

Systemes d'exploitation

Introduction

Sergiu Ivanov

`sergiu.ivanov@univ-evry.fr`

<https://www.ibisc.univ-evry.fr/~sivanov/fr/os-ueve.html>

Organisation du cours

Alternance CM/TD toutes les semaines

CM : slides + lecture du support du cours

- ▶ slides + support disponibles sur [E-media](#)
- ▶ support du cours : [lecture en Z](#)
- ▶ support du cours préparé par Jean-Marc Delosme

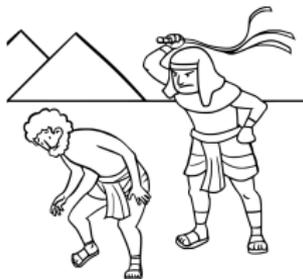
TD : sur papier

- ▶ sujets des TD seront disponibles sur [E-media](#)

Moyen de contrôle : [partiel](#)

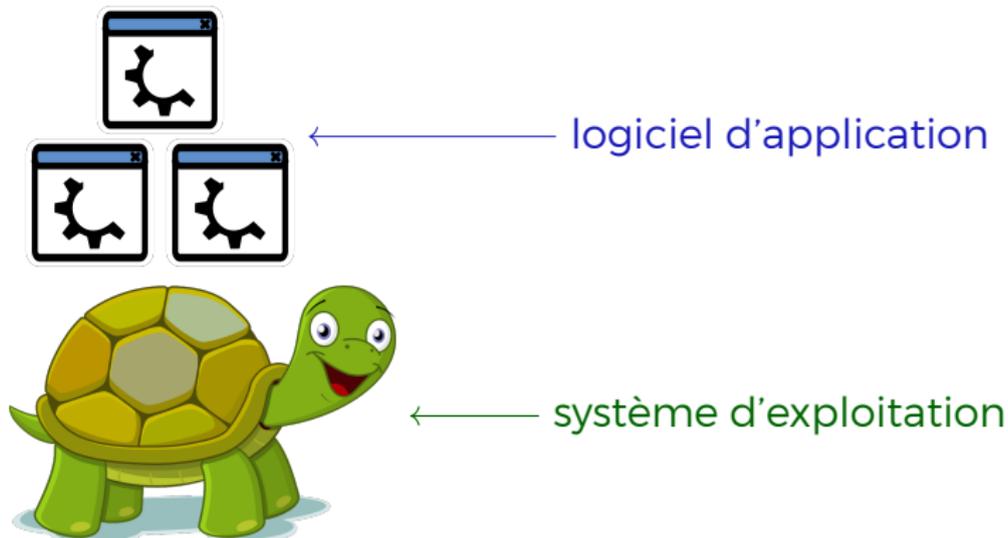
Toute question : sergiu.ivanov@univ-evry.fr

Qu'est-ce qu'un **système d'exploitation** ?



<https://openclipart.org/>

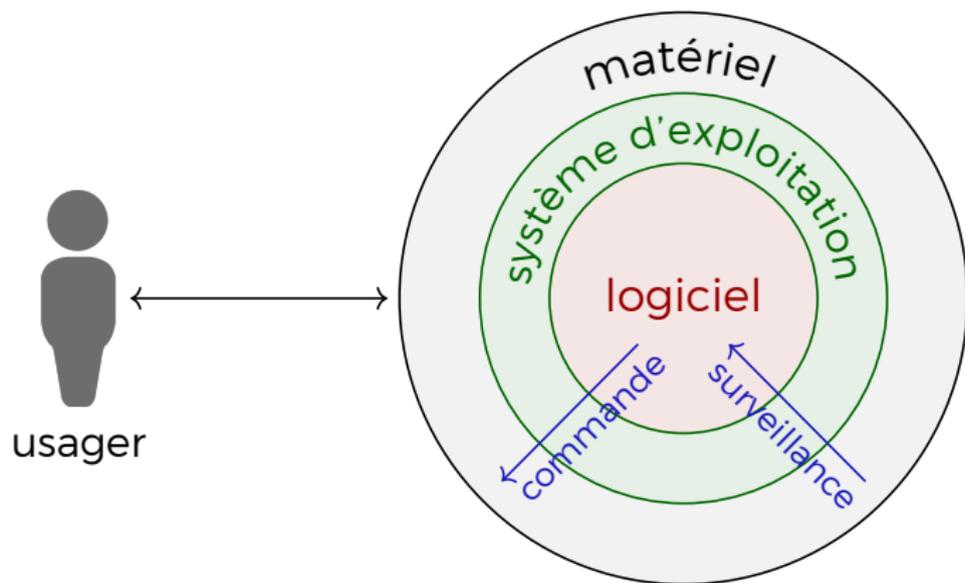
Système d'exploitation



<https://openclipart.org/>

Qu'est-ce qu'un **systeme informatique** ?

Structure d'un système informatique



Fonctions d'un système informatique

Fournir des **services** de

- ▶ gestion d'information
 - ▶ stockage, recherche, transmission, etc.
- ▶ préparation d'applications/programmes
- ▶ exécution des programmes

Qu'est-ce une **interface** ?

L'interface d'un système informatique

L'interface définit **un langage** qui permet aux usagers de communiquer avec le système.

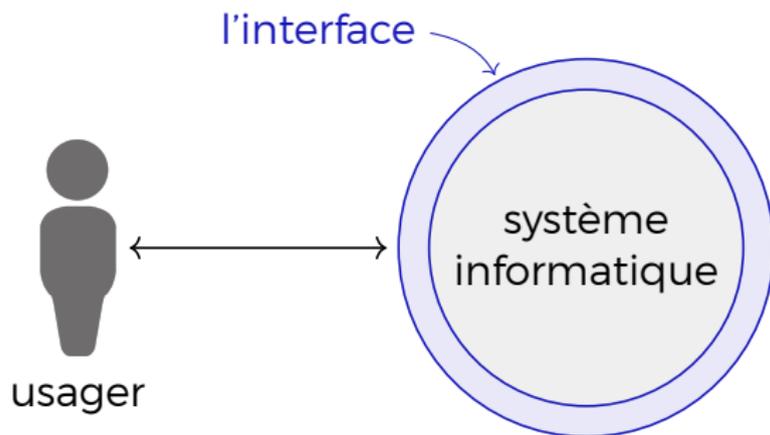
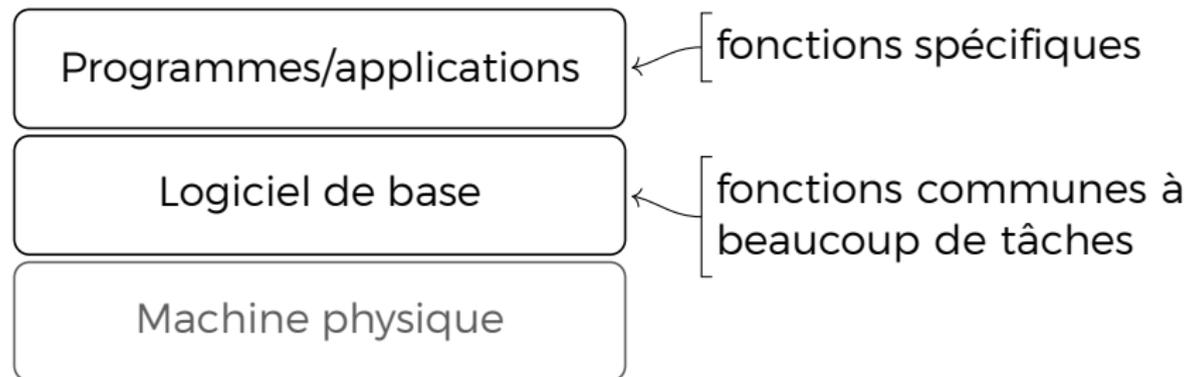
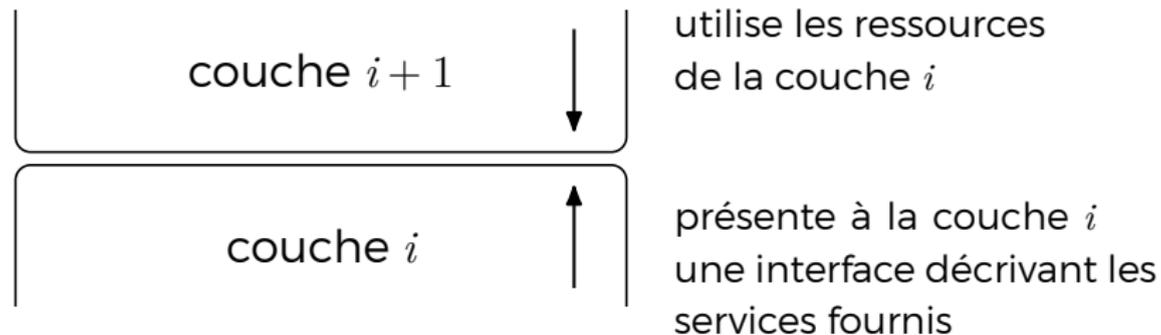


Schéma d'un système informatique





Les frontières entre les couches sont souvent imprécises.

Schéma d'un système informatique

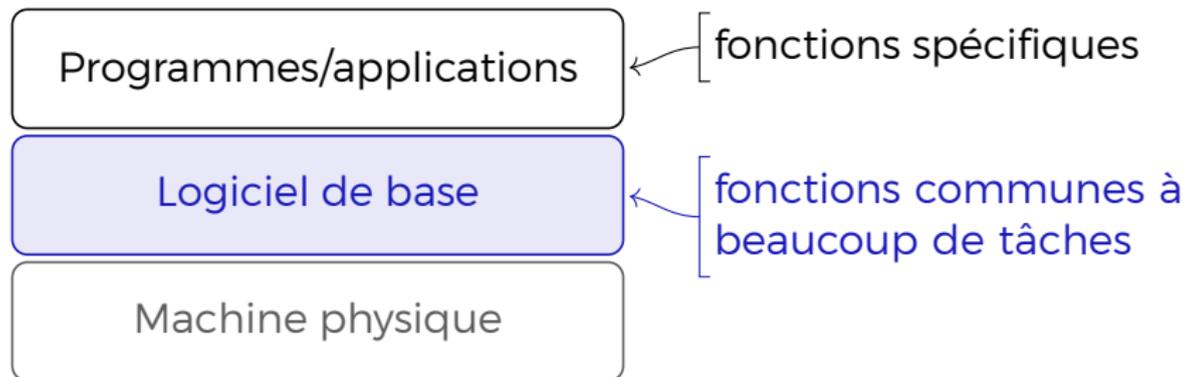
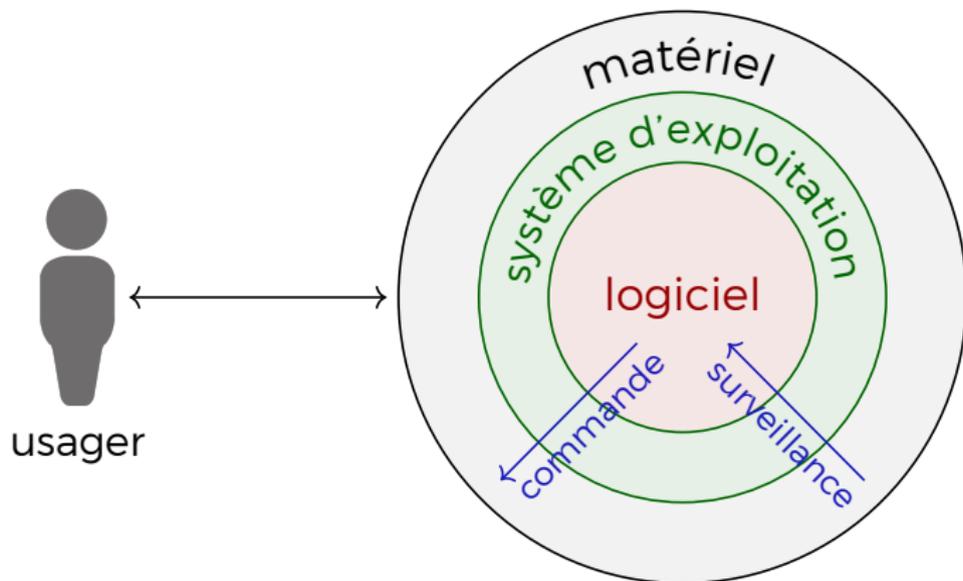


Schéma du logiciel de base





L'interface du système d'exploitation



Définit une **représentation abstraite** de la machine physique et du monde réel = **Machine abstraite**

Définition et réalisation d'une **machine abstraite** pour :

- ▶ la gestion de l'information
 - ▶ structuration, conservation, transfert
- ▶ l'exécution d'applications
 - ▶ séquentielle, parallèle, etc.
- ▶ des services divers
 - ▶ traitement de défaillances, mesure du temps, etc.

Gestion et partage de ressources :

- ▶ gestion de ressources physiques
 - ▶ mémoire, entrées, sorties, etc.
- ▶ partage et échange d'information entre usagers
- ▶ protection mutuelle des usagers
- ▶ services divers
 - ▶ facturation des ressources, mesures de performance/utilisation, etc.

Outline

1. Exemples de systèmes d'exploitation

Systèmes pour l'ordinateur individuel

Systèmes pour la commande de procédés industriels

Système à transactions

Systèmes en temps partagé

2. Évolution historique des SE, en bref

Outline

1. Exemples de systèmes d'exploitation

Systèmes pour l'ordinateur individuel

Systèmes pour la commande de procédés industriels

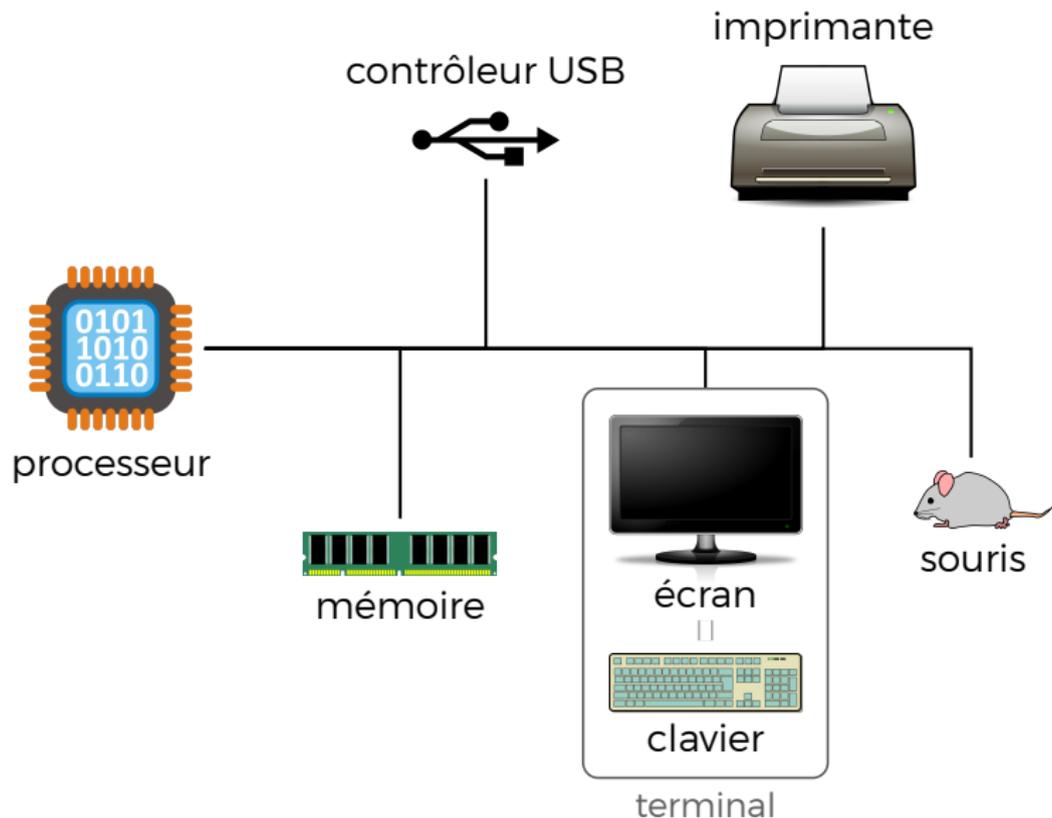
Système à transactions

Systèmes en temps partagé

2. Évolution historique des SE, en bref

Schéma de l'ordinateur individuel

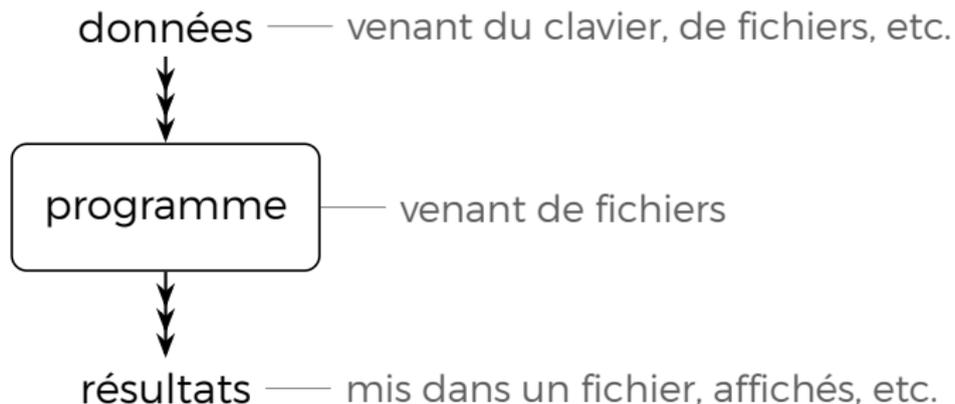
PC



<https://openclipart.org/>

Services attendus d'un SE individuel

- ▶ Gestion de fichiers
 - ▶ création, stockage, entrées-sorties ↔ fichiers
- ▶ Exécution de programmes



Un seul usager

Les SE individuels ont un seul usager.

- ▶ l'utilisateur a le contrôle total
- ▶ pas de partage de ressources entre des usagers

L'interface d'un SE individuel

L'interface côté usager : langage de commandes.

Une commande peut être donnée

- ▶ au clavier : [action] [paramètres]
- ▶ avec la souris (en cliquant les boutons affichés)

Actions typiques

- ▶ Composer un programme
 - ▶ via un éditeur de texte
- ▶ Exécuter le programme
 - ▶ données venant du clavier, de fichiers, etc.
- ▶ Modifier le programme
 - ▶ corrections, mises à jour, etc.
- ▶ Préparer la version finale du programme
 - ▶ retirer les données de débogage, etc.

Qualités attendues

- ▶ fiabilité
- ▶ efficacité
 - ▶ temps de réponse à l'utilisateur réduits
- ▶ simplicité d'utilisation
 - ▶ documentation facilement accessible
 - ▶ messages contextuels clairs
- ▶ facilité d'extension
 - ▶ installation de nouveaux utilitaires
 - ▶ installation de nouveaux périphériques

Outline

1. Exemples de systèmes d'exploitation

Systemes pour l'ordinateur individuel

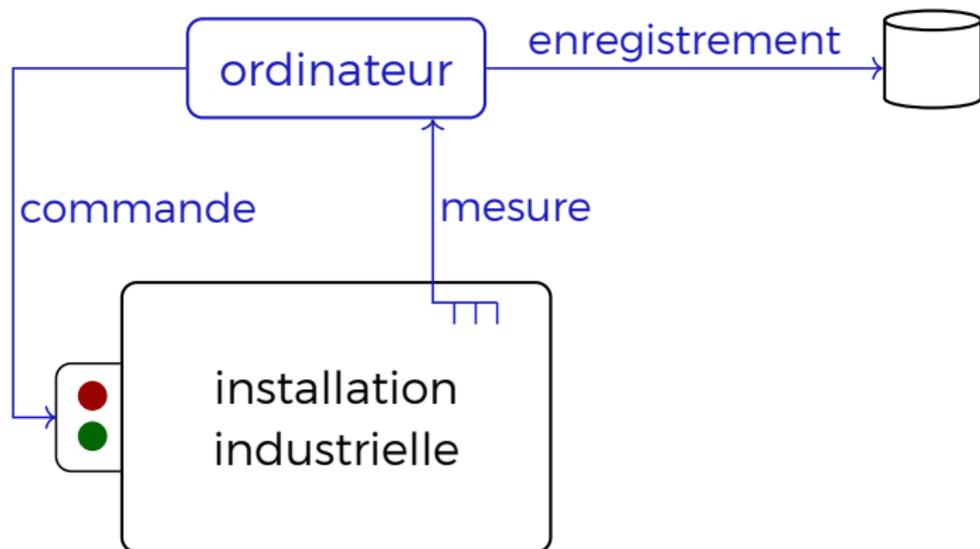
Systemes pour la commande de procedes industriels

Systeme a transactions

Systemes en temps partage

2. Evolution historique des SE, en bref

Commande de procédés industriels



Gestion de centrales électriques, surveillance médicale, pilotage des avions, commande de robots, etc.

période de
prélèvement
des mesures

T

t

temps de
traitement par
l'ordinateur

Contrainte: $T \geq t$

Contrainte de sécurité

La sécurité est **prioritaire**.

- ▶ détection fiable de **dépassements** de valeurs critiques
- ▶ le traitement de dépassement **interrompt** les opérateurs en cours

Fonctions des SE industriels

- ▶ Prise en compte du temps physique
 - ▶ traitement périodique
- ▶ Actions sur les organes externes
 - ▶ mesures, commandes, etc.
- ▶ Réactions aux évènement extérieurs
 - ▶ arrêt d'urgence, etc.
- ▶ Gestion de l'information
 - ▶ le journal de l'installation

Fiabilité

Le système doit assurer **en toute circonstance** la sécurité du dispositif qu'il commande, **même en cas de**

- ▶ panne de matériel
- ▶ accident
- ▶ erreur humaine

Outline

1. Exemples de systèmes d'exploitation

Systemes pour l'ordinateur individuel

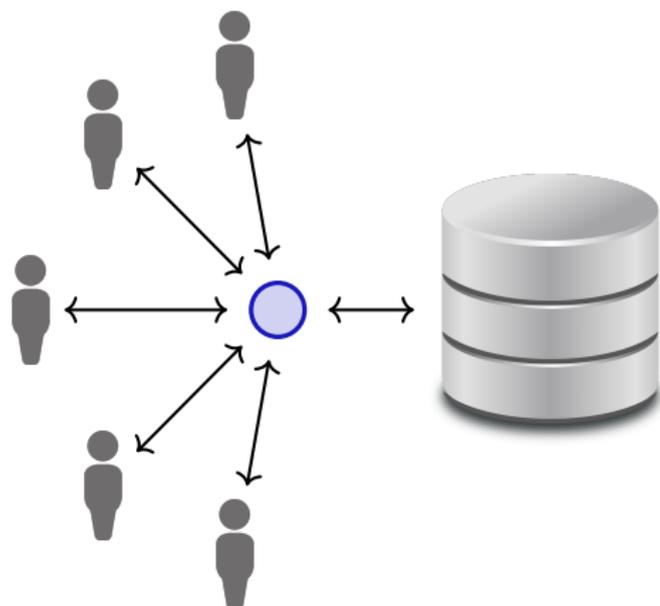
Systemes pour la commande de procedes industriels

Systeme a transactions

Systemes en temps partage

2. Evolution historique des SE, en bref

Système à transactions



Systèmes de réservation de places de train/avion, de gestion de comptes bancaires, etc.

Qualités attendues

- ▶ Disponibilité
- ▶ Fiabilité
- ▶ Tolérance aux pannes

L'exécution d'une transaction doit **préserver la cohérence** de la base de données.

Outline

1. Exemples de systèmes d'exploitation

Systèmes pour l'ordinateur individuel

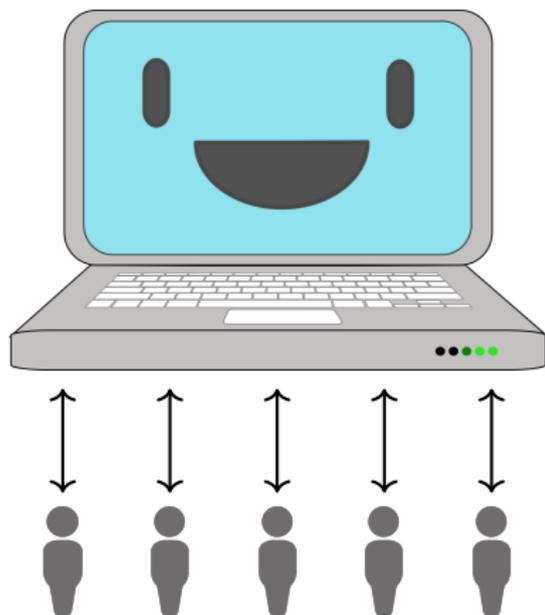
Systèmes pour la commande de procédés industriels

Système à transactions

Systèmes en temps partagé

2. Évolution historique des SE, en bref

Systemes en temps partage



- ▶ fournir des services égaux à un ensemble d'utilisateurs
- ▶ assurer la communication entre les utilisateurs et le partage d'information

Pour quoi faire ?

Défis des SE à temps partagé

- ▶ Définir la **machine abstraite** pour les usagers
- ▶ Partager uniformément les ressources physiques
 - ▶ processeurs, mémoires, dispositifs, etc.
- ▶ Partager l'information
 - ▶ selon les droits d'accès

Qualités attendues

- ▶ Disponibilité, fiabilité, sécurité
- ▶ Exploitation équitable des performances du matériel
- ▶ Bonne qualité des services offerts aux usagers
- ▶ Facilité de gestion des usagers

Outline

1. Exemples de systèmes d'exploitation

Systèmes pour l'ordinateur individuel

Systèmes pour la commande de procédés industriels

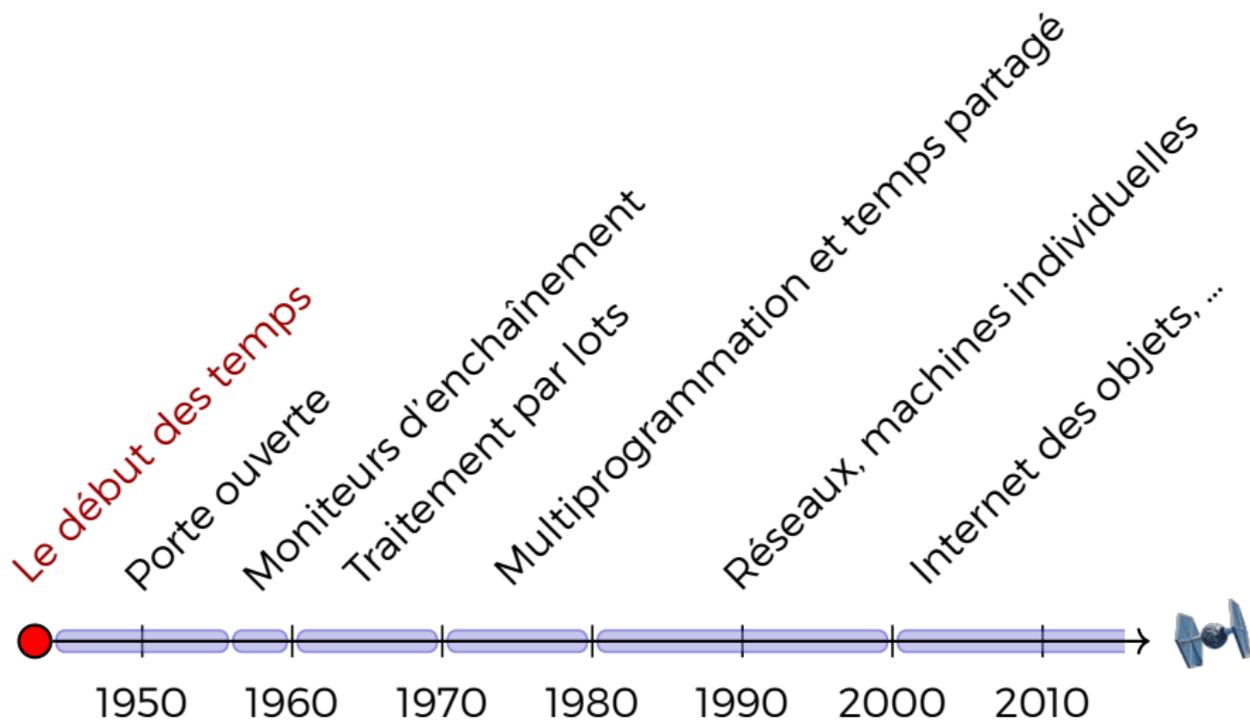
Système à transactions

Systèmes en temps partagé

2. Évolution historique des SE, en bref

Chronologie

approximative



<https://openclipart.org/>

Absence de système d'exploitation.

- ▶ créneaux d'utilisation par usager
- ▶ modification directe de la mémoire
- ▶ exécution pas à pas

Utilisation faible du matériel coûteux.

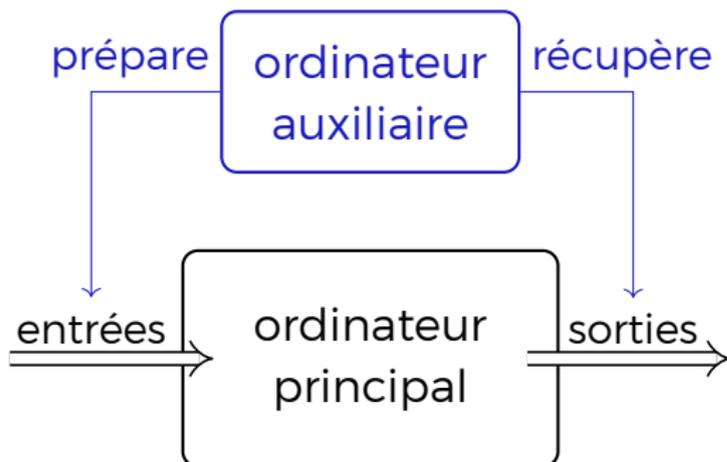
Moniteur = programme permettant d'exécuter en séquence une série de travaux préparés à l'avance

Fonction principale : gestion des ressources

- ▶ limiter le temps de processeur par tâche
- ▶ superviser les entrées-sorties
- ▶ protéger la mémoire du moniteur
 - ▶ mémoire système

Les périphériques d'entrée-sortie sont **lents**.

- ▶ saisie par l'utilisateur, affichage, etc.



L'ordinateur principal traite les **lots** préparés par l'**ordinateur auxiliaire** \implies meilleure exploitation

- ▶ Entrées-sorties tamponnées
 - ▶ processeurs spécialisés dans le transfert de l'information
- ▶ Multiprogrammation
 - ▶ partage des ressources entre plusieurs travaux
- ▶ Temps partagé
 - ▶ machine $\frac{1}{n}$ usagers

Multiprogrammation

La mémoire contient **plusieurs tâches** qui alternent entre **calcul** et **entrées-sorties**.

Multiprogrammation =

- ▶ une **tâche en attente** d'entrées libère l'unité centrale pour une autre tâche
- ▶ une **tâche active** peut être **mise en attente** pour libérer l'unité centrale pour une autre tâche

-
- + **Meilleure utilisation** des ressources
 - + **Meilleure expérience** pour l'utilisateur
 - **Système plus complexe**
 - ▶ protection mutuelle des tâches, partage des ressources

Progrès technologiques \implies microprocesseurs

Prix de composants \downarrow \implies ordinateurs individuels

Développement de réseaux \implies réseaux d'ordinateurs

Les défis de gestion de très grands réseaux d'ordinateurs
à performances élevés \implies nouvelles architectures de SE

Composants à prix très bas \implies ordinateurs (microcontrôleurs) dans les **dispositifs ménagers**

Réseaux sans fil \implies **beaucoup** de dispositifs **connectés**



Architectures de SE dédiés = $f(\text{organisation du matériel})$



- + Informatique omniprésente
- Défis de communication et de sécurité

Évolution cumulative !

Les apports de l'évolution **s'accumulent** !

- ▶ ils ne s'excluent donc pas !

Les éléments suivants sont **omniprésents** :

- ▶ entrées-sorties tamponnées
- ▶ multiprogrammation
- ▶ réseaux d'ordinateurs

On **retrouve encore** de nos jours

- ▶ le traitement par lots systèmes à haute fiabilité
- ▶ les systèmes à temps partagé superordinateurs
- ▶ l'exploitation porte ouverte systèmes embarqués