
 - Mois 01 à 09 : `0[1-9]`
 - Mois 10 à 12 : `1[0-2]`
- Formalisation des années
 - Une année contient simplement 2 ou 4 chiffres
 - On obtient alors : `\([0-9]{2}\|[0-9]{4}\)`

Egrep

Exemple Complexe

- Formalisation de la date
 - Concaténation des expressions formalisant les **jours**, **mois** et **années** avec un '/' entre elles :
- ```
\(0[1-9]\|[1-2][0-9]\|3[0-1]\)\^(0[1-9]\|1[0-2]\)\^([0-9]{2}\|[0-9]{4}\)
```
- Problème ?
    - Et oui, nous n'avons pas tenu compte du fait que certains mois ont 30 jours et d'autres 31.
    - Nous n'avons pas tenu compte du mois de février qui ne peut avoir que 28 ou 29 jours, ni du fait qu'il a 29 jours que lors des années bissextiles.
  - La solution vous est laissée en exercice ;-)