

## Une brève histoire de l'informatique

Sacha Krakowiak

Université de Grenoble & Aconit

## C'est quoi, l'informatique ?

### ❖ Quatre facettes

Une science

Science de l'artificiel ...

... mais pas seulement

Une technique et une industrie

Matériel, logiciel, services

Des applications

Dont le champ est croissant

Un impact sociétal

### ❖ Quatre concepts



### ❖ Une méthode

La modélisation et l'abstraction

Un va-et-vient théorie-pratique

Information

Un réducteur d'incertitude

Une représentation codée

La base de la communication

Algorithme

La notion clé !

d'Euclide à Turing

Machine

Réelle ou virtuelle

Langage

Exprime un algorithme  
pour une machine

## C'est quoi, l'information ?

### ❖ Un «réducteur d'incertitude»

### ❖ Je joue à pile ou face

avant de lancer : je ne sais rien

la pièce retombe : j'ai l'information

c'est l'information la plus élémentaire (2 possibilités)

ça s'appelle un *bit* (et plutôt que pile ou face, il est commode de dire 0 ou 1)

### ❖ Je lance la pièce deux fois ...

j'ai 4 résultats possibles : pile-pile, pile-face, face-pile, face-face

j'ai besoin de 2 bits : 00, 01, 10, 11

Mais en numération binaire (base 2), ça veut dire 0, 1, 2, 3

$0 \times 2 + 0 = 0$  ;  $0 \times 2 + 1 = 1$  ;  $1 \times 2 + 0 = 2$  ;  $1 \times 2 + 1 = 3$

... et on continue : une photo, un DVD, un livre : des centaines de millions de bits !

## C'est quoi, un algorithme ?

### ❖ Un procédé constructif ...

... qui réalise un objectif fixé, en un nombre fini d'étapes

... en combinant des opérations élémentaires connues

### ❖ Des exemples

les 4 opérations de l'arithmétique

la recherche d'un mot dans un texte

le calcul de la position par un navigateur GPS

une recette de cuisine

### ❖ Une notion finalement assez simple ?

pas si simple que ça !

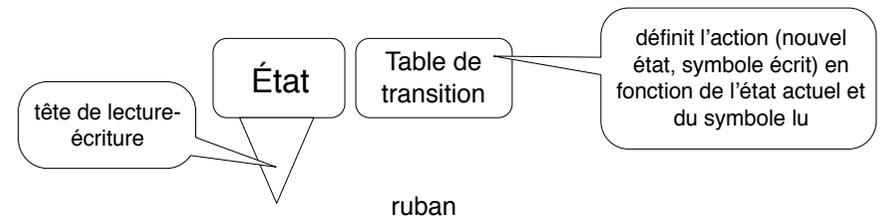
la définition précise de ce qu'est un algorithme marque le début de l'histoire de l'informatique



## Retour sur les algorithmes

- ❖ Que peut-on «effectivement» calculer ?
- ❖ De nombreuses propositions dans les années 1930
  - les fonctions récursives (Kleene)
  - le lambda-calcul (Church)
  - la machine de Turing
  - ...
- ❖ Tous ces schémas, définis indépendamment, sont équivalents !
  - D'où la thèse de Church-Turing (indémontrable, mais jusqu'ici non contredite) :
  - Un algorithme, c'est ce qui est réalisable par une machine de Turing (ou tout schéma équivalent)

## La machine de Turing (1936)



Alan Turing  
(1912-1954)



National Portrait Gallery

C'est une machine *abstraite* (elle serait monstrueusement inefficace !)

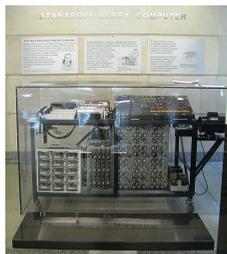
Il existe une machine *universelle* (capable de simuler n'importe quelle machine)

Il n'y a pas d'algorithme général pour dire si une machine va s'arrêter ou tourner indéfiniment

## Les premiers calculateurs (1)

- ❖ Atanasoff-Berry Computer (Iowa State Univ.), 1939
  - Spécialisée (systèmes linéaires), non programmable
  - Machine à tubes, arithmétique binaire, non Turing-complète

ABC



Wikimedia Commons CC BY-SA 3.0 by Manopp

Z3



Wikimedia Commons CC BY-SA 3.0 by Venusianer

- ❖ Zuse Z3, 1941
  - Électro-mécanique (relais), programme sur bande perforée
  - Arithmétique binaire, Turing-complète

## Les premiers calculateurs (2)

- ❖ Mark-1 (Harvard-IBM, Aiken), 1944
  - Arithmétique décimale, programme sur cartes perforées. Électro-mécanique, non Turing-complète

Mark-1



Wikimedia Commons CC BY-SA 3.0 by Topory

Colossus



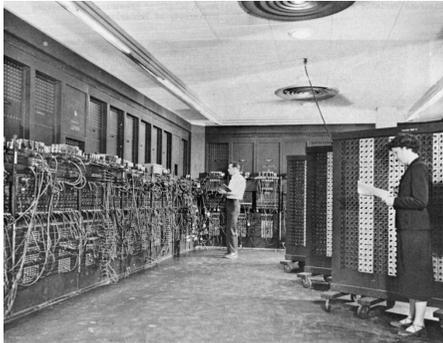
Wikimedia Commons CC BY-SA 3.0 by MaltaGC

- ❖ Colossus (Bletchley Park, Flowers), 1943-44
  - Électronique (tubes), binaire. Spécialisée dans le décryptage (cassage du code Lorenz). Non Turing-complète

## L'ENIAC

- ❖ Le premier ordinateur entièrement électronique, Univ. de Pennsylvanie (Eckert-Mauchly, 1946)

Arithmétique décimale, programmation par câblage  
19 000 tubes, 30 tonnes, 150 kw. Turing-complète



J. Presper Eckert

Images: US Dept of Energy



John W. Mauchly

## Le modèle «de von Neumann»

- ❖ von Neumann (avec Eckert et Mauchly, 1945)

- ❖ Traits essentiels

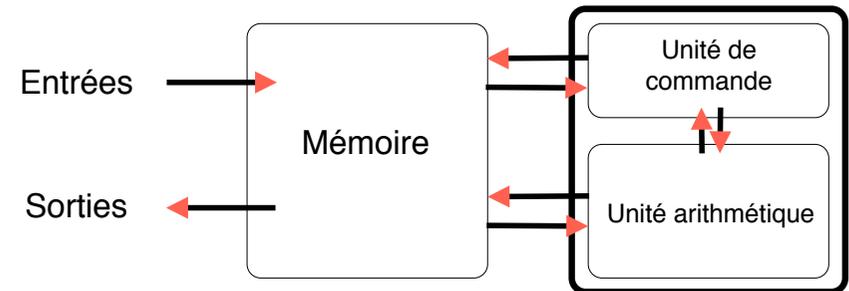
Séparation mémoire-unité centrale

Séparation calcul-séquencement

La mémoire contient les données et les instructions



Unité centrale (ou processeur)



## Le problème de la mémoire

- ❖ Initialement (avant 1950)

### Flip-Flops

Utilisés dans Colossus

Peu adaptés à une grande mémoire

### Ligne à retard (J. Presper Eckert)

Circulation d'impulsions sonores dans un tube de mercure

Accès séquentiel ; doivent être régénérées

### Tube cathodique (Freddie Williams, Tom Kilburn)

Tableau de charges électriques (bits) sur un tube

Accès aléatoire ; doit être régénéré

- ❖ Plus tard

Tambours magnétiques (~ 1952)

Tores de ferrite (~ 1953)

Transistors (~1958)

Circuits intégrés (~ 1971)

Le modèle de von Neumann prévoit une mémoire distincte de l'unité centrale. La technique reste à inventer...

### Mémoires secondaires

Bandes magnétiques ~1950)

Tambours magnétiques (~ 1952)

Disques magnétiques (~ 1955)

...

## La machine à programme enregistré : développement

- ❖ Les efforts initiaux

Les «Moore School lectures» (1946), diffusion des idées

À la suite d'ENIAC : EDVAC (1949-1951)

La machine de Princeton IAS (von Neumann, Bigelow)

- ❖ Les premières réalisations

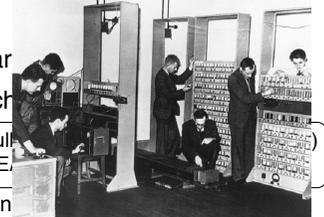
Mark-1, Université de Manchester (Williams, Kilburn, 1949)

EDSAC, Université de Cambridge (Wilkes, 1949)



Managers commerciaux de Manchester Mark-1  
EDVAC (Eckert-Mauchly)

En France 1952 : Bull  
1954 : SEAC  
EDSAC  
mémoire à tambour magnétique



## Où en est-on au début des années 50 ?

### ❖ Les avancées

Turing (et les autres) : définition et limites de l'algorithme, la machine universelle

von Neumann (et les autres): le modèle de base, toujours valable

les premiers ordinateurs commerciaux

### ❖ Les limites

une «technologie» déficiente, les tubes (peu fiables) ; les mémoires à lignes à retard (complexes, réglage délicat)

une programmation à très bas niveau, sans outils conceptuels, fastidieuse et sujette aux erreurs

un champ restreint d'applications

l'informatique est encore (pour longtemps) une affaire de professionnels

## Du binaire à l'assembleur

```
mov di,[wRootEntries] ;di=number of entries in the root dir.
mov cl,4 ;shift four bits right (/16)
shr di,cl ;di=di*32/512 (assumed sector size)
add [wResSecs],di ;so, di=number of root dir. sectors
;fix up the data sector variable
;now it points to the first cluster

read_root:
push di ;preserve the counter

push es ;preserve es
mov cl,1 ;one sector
call read_linear ;read sector (ax gets increased)
pop es ;restore es

mov di,bx ;es:di->RootDir
mov cl,512/32 ;16 entries, sector size of 512 assumed
;(32=size of one entry)

read_root_nextfile:
push cx ;preserve counter

mov cl,11 ;file name length
mov si,sFileName ;ds:di->OS file name
repe cmpsb ;compare
je short found_system ;jump if equal (file found)
add di,cx ;increase si by the rest of cx
add di,21 ;and fix up by 21 bytes - move to the
;next entry
pop cx ;restore counter
loop read_root_nextfile ;repeat for next entry
```

un morceau de programme en assembleur x86

## Qu'est ce qu'un langage de programmation ?

### ❖ Motivation

Un programme en assembleur décrit un algorithme en termes de *ce que sait faire la machine*

On souhaite une expression en termes de *ce que veut l'utilisateur*

### ❖ Qu'attend-on d'un langage de programmation ?

«Haut niveau» d'expression

proche du domaine d'application

indépendant de la machine d'exécution

Puissance d'expression

capable de décrire tous les traitements envisagés

Facilité d'apprentissage et commodité d'utilisation

Rigueur sémantique

Sûreté

réduit les risques d'erreur

si  $Z > 0$   
{ $X[3] = N+2$ }

trier(Tableau)

## Le premier langage de programmation

### ❖ Fortran (*Formula Translation*)

Développé chez IBM par l'équipe de John Backus (1954-57)

Langage orienté vers le calcul scientifique

Évaluation de formules

Définition de fonctions réutilisables

Toujours utilisé aujourd'hui (mais a beaucoup évolué)

### ❖ Une révolution dans la pratique de la programmation

Ouvre la programmation aux utilisateurs ...

... sans sacrifier l'efficacité

### ❖ Les limites du premier Fortran

Manque de rigueur dans la définition

Manque de sûreté : mauvaise détection des erreurs

## Les langages influents des années 60

- ❖ **Fortran** (1954) - John Backus  
Le premier langage de haut niveau, toujours utilisé
- ❖ **Algol 60** (1958-61) - comité de scientifiques (Peter Naur, rapporteur)  
Définition d'un langage sur une base rigoureuse  
objectif atteint, malgré définition par un comité ...  
Peu utilisé mais très forte influence sur les langages futurs
- ❖ **Lisp** (1958) - John Mc Carthy, MIT  
Premier langage «fonctionnel»  
Structure de base, pour programmes et données : la *liste*  
Le langage favori pour l'intelligence artificielle
- ❖ **Cobol** (1959) - comité d'industriels  
Langage pour applications de «gestion»  
Toujours utilisé, malgré son manque d'élégance
- ❖ **Basic** (1963) - John Kemeny  
Langage simple (dérivé de Fortran, interactif)  
Adapté au temps partagé

## Les pionniers des langages de programmation



**John Backus**  
1924 - 2007

Fortran

CC-BY-SA 2.0 by null0



**John McCarthy**  
1927 - 2011

Lisp



Cobol

**Grace Hopper**  
1906 - 1992



Algol 60

CC-BY-SA 3.0 by Erikj

**Peter Naur**  
1928 - ...

## Naissance des systèmes d'exploitation

- ❖ **Le problème**  
Gérer l'accès des utilisateurs à la machine  
Assurer la bonne exécution des programmes

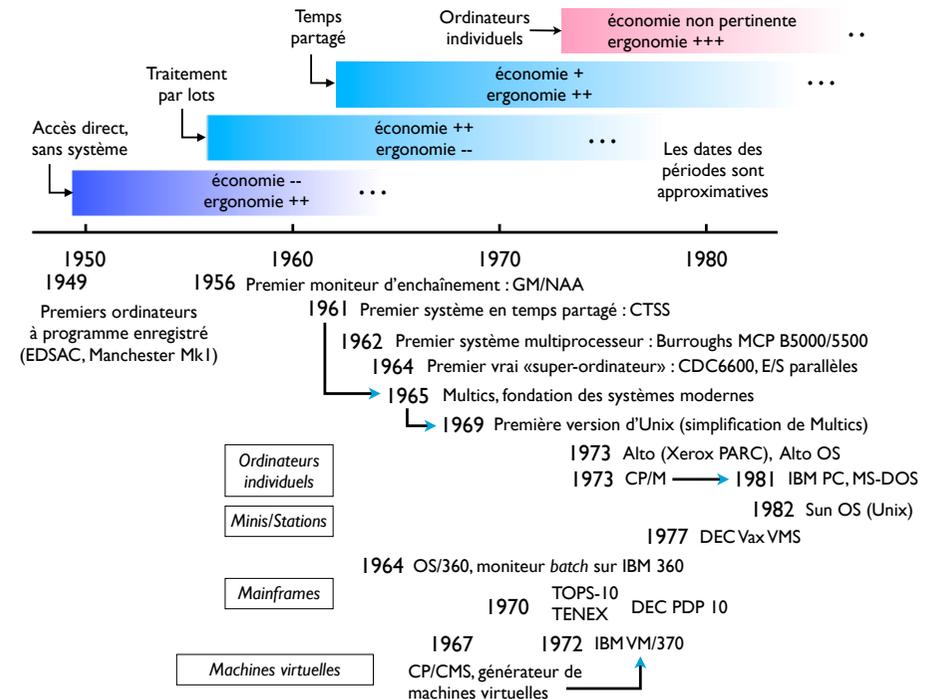
### Deux exigences qui s'opposent ...

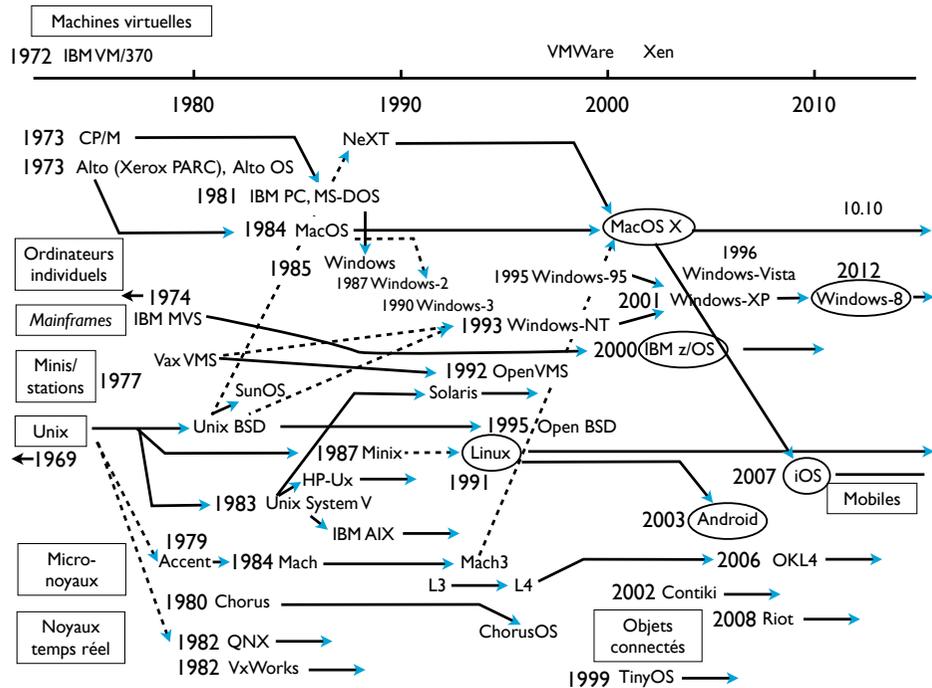
- l'utilisation optimale du matériel
- le confort (et l'efficacité) des utilisateurs

- ❖ **Les origines**  
Utilisation individuelle sur réservation  
Usage de «sous-programmes» pour les entrées-sorties

Traitement par lots  
La file d'attente physique devient virtuelle  
Le «moniteur» gère les ressources et les programmes  
Les premiers systèmes d'exploitation viennent des clients !

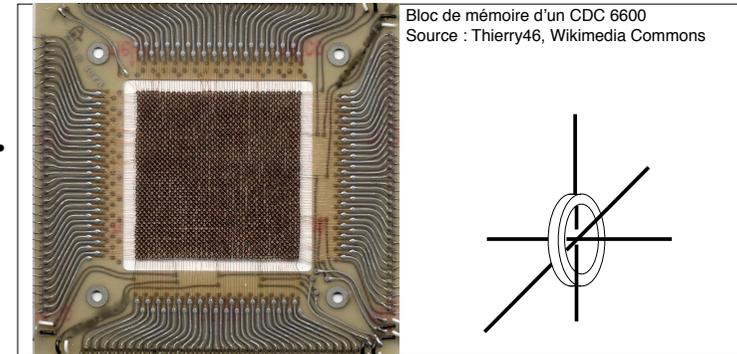
- ❖ **De nouveaux métiers pour l'exploitation**  
Opérateur, pupitreux, perfo-vérif.





## Nouvelles générations d'ordinateurs (1)

- ❖ Des tubes aux transistors et aux circuits intégrés
  - Inventions : 1947, transistor ; fin années 50 : circuits intégrés
  - À partir de 1955
    - circuits : les transistors remplacent les tubes
    - mémoires : les tores de ferrite remplacent les tambours magnétiques



## Nouvelles générations d'ordinateurs (2)

- ❖ La première «famille» : la série IBM 360 (1964)
  - Une ligne d'ordinateurs compatibles grâce à la microprogrammation
  - Des avancées techniques
    - Le cache (IBM 360/85, 1968)
    - Les machines virtuelles (IBM 360/67, 1966)
  - Un système d'exploitation gros et complexe (OS/360)
    - Traitement par lots, premier système utilisant les disques
  - Un succès commercial
- ❖ Le premier «super-calculateur» : CDC 6600
  - Seymour Cray (1964)
  - Processeurs spécialisés, traitement parallèle
- ❖ Le premier mini-ordinateur : DEC PDP-8 (1965)



## Années 1950-60 : les premières applications (1)

- ❖ Le calcul scientifique
  - L'objectif : trouver des solutions numériques à des problèmes de nature mathématique
  - Les bases : calcul numérique, puis analyse numérique
    - Une discipline en très fort développement
  - Les domaines d'application
    - Recherche : physique, chimie, mécanique, astrophysique, ...
      - Médecine et biologie peu impliquées au début
    - Sciences de l'ingénieur : électronique, électrotechnique, génie civil, nucléaire, spatial ...
      - Premier outil graphique : Sketchpad (Ivan Sutherland, 1961)
- ❖ La commande de procédés industriels
  - L'objectif : piloter un processus évolutif pour atteindre un but fixé
  - Les bases : l'automatique
  - Les applications : processus de fabrication, transports

## Années 1950-60 : les premières applications (2)

### ❖ Les applications de gestion

Objectif : automatiser les opérations de comptabilité, gestion des stocks, gestion de personnel, ...

Bases :

- analyse fonctionnelle (le «quoi ?» : besoins et contraintes)
- analyse organique (le «comment ?» : modèle de réalisation)

Limites

- méthodes encore peu évoluées, difficulté sous-estimée
- pas de notion globale de «système d'information»

### ❖ La recherche opérationnelle

Objectif : aide à la décision, notamment en environnement incertain

Bases : mathématiques «discrètes» (combinatoire, ...)

Applications : initialement militaires, puis approvisionnement, implantation, logistique, ...

### ❖ Une tentative prématurée : la traduction automatique

## La crise du logiciel et les débuts du génie logiciel

### ❖ Une prise de conscience (fin des années 60)

Les grands projets informatiques

- ne tiennent pas les délais
- dépassent leur budget
- répondent souvent mal aux attentes

Le constat obligé

- écrire des programmes corrects est *difficile*
- on en est encore à un stade artisanal

Les conclusions

- il faut développer des méthodes et des outils
- il faut mieux former les gens
- il faut passer à un stade industriel

### ❖ La naissance du génie logiciel

Les deux conférences *Software Engineering* (Garmisch, 1968 ; Rome, 1969)



## Le logiciel : comment ça se fabrique ?

### ❖ Les étapes de la création

Que veut-on faire ?

Cahier des charges  
Spécification

Modélisation

Représentation des objets du monde réel par des objets informatiques

Comment le faire ?

Le principe : l'algorithme  
une méthode correcte et efficace

La mise en œuvre : le programme  
une réalisation correcte et efficace de l'algorithme

La programmation n'est qu'une petite partie de la construction du logiciel

Des méthodes et outils de travail

Décomposition

Programmation  
et mise au point

Intégration de l'ensemble  
... et on recommence

A-t-on réussi ?

Test  
Vérification et validation  
Preuve

**Jamais du premier coup !**

## Les débuts du génie logiciel : mythes et déconvenues

### ❖ Le mythe de la solution «par les masses»

Si le projet est en retard, ajouter de la force de travail

**Faux !** cela ne fait qu'aggraver le problème

Frederick Brooks, *The Mythical Man-Month* (1975)

### ❖ Le mythe de la solution «par les outils»

On résoudra les problèmes en créant des outils plus raffinés

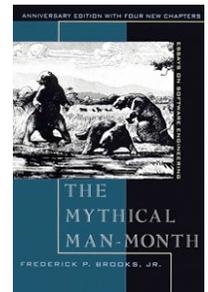
**Très discutable !** la qualité des équipes est le facteur dominant

### ❖ Le mythe de la «programmation automatique»

Les programmes du futur seront créés automatiquement

Le métier de programmeur va disparaître

**Faux !** Le métier va se transformer, mais dans le sens d'une plus grande qualification



## Éléments d'architecture du logiciel

### ❖ Conception descendante

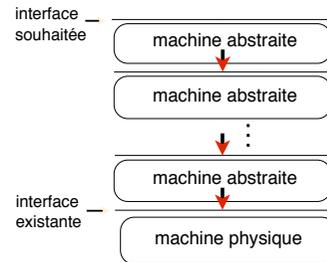
hiérarchie de «machines abstraites»

réalisation en logiciel

raffinement progressif



Edsger W. Dijkstra  
(1930 - 2002)  
crédit : U. of Texas at Austin



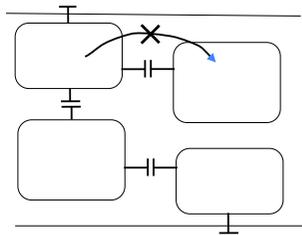
### ❖ Décomposition en modules

interface visible,  
réalisation cachée

tout doit passer  
par l'interface



David L. Parnas  
(1941 - )  
Wikimedia Commons, H. Baumeister



## Nouveaux langages des années 1970-80 (1)

### ❖ Dans la lignée d'Algol 60

Pascal (Niklaus Wirth, 1970), suivi de Modula-2 (1978)

Initialement destiné à l'enseignement ...

... mais aura une carrière plus large

Ada (Jean Ichbiah, 1979)

Une appel d'offres du ministère de la défense des USA

Une nouvelle tentative vers un langage universel

Utilisé dans les systèmes embarqués

Sûreté  
d'abord

### ❖ Retour vers la machine

BCPL (Martin Richards, 1967-69)

Premier langage à utiliser une machine virtuelle ...

... mais très proche de la machine réelle

C (Dennis Ritchie, 1971)

Issu de BCPL

Efficace, mais peu sûr : mieux vaut être expert

Efficacité  
d'abord

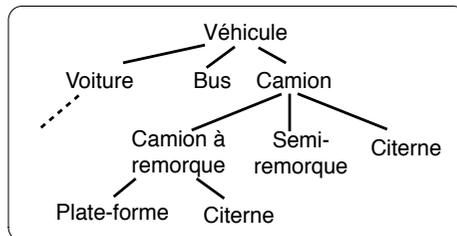
## Nouveaux langages des années 1970-80 (2)

### ❖ Langages à objets

*Modéliser le monde réel*

Simula (Dahl-Nygaard, 1967)

Smalltalk (Alan Kay et al.,  
Xerox PARC, 1978-80)



### ❖ Langages logiques

*Modéliser le raisonnement*

Prolog (Alain Colmerauer -  
Robert Kowalski, 1972)



Pierre est fils de Paul  
Julie est fille de Paul  
Marie est fille de Pierre  
Cécile est fille de Pierre  
Émile est fils de Julie  
A enfant de B = A fils de B **ou** A fille de B  
A cousin germain de B = A enfant de X  
**et** B enfant de Y **et** X enfant de Z  
**et** Y enfant de Z **et** X différent de Y  
Quels sont les cousins germains d'Émile ?  
--> Marie, Cécile

## Années 1970-80 : les nouvelles applications (1)

### ❖ Les systèmes d'information

Dans une entreprise ou une organisation : définition des différents processus (suite d'opérations réalisant une tâche) et des stocks et flux d'information associés

Couvre la gestion de la logistique, des stocks, du personnel, des relations clients, de la comptabilité, etc.

Outils nécessaires : modélisation, bases de données, traitement de la sécurité, applications spécifiques, ...

L'informatique n'est qu'un aspect des systèmes d'information

### ❖ La bureautique et le travail coopératif

Le rêve du «bureau sans papier»

Une transformation des conditions (et relations) de travail

Encore peu d'applications grand public

(mais Visicalc, premier tableur, en 1979 ; premier Word en 1983 ;  
outils Mac en 1984)

## Années 1970-80 : les nouvelles applications (2)

### ❖ Simulation numérique

Le gain de puissance des ordinateurs et les progrès en analyse numérique permettent des applications raffinées

Calcul de structures, mécanique des fluides, simulateurs de conduite, etc.

### ❖ Conception et fabrication assistées par ordinateur (CFAO)

CAO : l'informatique au bureau d'études

FAO : l'informatique à l'atelier et à l'usine

Conduite des procédés industriels : sûreté de fonctionnement et tolérances aux fautes, problèmes encore mal résolus

### ❖ Jeux sur ordinateur

Suivent les développements des outils graphiques

Premier jeu commercial en 1971



## Les bases de données : modèles

### ❖ Gérer et exploiter de grandes quantités de données

### ❖ Question 1 : assurer la conservation et l'accès rapide

Stockage et organisation des données sur les disques

### ❖ Question 2 : pouvoir répondre à des questions complexes

Modèle de données : entités et relations

Premières tentatives (années 1960) : hiérarchique, réseau

Une grande avancée : le modèle relationnel (Edgar Codd, 1970)

Nom	Prénom	Lieu	Service	...
Martin	Jeanne	5	Études	
Duval	Jacques	3	Après vente	
Bernard	Pierre	4	Commercial	
Lefèvre	Paule	1	Après vente	
...	...			

Lieu	Ville	Adresse	...
1	Bordeaux	...	
3	Paris	Centre	
4	Paris	Défense	
5	Grenoble	...	

## Les bases de données : transactions

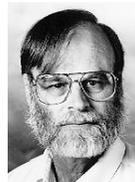
### ❖ Les aléas potentiels du traitement ...

Incohérence des données

Panne matérielle ou logicielle

Interférence entre traitements indépendants simultanés

Perte de résultats



### ❖ Une solution : les transactions (Jim Gray, 1981)

**Atomicité** : une transaction s'exécute complètement ou pas du tout

**Cohérence** : une transaction qui part d'un état cohérent aboutit à un état cohérent

**Isolation** : des transactions indépendantes n'interfèrent pas entre elles

**Durabilité** : une fois validés, les résultats seront préservés

## Les systèmes embarqués

### ❖ Quelques définitions

**Système embarqué** : système informatique associé à un objet physique

Interaction dans les deux sens : mesure (capteurs), commande (actionneurs)

**Système en temps réel** : système soumis à des contraintes de temps

**Système critique** : système dont la défaillance peut entraîner des conséquences catastrophiques (pertes de vie humaines, ...)

Beaucoup de systèmes embarqués sont aussi temps réel et critiques

### ❖ Brève histoire

Années 60 : systèmes de guidage de missiles (militaires)

Années 70 : routeurs, guidage de fusées (Apollo), fabrication automatisée, chimie

Années 80 et 90 : large extension (transports, énergie, télécommunications, satellites, carte à puce, terminaux de paiement, etc.)

Années 2000 : croissance explosive (téléphones, électro-ménager, puces radio (RFID), réseaux de capteurs, voitures, trains, avions, équipements médicaux, ...)

## Les systèmes embarqués

### ❖ Des problèmes ...

Les s.e. sont soumis à des contraintes : coût (d'où limites sur mémoire, processeurs), consommation d'énergie, environnement  
S'y ajoutent les aspects «temps réel» : respect des échéances  
Pour les systèmes critiques, la sûreté de fonctionnement est cruciale

### ❖ ... et quelques voies d'approche

Conception conjointe matériel-logiciel  
«Durcissement» du matériel (pour environnement hostile)  
Programmes optimisés (taille, élimination des fonctions inutiles)  
Modèles d'exécution spécifiques (langages synchrones)  
Vérification rigoureuse (*model checking*)  
Noyaux de systèmes spécialisés et certifiés  
Tolérance aux fautes

Confiance *justifiée* dans le bon fonctionnement

## Circuits intégrés et microprocesseurs

### ❖ Une révolution dans la fabrication des circuits

Des transistors (composants «discrets») ...  
... aux circuits intégrés (dans le silicium)

Intel 4004 (1971)  
Intel Museum

SSI (1958), MSI (1968), LSI (1973), VLSI (1980)

### ❖ Conception

Des outils puissants

### ❖ Fabrication

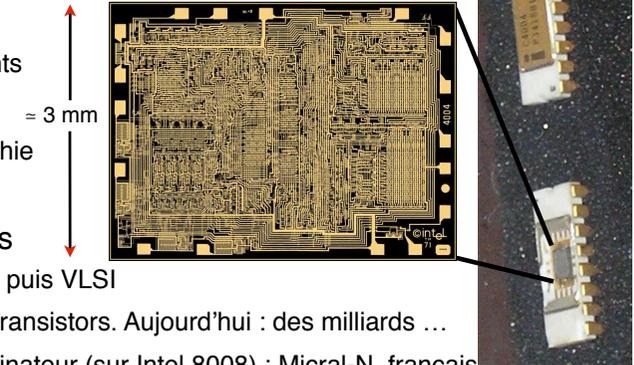
La micro-lithographie  
Le test (et le tri)

### ❖ Microprocesseurs

Utilisation des LSI puis VLSI

Intel 4004 : 2300 transistors. Aujourd'hui : des milliards ...

Premier micro-ordinateur (sur Intel 8008) : Micral-N, français



## Circuits intégrés et microprocesseurs

### ❖ La loi de Moore

une loi empirique

le nombre de transistors d'un circuit intégré double tous les 2 ans

aujourd'hui :

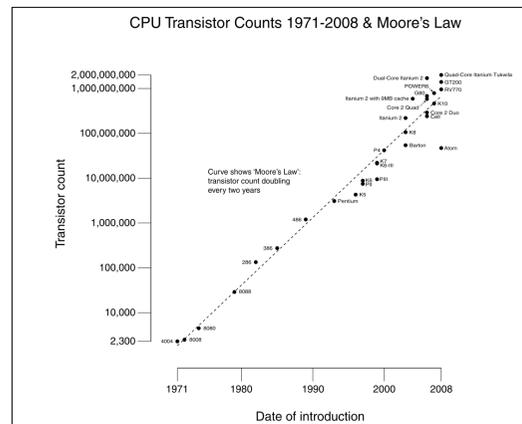
1 transistor  $\approx$  32 nm  
(1 cheveu  $\approx$  100 000 nm)

### ❖ Les limites

dimensions atomiques

1 atome de Si  $\approx$  0,11 nm  
2020 ???

évacuation de la chaleur multi-cœurs



Wgsimon, Wikimedia Commons

## Xerox PARC : un nouveau visage pour l'informatique

### ❖ Histoire

1970 : Création du Xerox PARC (*Palo Alto Research Center*)

1973-78 : révolution dans le paysage informatique

1981 : commercialisation manquée (Xerox Star)

2002 : PARC devient une société indépendante (filiale de Xerox)

### ❖ Principales contributions

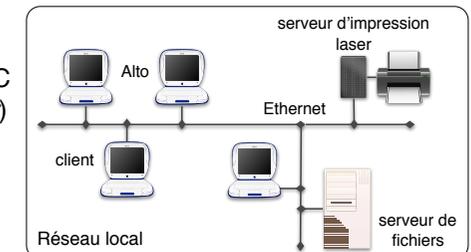
L'interface graphique *bitmap* et l'éditeur de texte *WYSIWYG*

Le réseau local Ethernet

L'imprimante à laser

Le langage Smalltalk

Le langage InterPress (précurseur de PostScript et de PDF)



## L'Ethernet

- ❖ Un réseau local (Robert Metcalfe, David Boggs, 1973)  
Communication rapide à faible distance (< 1 km)
- ❖ Un principe simple et efficace  
Si deux stations émettent en même temps (brouillage),  
chacune s'arrête et recommence un peu plus tard avec un  
délai aléatoire (et croissant)  
Inspiré d'un réseau radio (Aloha, 1970)  
Utilise initialement un câble coaxial

© 2010 DigiBarn Museum © 2007 David Monniaux



- ❖ Avantages  
Simplicité d'installation  
Capacité de croissance  
Performances : 3 Mbit/s en 1973, 10 à 100 Gbit/s aujourd'hui

## L'Alto : premier ordinateur personnel moderne

- ❖ Objectif : le «bureau électronique»  
Faciliter l'accès à l'information, la  
communication et le partage  
Fournir une interface simple et interactive
- ❖ Moyens

Serveurs spécialisés sur le réseau

Fichiers  
Serveurs d'impression  
Langages pour l'impression  
(retombée : Adobe)

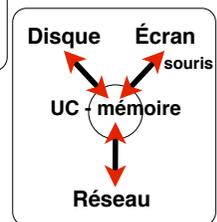
Écran à points, éditeurs WYSIWYG

«What You See is What You Get»

Principaux auteurs : Chuck Thacker, Butler Lampson,  
avec forte influence de Douglas Engelbart et Alan Kay

Caractéristiques

Mémoire : 125-512 Ko  
Disque : 2,5 Mo  
(amovible)  
Écran : 606x808  
(blanc et noir)  
Ethernet : 3Mbit/s



## Ordinateurs personnels

- ❖ Pendant ce temps, loin de Xerox PARC ...
- ❖ ... les premiers ordinateurs personnels «classiques»  
1973 : Micral N (français !)  
1974 : Altaïr  
1976 : Apple I  
1977 : Commodore PET, TRS 80  
1977 : Apple II (très populaire, produit jusqu'en 1993)  
graphique, son



Apple II © Rama

- ❖ Traits communs  
La plupart (avant Apple II) vendus en «kit»  
public plutôt «technique»  
Langage : Basic  
Logiciel : jeux, applications domestiques

## Ordinateurs personnels : l'IBM PC

- ❖ Naissance du PC (1981)  
Une stratégie nouvelle chez IBM  
utiliser des composants standard (non fabriqués chez IBM)  
rendre les spécifications publiques (d'où les «compatibles»)  
créer une division légère autonome pour la conception du PC  
Un succès commercial (mais surtout pour les clones ...)  
Les «compatibles»  
2 milliards de PC au total vendus jusqu'en 2008
- ❖ Les systèmes d'exploitation  
voir plus loin



IBM PC 5150  
«Tezza's collection»

## Ordinateurs personnels : les retombées de Xerox PARC

### ❖ Un transfert laborieux ...

En 1981 : Xerox Star  
trop lent, trop cher, mal présenté ...



En 1983 : Apple Lisa  
trop lent, trop cher, vite supplanté par le Mac ...

### ❖ La naissance du Macintosh (1984)

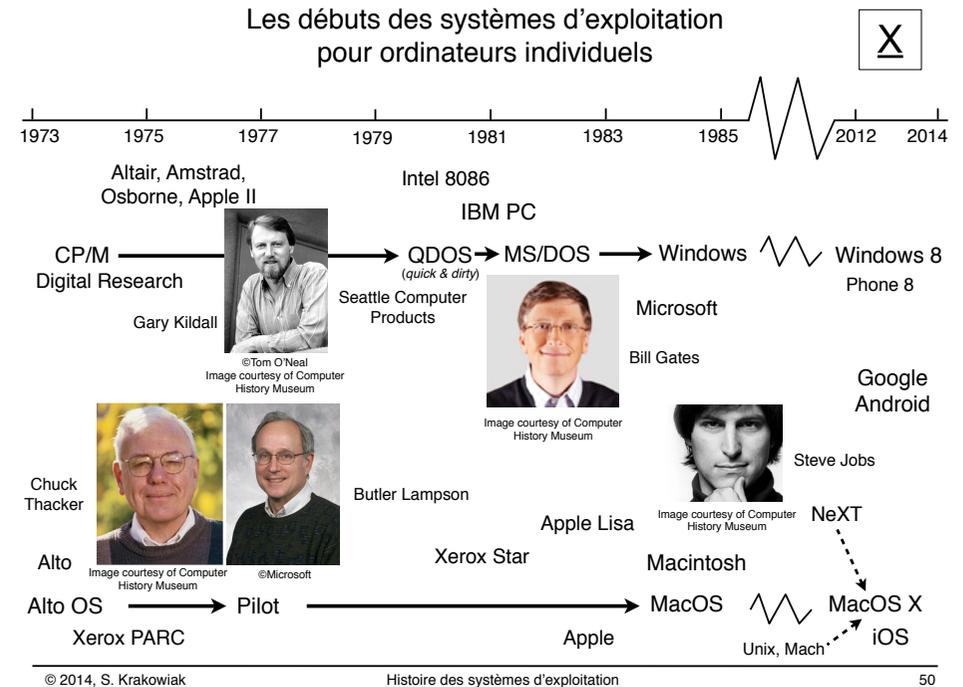
Un succès commercial ...  
... malgré les limitations initiales  
petite mémoire, pas de disque dur, peu de logiciel  
Publication assistée par ordinateur



### ❖ Le modèle «stations de travail - serveurs»

Le nouvel outil des professionnels (sous Unix)  
Apollo (1980), Silicon Graphics (1981), Sun (1982), ...  
Graphique, conception assistée, développement de logiciel, ...

## Les débuts des systèmes d'exploitation pour ordinateurs individuels



## Les réseaux : naissance de l'Arpanet

### ❖ Les origines

Dans les années 60, réflexions sur un réseau global d'ordinateurs  
Paul Baran, Robert Kahn, John Licklider, Lawrence Roberts  
Principes de la «commutation par paquets»  
Leonard Kleinrock

Lancement du projet par l'ARPA (agence militaire de financement de la recherche) en 1966-67

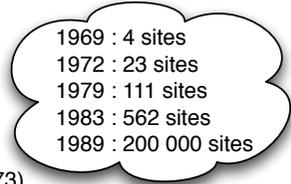
### ❖ Premiers échanges

Octobre 1969, 4 sites : UCLA, UCSB, SRI, Univ. Utah



### ❖ Développement

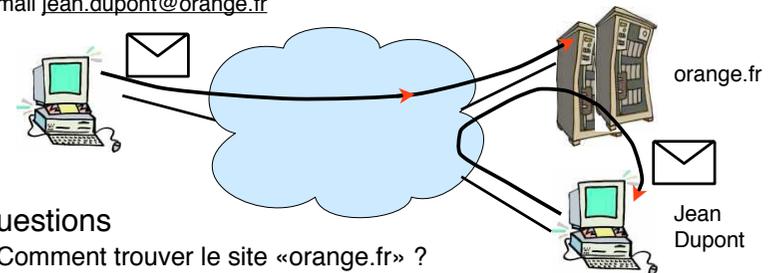
Transfert de fichiers (1971)  
Courrier électronique (Ray Tomlinson, 1972)  
Protocoles TCP et IP (Vinton Cerf, Robert Kahn, Jon Postel, 1973)



## Comment fonctionne l'Internet ? (1)

### ❖ Exemple : envoyer un courrier électronique

mail [jean.dupont@orange.fr](mailto:jean.dupont@orange.fr)



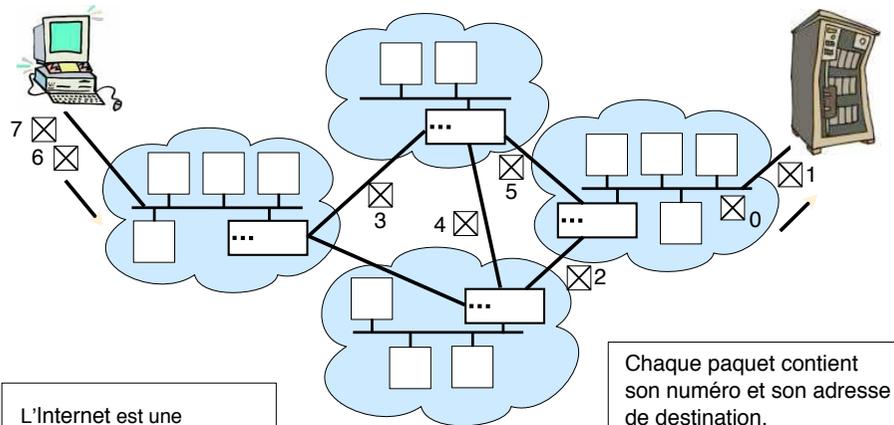
### ❖ Questions

Comment trouver le site «orange.fr» ?  
Comment acheminer le message vers ce site ?

### ❖ Trouver le site de destination

Chaque site connecté a une «adresse IP» qui indique où il se trouve  
Il y a un annuaire qui associe les noms («orange.fr») aux adresses (81.52.142.217)  
Comment trouver l'annuaire ? Réponse : on connaît son adresse IP !

## Comment fonctionne l'Internet (2) ?



L'Internet est une interconnexion de réseaux (*internetwork*), reliés entre eux par des routeurs

Chaque message est découpé en *paquets* de taille fixe

Chaque paquet contient son numéro et son adresse de destination.

Les routeurs contiennent les informations permettant d'acheminer le paquet vers son site destinataire.

## Comment fonctionne l'Internet ? (3)

### ❖ Les protocoles

Un protocole est un ensemble de règles pour organiser la communication sur un réseau

Il y a de nombreux protocoles qui s'appuient les uns sur les autres (pile)

### ❖ Le protocole IP (*Internet Protocol*) : le centre de tout

Assure l'acheminement d'un paquet entre un site et un autre

Pas de garanties (le paquet peut arriver abîmé, être retardé, ou se perdre, ...), ce qui permet de garder un protocole simple

Les garanties sont assurées par les protocoles qui utilisent IP

### ❖ Le protocole TCP : la base des applications

Assure le transport d'un flot d'information entre sites, avec garanties

Beaucoup d'applications (ex. : le Web) fondent leurs protocoles sur TCP

## De l'Arpanet à l'Internet

- ❖ Transition au début des années 1980 (environ 1 000 sites)  
Une pile de protocoles, TCP/IP ; un annuaire, le DNS ; une organisation
- ❖ TCP-IP (1973-1983), une avancée décisive (inspiration de Louis Pouzin)
- ❖ Le *Domain Name System*, l'annuaire de l'Internet (1983-84)  
Vers 1982, un seul annuaire centralisé, goulot d'étranglement  
Le DNS est un annuaire *réparti sur tout le réseau*  
Il a prouvé son efficacité (nombre de sites multiplié par 1 million !)
- ❖ La gouvernance de l'Internet
  - Internet Society* (ISOC) : orientations
  - Internet Architecture Board* (IAB) : management
  - Internet Engineering Task Force* (IETF) : choix techniques
  - Internet Corporation for Assigned Names and Numbers* (ICANN) : noms de domaines

Il n'y a pas de «roi de l'Internet» ...

## Applications de l'Internet

- ❖ Les premiers usages professionnels
  - rlogin* (1970) : accès à un ordinateur distant
  - ftp* (1971) : transfert de fichiers
  - mail* (1971) : courrier électronique
  - news* (1979) : groupes thématiques, participation ouvertepremières applications réparties (années 1980)
- ❖ La virtualisation à grande échelle  
*Cloud Computing*
- ❖ L'ouverture au grand public
  - Le Web (inventé en 1991, largement diffusé en 1995)  
ouvre l'ère des services
  - Les réseaux sociaux (existent depuis les *news*, mais explosion après 2005)

## Remarques finales (1)

### ❖ L'informatique parmi les sciences

L'informatique a acquis un statut de science ...

... mais ce n'est pas encore évident pour tout le monde  
et notamment en France

Les autres sciences intègrent la démarche informatique  
ce qui ouvre de nombreuses voies de recherche

### ❖ L'informatique dans la société

L'informatique est présente partout ...

... mais elle reste au fond mal connue et mal comprise  
notamment du monde politique

Il faut promouvoir une vraie culture informatique ...

pas seulement apprendre à utiliser un traitement de textes ou l'Internet

... donc un enseignement de l'informatique comme science  
dès le lycée

L'évolution du droit et des usages est un défi majeur

## Remarques finales (2)

### Deux actions en cours, parmi d'autres

#### ❖ Pour un enseignement de l'informatique à tous

Pourquoi ?

Comprendre et maîtriser le monde devenu numérique

Comprendre les grandes notions de la science informatique

Comment ?

Introduire l'enseignement de la programmation dès l'école primaire

il y a de bons outils (ludiques) pour cela (exemple : Scratch)

Poursuivre au collège, au lycée, et en formation continue

#### ❖ Pour un musée de l'informatique et de la société numérique

Raconter l'histoire de l'informatique, préserver le patrimoine

Aider à comprendre le présent, susciter des vocations

Un projet est en cours (réparti, partiellement virtuel)

<http://musee-informatique-numerique.fr/>

## Pour en savoir plus (1)

### ❖ Un site de diffusion de la culture informatique

Interstices (INRIA, CNRS, universités) : <http://interstices.info/>

### ❖ Un cours en ligne : Gérard Berry, Collège de France

[http://www.college-de-france.fr/default/EN/all/inn\\_tec2007/](http://www.college-de-france.fr/default/EN/all/inn_tec2007/)

### ❖ Une biographie

Andrew Hodges. *Alan Turing, ou l'énigme de l'intelligence*, Payot, 1988

### ❖ Un livre d'histoire

Pierre-Éric Mounier-Kuhn. *L'informatique en France de la seconde guerre mondiale au Plan Calcul*, Presses de l'université Paris-Sorbonne, 2010

### ❖ L'association Aconit (<http://www.aconit.org/>)

## Pour en savoir plus (2)

### ❖ Sur l'histoire des ordinateurs

Martin Campbell-Kelly, William Aspray, Nathan Emsberger, Jeffrey R. Yost

*Computer: A History of the Information Machine*, 3rd. ed., Westview Press

### ❖ Sur l'Internet

Christian Huitema, *Et Dieu créa l'Internet*, Eyrolles, 1995

*A Brief History of the Internet*

<http://www.isoc.org/internet/history/brief.shtml>

### ❖ Sur les ordinateurs personnels

Michel Volle, *Histoire du micro-ordinateur*

<http://www.volle.com/ulb/021122/textes/histoiremico.htm>

### ❖ Sur l'épistémologie de l'informatique

Gilles Dowek, *Les métamorphoses du calcul*, Éditions du Pommier, 2007

Page web personnelle : <http://proton.inrialpes.fr/~krakowia/fr-index.html>