Systèmes informatiques

Olivier Lecarme

Licence Mathématiques-Informatique, Semestre 2

2005–2006

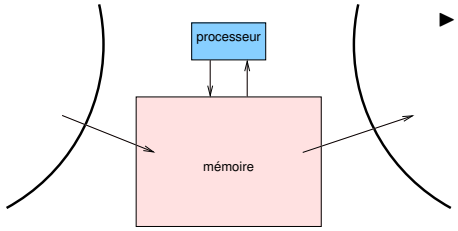
Premier cours : Introduction

Composants d'un système informatique

Un ***système informatique*** comprend trois composants :

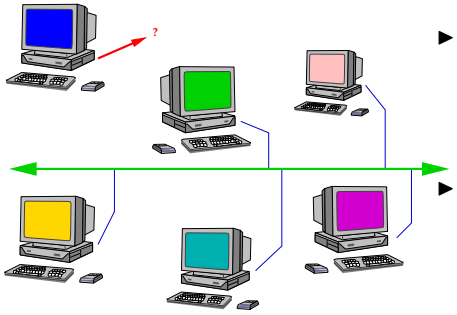
- l'ordinateur proprement dit
- les moyens de communication
- le système d'exploitation

L'ordinateur



- ***machine de von Neumann*** avec processeur, mémoire et organes de communication
- ensemble physique comprenant :
  - un processeur, partie active sans capacité
  - une mémoire principale, partie passive de grande capacité
  - des organes de communication entre processeur et mémoire (goulot d'étranglement des données)
  - des organes de communication vers l'extérieur (entrée et sortie)

Les moyens de communication



- Un ordinateur isolé n'est pas un système informatique, à la rigueur une machine de bureautique ou un gestionnaire d'agenda
- L'appartenance à un ***réseau*** est indispensable à l'échange d'informations avec d'autres systèmes informatiques

Le système d'exploitation

- partie logicielle du système informatique : plusieurs systèmes d'exploitation peuvent être utilisés sur le même ordinateur
- intermédiaire obligé entre utilisateur et matériel
- gère l'utilisation de la totalité des ressources : temps, mémoire, fichiers, communications, etc.
- fournit un ensemble de programmes utilitaires pour ce qu'il ne réalise pas lui-même

Historique

- avant 1960, ***utilisation individuelle*** et ***interactive*** de l'ordinateur :
  - réservé pour une certaine durée
  - essais et erreur, réflexion, réparation
  - beaucoup de temps inactif pour l'ordinateur
- premiers systèmes d'exploitation au début des années 1960 :
  - ***traitement par lots***
  - pas de connexion directe de l'utilisateur
  - travaux enchaînés en différé
  - amélioration de l'utilisation du temps de l'ordinateur
  - disparition de l'interactivité

Suite de l'historique

- ré-introduction de l'interactivité vers la fin des années 1960
  - ***accès multiple*** : plusieurs utilisateurs connectés en même temps sur la même machine
  - ***temps partagé*** : découpage du temps du processeur en périodes très courtes réparties entre les utilisateurs
  - encore amélioration de l'utilisation du temps de l'ordinateur
- début des années 1970, au moins un système différent par modèle de machine
- depuis, réduction énorme du nombre de modèles et du nombre de systèmes, apparition de l'***ordinateur personnel*** et de l'***écran graphique***

La situation actuelle

- ***MVS***, dernier avatar du système OS/360 d'IBM vers 1965, sur gros ordinateurs IBM
- ***VMS***, système propre au constructeur DEC (absorbé successivement par Compaq puis Hewlett-Packard), occupant une niche de sécurité
- ***Windows***, systèmes successifs dérivés de MS-DOS : domination écrasante sur le marché de l'ordinateur personnel
- ***Mac-OS***, système inséparable de son ordinateur
- ***Unix***, seul système non lié à un constructeur ou un fabricant de logiciel, seul fonctionnant sur tout ordinateur, sous diverses formes
- quelques autres systèmes peu répandus tels que OS/2, BeOS, etc.

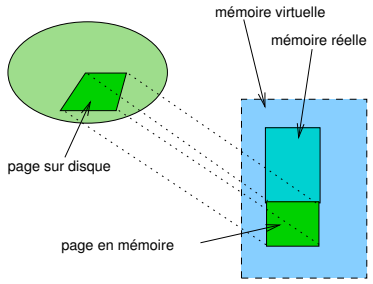
Comparaison de ces systèmes

- MS-DOS est **mono-tâche** et **mono-utilisateur**
- Windows est **multi-tâches** et normalement **mono-utilisateur**
- Mac-OS est **multi-tâches** et **mono-utilisateur** jusqu'à sa version X, où il s'ajoute un noyau de type Unix
- Unix est d'emblée **multi-tâches**, **multi-utilisateurs**, mais de plus **multi-sessions** et **multi-postes**
- Unix est le seul système non lié à un type de matériel

Gestion du processeur

- au niveau proche du ***matériel*** :
  - interruptions (événements extérieurs) et trappes (événements dans le programme en cours)
  - alternance entre mode système et mode utilisateur
  - masquer tout ce qui dépend du modèle de processeur
- au niveau proche de l'***utilisateur*** :
  - alternance entre ***processus*** indépendants
  - synchronisation par horloge ou par événements précis

Gestion de la mémoire



- espace fini, programmes en concurrence pour l'utiliser
- le partage du temps implique le partage de la mémoire
- récupération de l'espace inutilisé
- concept de ***mémoire virtuelle***, beaucoup plus grande que la mémoire réelle et représentée sur disques
- échanges entre mémoire réelle et mémoire virtuelle avec rapidité et fiabilité

Gestion du système de fichiers

- gestion de l'espace offert par les supports externes
- création et suppression de fichiers
- construction, lecture, déplacement
- moyens d'assez haut niveau pour masquer les caractéristiques du support physique

Gestion des organes périphériques

- boîtiers de disques, cassettes, CDs, DVDs, clés USB, mémoires externes, etc.
- hauts-parleurs, micros, caméras, etc.
- imprimantes, tablettes graphiques, etc.
- écran, clavier, souris, pointeur, manettes, etc.
- gestion d'un **grand nombre de protocoles de communication**, de niveaux très variés
- **sécurité, fiabilité**

Origines

- première version en **1969** : laboratoire de recherche de Bell Telephone, programmeur isolé (Kenneth Thompson)
- développement d'un programme de simulation d'exploration spatiale
- beaucoup d'idées tirées de ***Multics***, projet commun Bell Telephone – MIT – General Electric
- définition d'un **langage de programmation spécifique** pour programmer le système, tiré de BCPL : ***B***

Premiers développements

- version 5 en 1973, reprogrammée avec Dennis Ritchie avec le successeur de B : ***C***
- première distribution commercialisée en 1976 avec la version 6
- **distribution gratuite** aux universités avec le texte source
- système de licence basé sur le droit d'auteur, très protecteur et limitatif
- vente de la licence d'une société à une autre
- éclatement en 1977 en plusieurs versions indépendantes
- débuts de la version de l'Université de Californie à Berkeley (***BSD***), base de presque de toutes les versions sans problèmes de licences

Suite des développements

- commercialisation de la version 7 en 1978, la première véritablement **transportable**
- début de la version « System V » en 1983, à la suite de la scission entre les laboratoires Bell et la compagnie mère
- débuts du projet GNU en 1984, pour construire une version complète de Unix entièrement libre
- débuts des systèmes de Sun Microsystems en 1984 (SunOS), se poursuivant avec Solaris en 1992
- système OSF/1 de l'Open Software Foundation en 1990
- apparition des trois versions non commerciales du ***noyau*** : NetBSD et FreeBSD en 1992, Linux en 1993 ; toutes trois utilisent pour tout le reste les composants du projet GNU, qui représentent 90 % du total

Situation actuelle

- toutes les versions s'appuient sur System V, sur BSD ou sur les deux
- le degré de compatibilité permet de travailler en général sans souci des différences pour l'utilisateur ordinaire
- le nom Unix n'est plus une marque déposée
- parmi les versions qui sont du logiciel libre, GNU/Linux s'est le plus développé, et est maintenant soutenu par les grands constructeurs tels qu'IBM ou HP
- attention au nom Linux, qui n'est qu'un des noyaux utilisables par le système GNU
- utiliser de préférence le nom GNU/Linux, ou le nom Unix pour être générique

Distributions de GNU/Linux

- les ***versions commerciales*** n'ajoutent pas grand'chose
- elles ne sont utiles que dans le monde de l'entreprise, pour ajouter une possibilité de maintenance
- les ***versions gratuites*** sont de même niveau de qualité, mais demandent plus ou moins d'efforts à l'installateur
- certaines versions fonctionnent **sans installation** (***live CD***) mais ne peuvent servir que de test
- les versions les plus populaires ne sont pas forcément les meilleures
- l'installation est à la portée de toute personne soigneuse et moyennement avertie

Le concept d'utilisateur

- sur un ***ordinateur personnel banalisé*** :
  - l'utilisateur ne s'identifie pas
  - si l'ordinateur est en libre service, l'utilisateur doit transporter ses propres données
  - les fichiers présents sur l'ordinateur sont **à tout le monde** et à personne
- avec Unix :
  - chaque utilisateur doit s'identifier
  - ses fichiers lui appartiennent et il peut en autoriser ou interdire l'accès par les autres
  - ses données sont contenues dans son ***répertoire personnel***

La procédure de connexion

- l'utilisateur est désigné par son ***nom d'utilisateur***
- cette désignation est authentifiée par un ***mot de passe***
- la procédure de ***connexion*** vérifie ces deux informations, et n'accepte qu'un utilisateur duement enregistré
- en fin de séance de travail, l'utilisateur se ***déconnecte***, ce qui libère l'ordinateur
- l'ordinateur fonctionne en permanence, on ne l'arrête pas
- le mécanisme de ***partage des fichiers*** permet à l'utilisateur d'atteindre ses données depuis n'importe quel ordinateur relié au ***serveur de fichiers***
- le mécanisme de ***connexion à distance*** permet d'atteindre les fichiers même depuis un ordinateur lointain

Attributs de l'utilisateur

- ***nom d'utilisateur*** : pour les étudiants, identification attribué par le Bureau virtuel (universités de la région PACA)
- ***mot de passe*** : l'utilisateur le choisit lui-même, il est conservé sous forme cryptée ; c'est l'élément fondamental de la sécurité
- ***répertoire personnel*** : sa place dans la hiérarchie des fichiers est déterminée par l'administrateur ; l'utilisateur ne peut normalement placer ses fichiers qu'ici
- ***programme de démarrage*** : programme avec lequel l'utilisateur dialogue au démarrage de la connexion ; la fin de ce programme termine la connexion
- ***groupe(s) d'appartenance***

À propos du mot de passe

- c'est une **faute grave** de le communiquer à quelqu'un
- personne ne doit pouvoir le deviner
- vous devez donc **respecter quelques règles** :
  - ne l'écrivez nulle part
  - choisissez une chaîne facile à retenir, mais ne figurant dans aucun dictionnaire
  - incluez dans cette chaîne au moins un signe de ponctuation
  - incluez dans cette chaîne au moins un chiffre non évident (non pas `0` au lieu de `O` ou `1` au lieu de `l`)
  - mélangez majuscules et minuscules
  - n'utilisez jamais de caractères accentués
  - n'utilisez jamais le clavier auxiliaire
  - apprenez à taper votre mot de passe rapidement
  - changez-le s'il vous paraît non sûr

Le super-utilisateur

- l'administrateur a les privilèges du ***super-utilisateur***
- il peut :
  - lire et modifier tout fichier sur le système
  - enregistrer les nouveaux utilisateurs et initialiser leur environnement de travail
  - supprimer un utilisateur
  - installer ou mettre à jour des logiciels
  - surveiller le bon fonctionnement du système et corriger les défauts
  - effectuer les sauvegardes périodiques des programmes et données
- l'enseignant aura des privilèges limités qui lui permettront de suppléer l'administrateur

Le concept de processus

- un ***processus*** est un programme en cours d'exécution
- la plupart des commandes exécutent un programme, et donc lancent un processus
- le programme de démarrage correspond à un processus présent pendant toute la session
- des dizaines ou centaines de processus sont en fonctionnement à tout moment

Caractéristiques d'un processus

- identifié par un numéro entier
- associé à un ensemble d'informations, son ***image*** :
  - code du programme en cours d'exécution
  - données traitées par ce code
  - identification des fichiers en cours de traitement et leur état
  - répertoire courant
  - identité du propriétaire du processus
  - terminal associé
  - etc.

États des processus

- les processus sont dans différents états :
  - en attente d'un événement extérieur (action de l'utilisateur)
  - en attente d'exécution (tranche de temps)
  - en attente de l'arrivée d'une partie de la mémoire virtuelle
  - en exécution (un seul à la fois)
- en fait, tout se passe comme s'ils s'exécutaient simultanément :
  - je compile un programme
  - je reçois du courrier
  - mon voisin exécute un programme sur ma machine
  - etc.
- tout processus est lancé par un processus père :
  - arbre généalogique des processus
  - ***propriétaire réel*** (utilisateur qui l'a lancé)
  - ***propriétaire effectif*** (utilisateur donnant les droits du processus)

Le concept de fichier

- toutes les informations extérieures au processus sont des fichiers
- un fichier peut être associé au clavier, à l'écran, à l'imprimante, etc.
- quatre catégories de fichiers :
  - fichiers ordinaires
  - ***répertoires***
  - fichiers ***spéciaux***
  - ***liens symboliques***

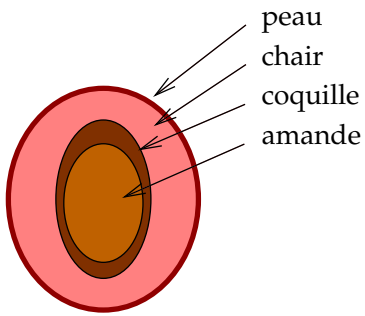
Fichiers et répertoires

- ***fichier ordinaire*** :
  - suite d'octets sans structure particulière
  - contient des données ou du programme
  - fichiers de texte structurés en lignes par une ***marque de fin***, de nombre ordinal 10 ; conversions nécessaires depuis et vers MS-DOS (Windows) ou Mac-OS
- ***répertoire*** :
  - nœud de la hiérarchie des fichiers
  - fichier de références à d'autres fichiers

Fichiers spéciaux et liens symboliques

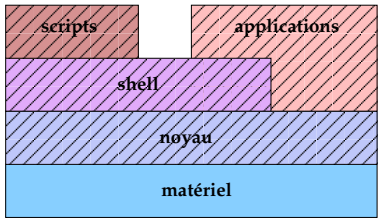
- ***fichier spécial*** :
  - fichier virtuel, représentation d'un organe périphérique
  - accès par un programme pilote, spécifique du périphérique
- ***lien symbolique*** :
  - fichier contenant la chaîne de caractères qui représente le nom d'un autre fichier
  - moyen de référence indirecte
  - moyen de construire un graphe quelconque et plus seulement une arborescence
- ***cheminement*** dans la hiérarchie :
  - le passage d'un répertoire à un autre se note /
  - le répertoire racine s'appelle seulement /

Structure générale



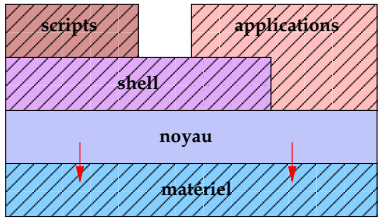
- une métaphore commune assimile le système à un fruit
- les couches concentriques représentent les composants de plus ou moins haut niveau
- la terminologie anglophone parle d'amande (***kernel***)
- la terminologie francophone préfère parler de ***noyau***

Meilleure représentation



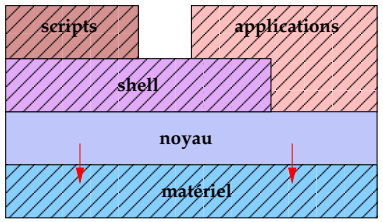
- le ***matériel*** est au niveau le plus bas :
  - le processeur
  - son langage propre (langage machine)

Le noyau



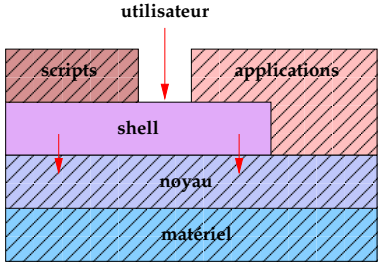
- le ***noyau*** de Unix masque le matériel
- on n'accède aux ressources du matériel que par les ***opérations primitives***
- cela comprend :
  - gestion du système de fichiers
  - partage du temps du processeur
  - partage de la mémoire
  - accès aux périphériques grâce aux ***pilotes***

Le noyau



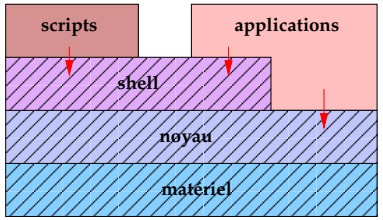
- accès aux primitives par instructions d'***appel au système***
- les primitives s'exécutent en ***mode privilégié***
- le reste des programmes est en ***mode utilisateur***
- les primitives permettent de :
  - lancer des processus
  - lire ou écrire sur des fichiers
  - obtenir de la place en mémoire
  - etc.

Le shell



- le programme de démarrage et interprète des commandes est le ***shell*** (coquille de l'amande en anglais)
- le shell est « à l'écoute » de l'utilisateur
- il interprète et exécute les commandes tapées
- quand le processus appelé par la commande se termine, le processus du shell redevient actif

Scripts et applications



- la plupart des programmes d'application communiquent avec le noyau sans passer par le shell
- le shell reconnaît un langage directement interprétable ou ***langage de script***
- on peut utiliser ce langage pour construire des ***scripts***

Types de shells existants

- le shell est **indépendant du noyau**
- il existe plusieurs shells plus ou moins perfectionnés :
  - `sh`, shell de Steven Bourne, conçu au début de Unix, le seul présent partout
  - `csh`, shell de la première version BSD, de syntaxe proche de celle de C
  - shells perfectionnés dérivés des précédents :
    - `ksh`, dérivé de `sh`
    - `tcsh`, dérivé de `csh`
    - `bash`, version améliorée de `sh`, défini par la Free Software Foundation
    - `zsh`, qui englobe tous les autres et que nous utiliserons

Le point de vue de l'utilisateur

- l'ordinateur lui-même est l'***hôte***
- les utilisateurs se connectent à un hôte donné à l'aide d'un ***terminal*** :
  - ***terminal alphanumérique***, aujourd'hui simulé par une fenêtre de l'outil XTERM (ou une de ses variantes)
  - ***terminal graphique***, sans possibilités de calcul locales, également appelé ***terminal X***
  - ***station de travail*** ou ***ordinateur personnel***, où les composants du terminal graphique sont indissociables de l'ordinateur
- l'utilisation d'un terminal alphanumérique simulé sur un terminal graphique ou une station de travail permet la connexion à un ***ordinateur distant***

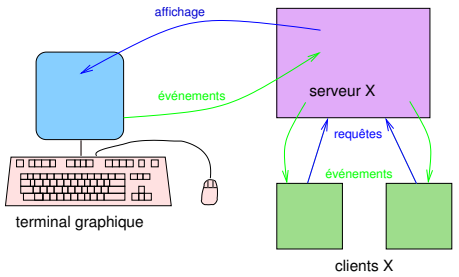
Interface alphanumérique

- l'***interface alphanumérique*** est celle d'une fenêtre de XTERM
- un seul processus peut communiquer avec l'interface (clavier et affichage, pas de souris)
- le processus attaché au terminal est ***interactif***, à l'écoute des commandes tapées par l'utilisateur
- commandes sous forme de suites de caractères
- le processus interactif est actif ***au premier plan***
- on peut lancer un processus détaché du terminal, qui passe ***en arrière-plan***

Interface graphique

- l'***interface graphique*** nécessite un ***système de fenêtrage***
- celui qu'on utilise avec Unix s'appelle X ou X11, produit construit au MIT et librement disponible
- idée fondamentale s'appuyant sur la ***relation client-serveur*** :
  - le ***serveur X*** gère le terminal graphique dans sa totalité :
    - affichage sur l'écran graphique
    - reconnaissance des signaux (***événements***) envoyés par le clavier et la souris
    - reconnaissance d'événements graphiques (passage de la souris dans une fenêtre, recouvrement d'une fenêtre par une autre, etc.)
  - les ***clients*** sont des programmes qui envoient au serveur des ***requêtes*** d'affichage et reçoivent la notification des événements qui les concernent

Le serveur X



- le système de fenêtrage est **indépendant des machines**
- le même serveur X peut satisfaire des requêtes provenant de **plusieurs machines**
- les clients ne savent pas comment fonctionne le serveur, et *vice-versa*

Le gestionnaire de fenêtres

- le système X n'impose **aucun comportement particulier** aux clients
- l'interface graphique n'est pas imposée (contrairement à Windows ou Mac-OS) :
  - décor des fenêtres
  - présence de menus déroulants ou surgissants
  - disposition et apparence de boutons ou icônes
  - traitement possible des fenêtres
  - manière de rendre une fenêtre active
  - manière de changer une fenêtre de place
  - etc.
- tout cela est réalisé par un client particulier, le ***gestionnaire de fenêtres***
- plus récemment on a ajouté par au-dessus un ***environnement de bureau***, qui codifie des comportements et des apparences

Connexion par interface graphique

- ce processus de connexion rappelle celui que vous connaissez avec Windows
- la différence majeure est qu'il est **obligatoire** et **personnel**
- l'écran d'accueil sera celui de GDM
- des options en bas d'écran permettent de choisir :
  - la langue de dialogue
  - le type de session : vous choisirez GNOME
  - l'arrêt du système : **vous ne le ferez jamais** !

Connexion par interface graphique (suite)

- en milieu d'écran apparaît la **fenêtre de dialogue**
- saisie du nom d'utilisateur, en minuscules et sans caractères accentués ni espaces : ce n'est pas votre nom !
- saisie du mot de passe, sans utiliser le clavier auxiliaire, et sans caractères accentués : il doit pouvoir être saisi sur tout clavier
- le système vérifie l'**adéquation des deux informations**
  - si elle est bonne, la session commence
  - si elle ne l'est pas, le système ne dit pas pourquoi (par sécurité)

Démarrage de la session

- une fois l'identification faite, le système fait démarrer :
  - le serveur X, c'est-à-dire le système de fenêtrage
  - le gestionnaire de fenêtres, ici Metacity
  - l'environnement de bureau, ici Gnome
  - un ou plusieurs tableaux de bord, qui rassemblent les moyens graphiques de communication
  - un ou plusieurs clients X, c'est-à-dire des applications d'utilisation fréquente
- tout ceci constitue la configuration de la session
- on peut la modifier pendant toute la session
- on peut conserver le nouvel état au moment de la **déconnexion**

Connexion par interface textuelle

- la **connexion par interface textuelle** sert dans de nombreuses circonstances :
  - changement d'identité sur la même machine
  - connexion à une **machine différente mais proche**
  - **connexion à distance**, éventuellement depuis un système autre que Unix
- on est dans une interface textuelle locale, typiquement une fenêtre Xterm
- depuis Windows on utilise un outil libre, Putty
- une commande permet de **lancer la connexion**, nous la verrons plus tard
- le système demande successivement le nom d'utilisateur et le mot de passe
- si l'identification est correcte, on se trouve sur la nouvelle machine mais toujours avec une **interface textuelle** : dialogue avec un shell

L'environnement graphique

- l'**environnement graphique** a un certain nombre de différences fondamentales par rapport à Windows :
  - on a normalement **plusieurs applications en cours**, donc aucune élargie à tout l'écran
  - on utilise en général **plusieurs bureaux**, c'est-à-dire plusieurs écrans virtuels
  - on utiliser en général un mécanisme qui rend immédiatement **active** la fenêtre dans laquelle est le pointeur
  - surtout, tout est **facilement paramétrable**

Le tableau de bord

- le **tableau de bord** a été paramétré au premier semestre pour la plupart d'entre vous
- on doit y trouver :
  - le changeur de bureau, qui permet de changer d'écran virtuel (mais des touches le permettent aussi)
  - quelques boutons de lancement des applications les plus fréquentes
  - la liste des fenêtres du bureau visible
  - d'autres éléments moins importants
  - un phylactère explicatif s'ouvre quand le pointeur passe dessus
- on le paramètre facilement par le menu accessible par le bouton 3 de la souris

Le gestionnaire de fenêtres

- le **gestionnaire de fenêtres** est METACITY
- c'est lui qui définit le décor des fenêtres
- quelques points nouveaux à connaître :
  - le bouton de gauche de la barre de titre permet de choisir le bureau sur lequel est la fenêtre
  - le bouton de fermeture de la fenêtre n'est pas à utiliser sans réflexion
  - celui qui élargit la fenêtre à tout l'écran est inutile dans 99 % des cas
  - au contraire, le plus important est celui qui minimise la fenêtre, c'est-à-dire la dissimule temporairement mais sans faire se terminer l'application correspondante

Xterm

- XTERM est un client graphique simulant un **terminal alphanumérique** de type VT100
- il existe beaucoup d'outils de même nature, nous choisissons le plus classique
- ETERM est aussi complet, avec des perfectionnements de présentation
- dans la fenêtre, un **shell** est à l'écoute :
  - **lit et interprète** les commandes saisies
  - **affiche les résultats**
  - signale qu'il est en attente par une **invite**
  - si on ne voit pas l'invite, c'est qu'un processus masque le shell
- élargir la fenêtre est presque toujours inutile
- l'allonger en hauteur est souvent intéressant

Mode de fonctionnement de Xterm

- le caractère saisi au clavier est envoyé au processus, qui en **envoie l'écho** dans la fenêtre
- le clavier comportant un nombre insuffisant de touches, on utilise des **combinaisons de touches** pour saisir certains caractères
- la touche `Ctrl` retranche 64 au code de la touche enfoncée en même temps :
  - `C-a` envoie le code 0 (zéro)
  - `C-g` correspond au signal auditif et annule en général ce qui est en cours
  - `C-j` est la fin de ligne
  - `C-m` est le retour, noté `RET`
- les touches en-dehors du clavier principal envoient des codes plus compliqués et mal normalisés
- le fonctionnement du clavier hors d'une interface graphique est donc plus primitif (problème de l'effacement)

Saisie des commandes

- la ligne saisie n'est envoyée au shell qu'après appui sur la touche `RET` (touche `Entrée`)
- cette touche peut être tapée **n'importe où** dans la ligne
- tant qu'elle n'est pas tapée on peut **corriger la ligne**
  - déplacements par les touches ← et →
  - `C-a` amène en début de ligne, `C-e` en fin de ligne
  - `C-w` **efface** le mot précédent, `C-k` tout ce qui suit le curseur, `C-u` toute la ligne
  - `C-c` **abandonne** la commande en cours de saisie

Dialogue avec XTERM

- la touche `Ctrl` et les trois boutons de la souris font surgir trois **menus de paramétrage** :
  - le bouton 1 ouvre le menu principal, intéressant surtout pour communiquer avec le processus en cours dans la fenêtre
  - le bouton 2 ouvre le menu d'options
  - le bouton 3 ouvre le menu des polices de caractères
- la barre de défilement permet de remonter dans les affichages précédents
- la molette de la souris également
- on peut **copier du texte** d'une fenêtre XTERM vers une autre :
  - bouton 1 de la souris en début de zone
  - bouton 3 en fin de zone : la zone est **sélectionnée** et copiée dans le tampon de sélection
  - bouton 2 pour coller la sélection **après le curseur**

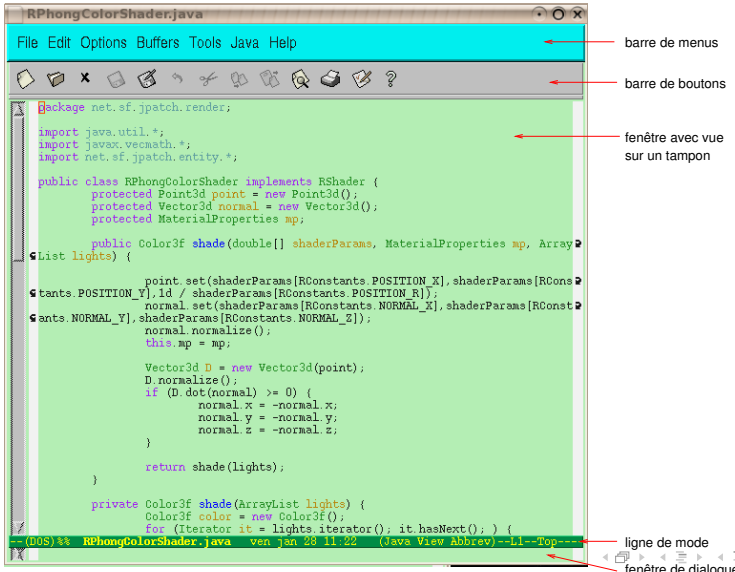
Édition de texte

- un **éditeur de texte** est un programme permettant de construire et modifier des fichiers de texte
- il existe des **éditeurs spécialisés**, intégrés dans une application
- il existe aussi des **éditeurs universels**, qui peuvent travailler sur **tout type de fichier**
- les éditeurs les plus simples ne font que cela : ED, EX, VI, VIM, XEDIT, NANO, etc.
- EMACS est le seul véritable **éditeur universel** :
  - véritable environnement de programmation
  - fonctionne sous tout système
  - logiciel libre
  - facile à étendre et adapter
  - très riche, donc long à apprendre

Utilisations d'Emacs

- fonctionnement sous mode graphique ou textuel
- édition de texte :
  - modes spécialisés
  - traitement automatique d'abréviations
  - présentation personnalisée
  - correcteur orthographique
- gestion de la hiérarchie des fichiers
- courrier électronique
- environnement de programmation
- aide en ligne sur EMACS et tous les logiciels de GNU
- agenda, calendrier
- personnalisation interactive
- etc.

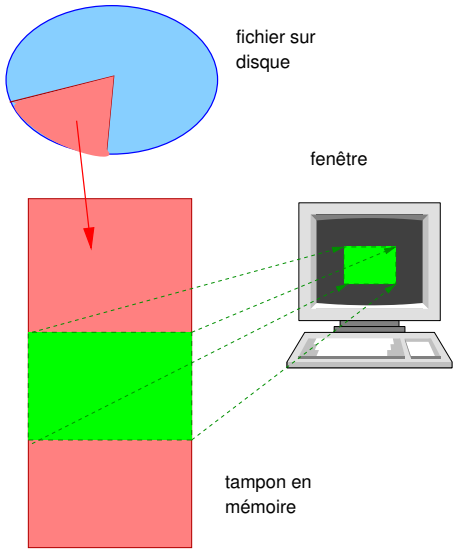
Un cadre d'Emacs



Organisation du cadre d'Emacs

- Emacs étant bien antérieur aux systèmes de fenêtrage, il a une terminologie différente
- Le ***cadre*** comporte cinq zones (quatre avec interface textuelle) :
  - barre de menus, dépendant du mode de la fenêtre, accessible également en mode textuel (`M-‘`)
  - barre de boutons, dépendant du mode et absente en mode textuel
  - fenêtre affichant une vue sur le tampon courant
  - ligne de mode, affichant des informations importantes sur le contenu de la fenêtre
  - fenêtre de dialogue, servant à l'affichage d'informations et à la saisie de commandes

Fichier et tampon



- Emacs ne travaille pas directement sur le fichier
- l'opération de ***visite*** copie le fichier en mémoire dans un ***tampon***
- la fenêtre affiche une partie du tampon
- les modifications sont faites **sur le tampon**
- l'opération de ***sauvegarde*** copie le tampon dans le fichier

Les commandes d'Emacs

- les ***commandes*** d'Emacs sont trop nombreuses pour être toutes mémorisées
- les moyens de les envoyer sont les suivantes :
  - **commandes abrégées** par combinaison de touches du clavier
  - **commandes textuelles** par utilisation du nom complet de la commande
  - **commandes par menu** à partir de la barre de menus
  - **commandes par touche spécialisée**
  - **commandes par bouton** à partir de la barre de boutons
- les deux premiers moyens sont **les plus rapides**, et les plus faciles à utiliser en mode textuel
- la souris est à utiliser **avec modération** car non ergonomique (obligation d'abandonner le clavier)

Principes des commandes abrégées

- toute commande abrégée correspond à une commande textuelle
- dans les modes normaux, les commandes abrégées nécessitent une touche `Ctrl` (notée `C-`) ou `Alt` (notée `M-`), ou les deux (`M-C-`)
- certaines combinaisons de touches servent de **préfixes** : `C-h`, `C-x` et `C-c`
- `M-x` est la ***commande universelle***, qui précède une commande textuelle
- toute commande peut être précédée d'un ***argument numérique*** qui en **modifie le comportement** :
  - `M-` suivie éventuellement du signe `-` et de chiffres qui représentent un nombre décimal
  - `C-u` qui sert d'**argument universel** et représente la valeur 4

Les tampons

- Emacs peut travailler sur **plusieurs tampons** en même temps
- à chaque tampon sont associées des **informations fondamentales** :
  - nom du tampon, déduit du nom du fichier
  - nom et emplacement du fichier associé
  - mode du tampon, c'est-à-dire ensemble des commandes et règles de comportement, déduites de la nature du texte
  - mode de codage du texte
  - mode de saisie des caractères absents du clavier
  - indicateur de modification du tampon par rapport au fichier
  - position du curseur, appelée le ***point***
  - position des marques éventuelles : la zone entre le point et la dernière marque est la ***région***

Propriété techniques principales de Unix

- usage général
- multi-utilisateurs
- multi-tâches
- interactif
- temps partagé
- indépendant des architectures de machines

Propriété philosophique fondamentale

Unix est presque entièrement détaché des contraintes commerciales

- dans le monde du PC on peut ignorer les distributions commerciales
- sur les autres ordinateurs, la plupart des composants peuvent être remplacés par des composants non commerciaux
- trois grands types de produits logiciels sans contraintes commerciales :
  - ***domaine public***, par exemple LaTeX (présentation de textes)
  - ***distribution gratuite***, par exemple X (système de fenêtrage), mais aussi OpenOffice ou Acrobat Reader
  - ***logiciel libre***, par exemple Emacs, mais aussi tout le système GNU

La *Free Software Foundation*

- fondée par Richard Stallman, auteur d'Emacs
- promouvoir la construction de ***logiciel libre*** :
  - disponible gratuitement, y compris par copie
  - fourni avec son texte source complet
  - tout produit dérivé doit à son tour être libre
- projet GNU ayant conduit à une reconstruction complète de tous les composants de Unix
- dans le monde du logiciel libre, la copie illégale de logiciel **n'existe pas**

Difficultés d'apprentissage de Unix

- l'apprentissage de Unix est long et difficile :
  - rechercher l'information
  - apprendre à se servir des outils de recherche
  - pas d'ordre logique d'apprentissage
  - nécessité d'être rapidement opérationnel
  - nécessité de beaucoup revenir sur la plupart des points
- évolution de Unix par accumulation, entraînant beaucoup de redondance
- beaucoup d'outils voisins mais incompatibles

Règles de bonne conduite

- le monde de Unix implique la vie en société
- ne faites pas à un autre ce que vous n'aimeriez pas qu'il vous fasse
- n'encombrez pas l'espace commun
- ne monopolisez pas les ressources communes
- ne laissez personne usurper votre identité
- n'utilisez jamais que votre propre identité
- **usurper l'identité d'un autre est une faute très grave**

Abondance des sigles et noms propres

- pour beaucoup de sigles (Unix, X, Emacs par exemple) la signification première a perdu tout intérêt
- retenir les sigles qui sont devenus des noms propres
- éviter les sigles inutiles
- tous les documents écrits fournis y feront attention

Abondance des termes techniques

- beaucoup de termes empruntés à l'anglais mais détournés de leur usage
- beaucoup de termes inventés, mais ressemblant à de l'anglais
- **évitez le jargon**, surtout si vous ne le comprenez pas
- cours et TP se limiteront à un vocabulaire francophone, sauf exceptions

Deux pratiques à maîtriser

- les messages d'erreur seront généralement en anglais
- vous devez apprendre à lire l'anglais technique
- il vous sera indispensable dans votre vie professionnelle
- vous devez apprendre la dactylographie
- les travaux pratiques ne doivent pas être ralentis par ces deux handicaps, donc améliorez-vous par vous-mêmes en-dehors des séances