

Histoire du Génie Logiciel

Voyage partiel et partial

Objectifs

Prendre conscience que le génie logiciel est une discipline jeune

Réaliser que les découvertes viennent de la recherche et de l'industrie

Comprendre les relations avec les domaines et les autres disciplines

Identifier les moments et les ressources clés

Travail en cours de consolidation, merci de repérer les erreurs / imprécisions

Plan

Naissance du génie logiciel – (12^{ème} – 1970)

Production et crise – (1970 – 1995)

A la vitesse d'Internet – (1995 – 2010)

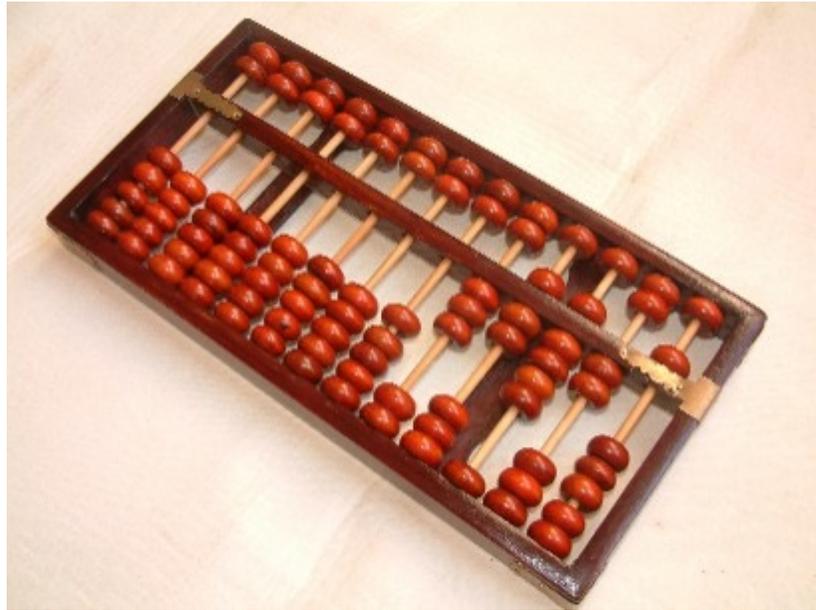
Le logiciel mange le monde – (2010 – Maintenant)

L'enseignement du GL en France

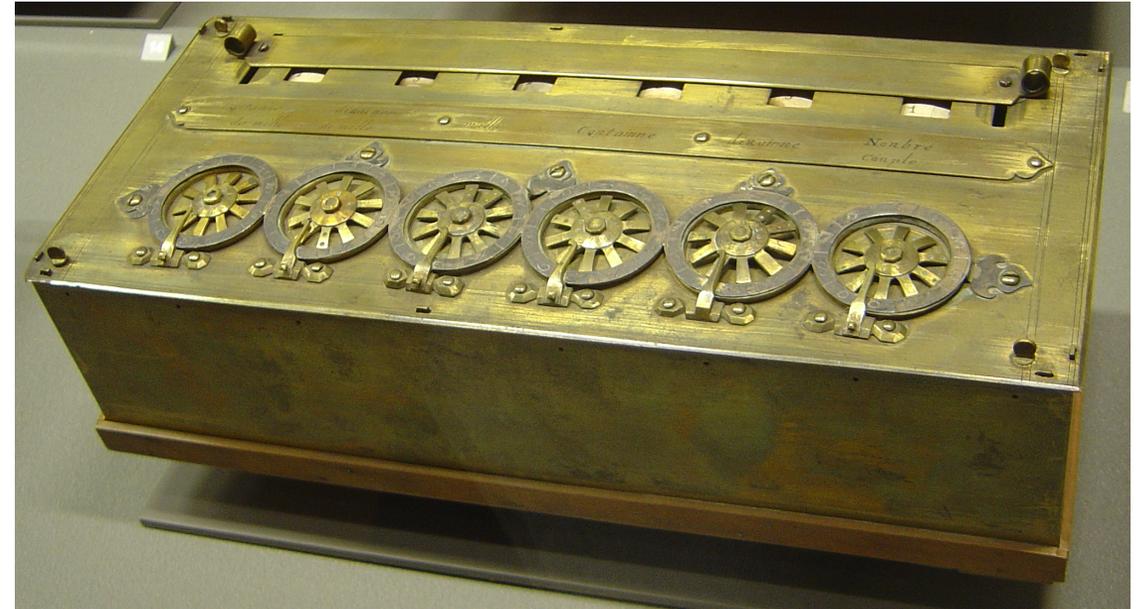
Naissance du GL

XII^e - 1970

L'abaque



Boulard XII^e / XV^e



Pascaline 1642

La machine à calculer



Machine à cylindre cannelé (concept inventé par Gottfried Leibniz 1685)



Arithmomètre (Thomas de Colmar en 1820)
sur la base des cylindres cannelés)

La machine analytique



Machine à calculer programmable – Charles Babbage 1834 (1906)



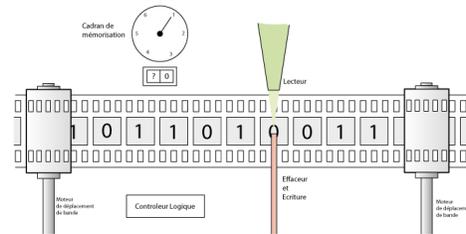
Cartes perforées (Instructions et données)

Premier programme 1843

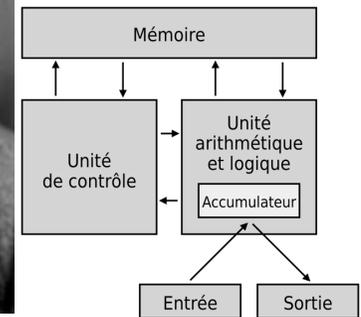


Ada Lovelace 1812 - 1852

Ordinateur moderne

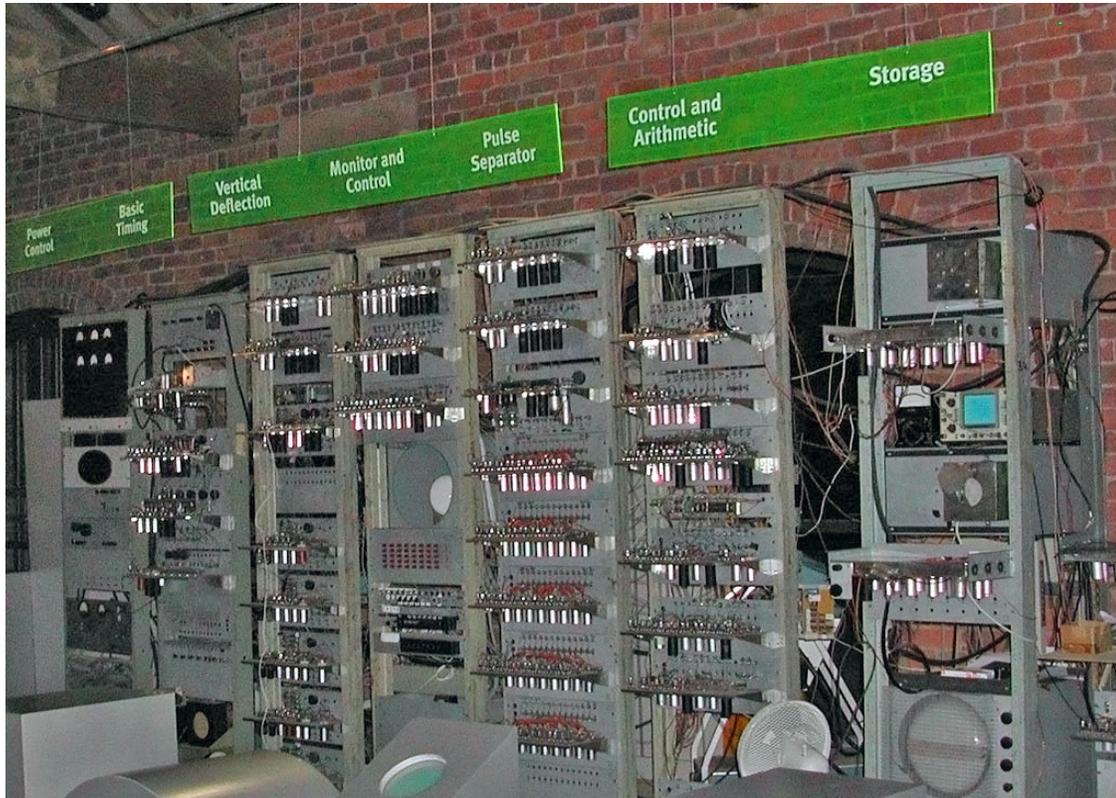


« *On Computable Numbers* »
Alan Turing, 1936

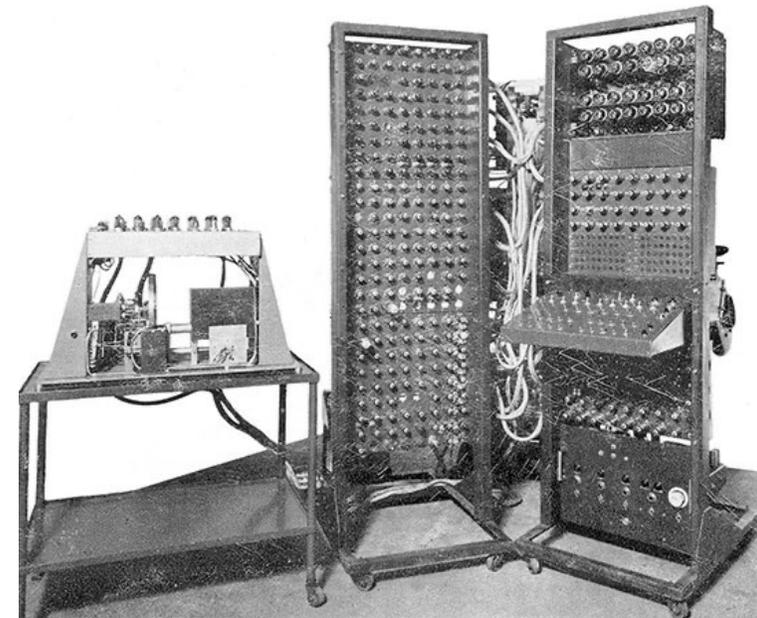


John von Neumann, 1945

Les premiers ordinateurs

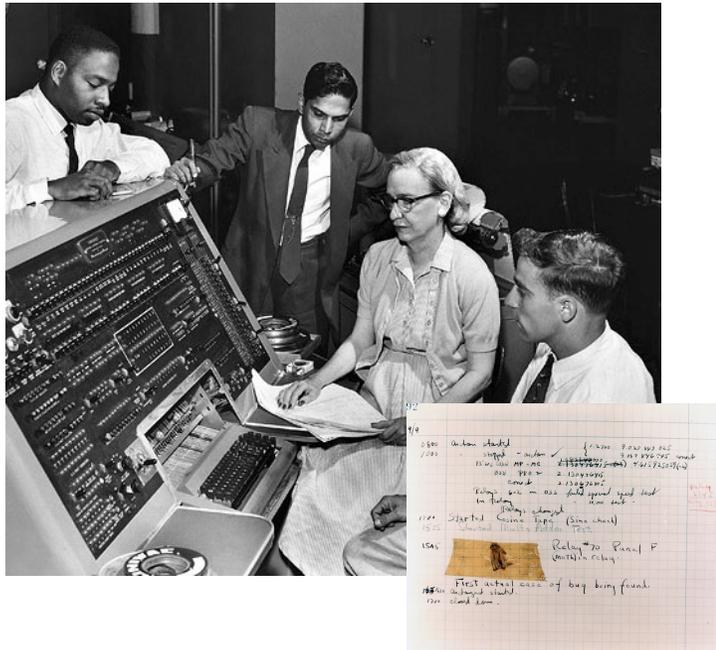


Manchester Baby
Small-Scale Experimental Machine (SSEM),
University of Manchester (1948)



La Machine de Couffignal 1952

Les premiers logiciels

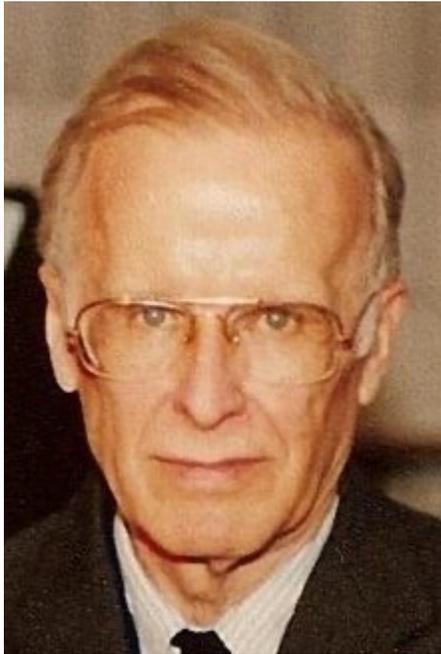


Grace Hopper
Premier compilateur 1950s
FLOW-MATIC



Tom Kilburn
Programme stocké
dans une mémoire électronique

Premiers langages de programmation



John Backus
FORTRAN, IBM 1950>1957

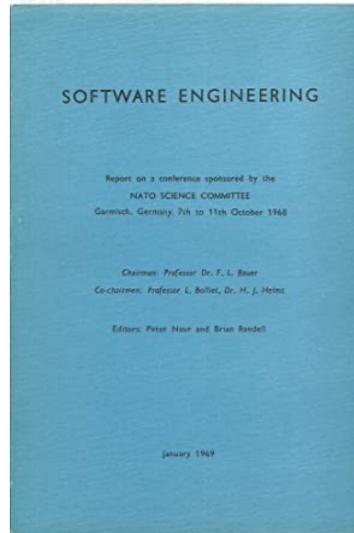


Mary K. Hawes
COBOL 1959

Software Engineering



Margaret Hamilton
Mission Apollo, 70s



Garmisch Report
1969



1969 Procès
IBM System/360

En résumé

Besoin de calcul

Précision, Sans erreur, Gestion de la Complexité

Automatisation

Abaque, machine à calculer, ordinateurs

Programme

Suite d'instructions, Stockage dans la machine, Numérisation

Programmation

Théorie, Méthode, Objets (langage, compilateur, tests, etc.)

Séparation Soft / Hard

=> Nouvelle discipline scientifique

Production et Crise

1970 - 1995

Définitions

Logiciel : programme et documentation associée.

Génie Logiciel : Discipline d'ingénierie qui concerne tous les aspects de la production de logiciel, de la conception initiale à son opérationnalisation et sa maintenance.

Software Engineering, Ian Sommerville

Les 3 contraintes de l'ingénierie

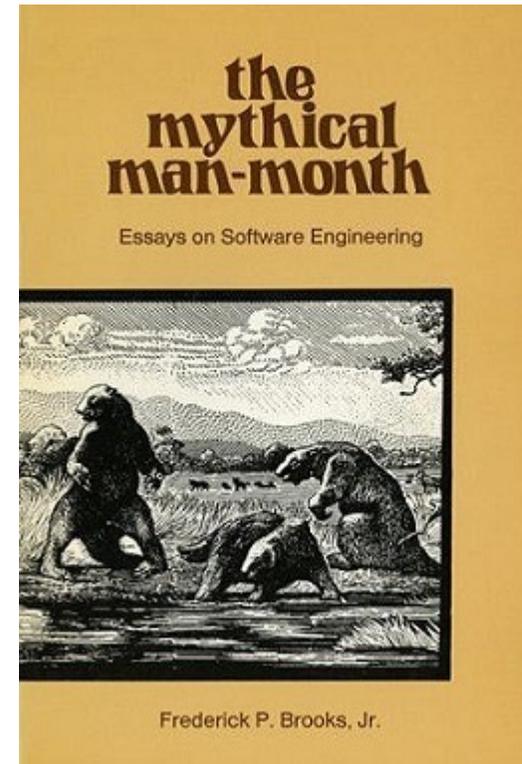


Constat d'échec

"No scene from prehistory is quite so vivid as that of the mortal struggles of great beasts in the tar pits... The fiercer the struggle, the more entangling the tar, and no beast is so strong or so skillful but that he ultimately sinks.

Large-system programming has over the past decade been such a tar pit... Most have emerged with running systems-few have met goals, schedules, and budgets."

Fred Brooks, 1975



Phases du génie logiciel

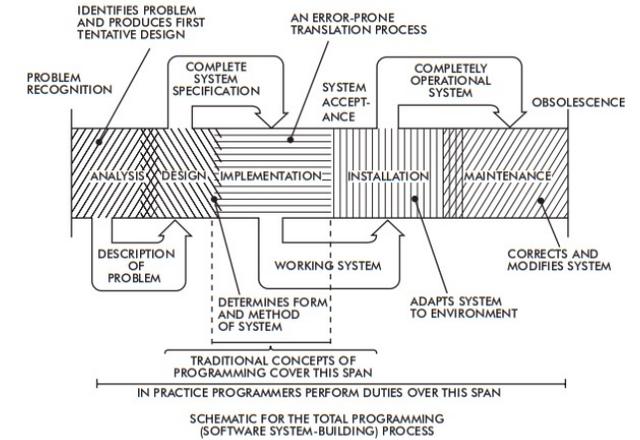
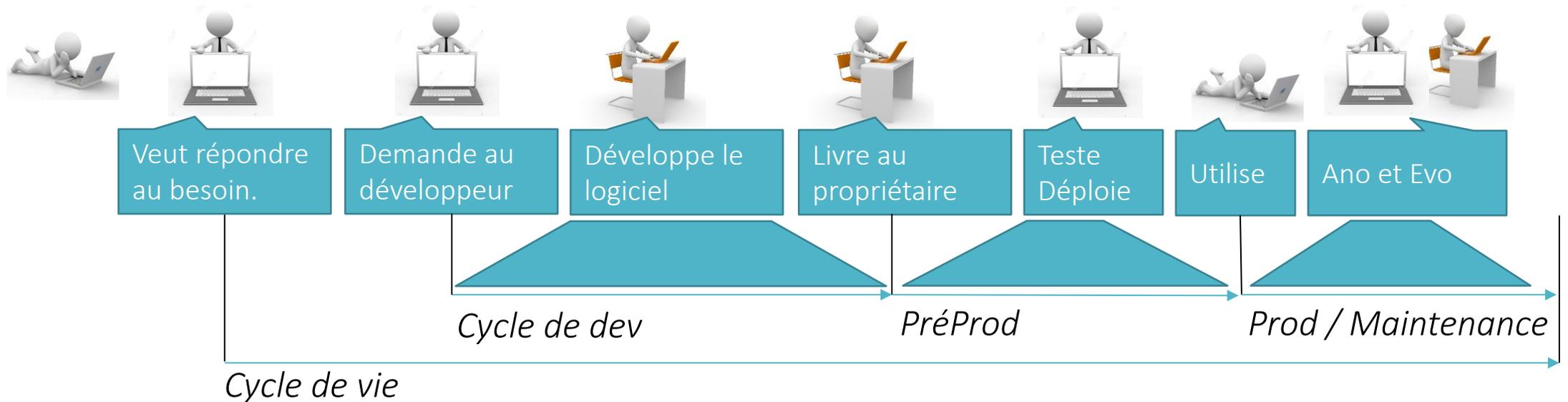


Figure 2. From Selig: Documentation for service and users. Originally due to Constantine.



Les méthodes

Waterfall – 1956

Phases réalisées en séquences

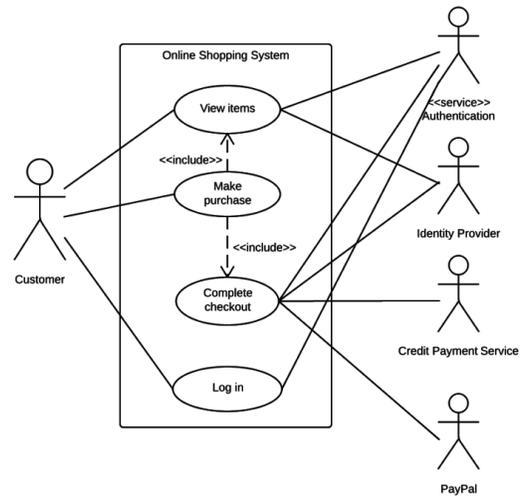
Cycle en V – 1980

Séparation des activités de vérification et validation

Cycle en spirale – 1988 (Barry W. Boehm)

Réalisation par itération

Analyse



Use Case
Yvar Jacobson 80s



METEOR 1998
Jean-Raymond Abrial

Conception

Essor des langage de programmation (<https://www.levenez.com/lang/lang.pdf>)

Basic 1964, Pascal 1970, SmallTalk 1971, Scheme 1975, C 1971, Eiffel 1986, Caml 1987, Haskell 1987, Python 1991, Java 1995

Bases de données

BDR (E.F. Codd 1970), ER (P. Chen 1976), SQL 1975 (1986),

Architecture

Merise 70s, Parnas 1972, OMT 1991, Design Patterns 1994, UML 1997

Production et maintenance

Infrastructures

IBM CICS 1966, Station Sun 80s, Windows 1985 (NT4 96), Linux 1991

Gestion de versions

SCCS 1972, RCS 1985, CVS 1990

Evolution

Lois de Lehman 1974, ITIL 80s,

En France

Les pionniers 50s

Grenoble (Jean Kuntzmann - Math),
Toulouse (Emile Durand – Physique),
Nancy (Jean Legras – Math)

2^{ème} vague 60s

Paris, Lille, Clermont-Ferrand,
Strasbourg, Nantes, Rennes, Besançon,
Bordeaux

*(Jean Hardouin-Duparc – Cristallographe,
Yves Haugazeau, J-M. Blondel, J-L. Joly, A.
Bouchet, P. Morel, Robert Cori, Bruno
Courcelle), LaBRI 1988.*

Plan Calcul 1966 - 1975

Charles de Gaulle (Michel Debré) :
autonomie en informatique. Création
IRIA puis Inria (1967). Fin au projet par
Valéry Giscard d'Estaing.

Création des IUT 1966

Christian Fouchet (Exp. 1965),
Informatique, Génie électrique et
Informatique Industrielle

Plan Informatique pour tous 1981

Elémentaire, Collège, Lycée

En résumé

Production en crise

Propositions de multiples solutions

Méthodes, languages, etc.

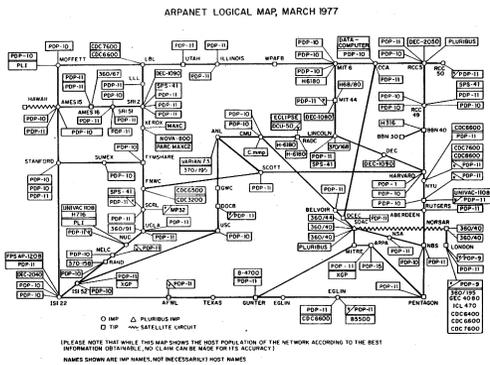
Le bug reste omniprésent



Logiciel et Internet

1995 - 2010

Internet



ARPANET
1967



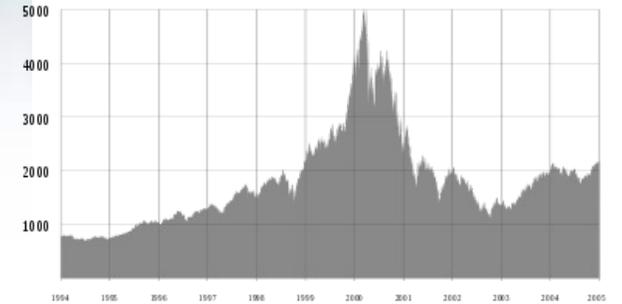
Mail
1971



TCP/IP Internet
1983



WWW
Tim Berners-Lee
1991



La Bulle
2000

“Information Systems Development @ Internet Speed: A New Paradigm in the Making!” R. Baskerville (2002)

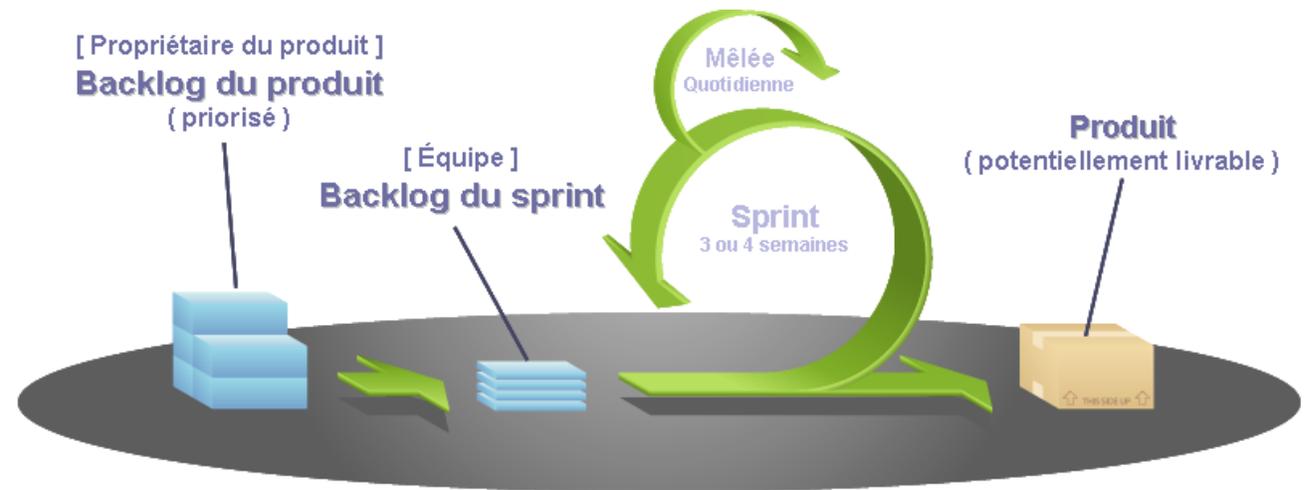
Les méthodes

Scrum – 1995 (Ken Schwaber)

Spirale, Agile, Time Box

Agile – 1999 (Kent Beck)

Programmation extrême



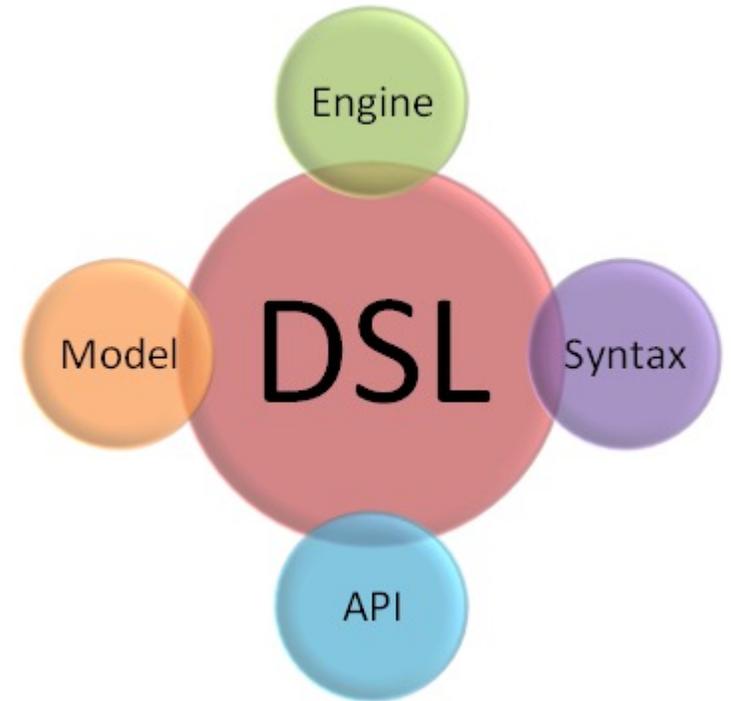
Analyse

UML -> MDA -> DSL

Modélisation du problème, Génération du code

Behavioral Driven Development, Dan North, 2006

Expression du problème à l'aide de scénario



Conception

Langage de programmation web (<https://www.levenez.com/lang/lang.pdf>)

JavaScript 1995, PHP 1995,

Bases de données

NoSQL 1998, MongoDB 2009

Architecture

SOAP 2003, REST 2000, Hadoop 2006

Production et maintenance

Infrastructures

LAMP 1998, AWS 2006, Iphone 2007, Android 2007

Gestion de version

SVN 2000, Git 2005

Evolution

Fork, PullRequest

En France

Enseignement post-bac:

Ecoles, IUT, Licence, DEA, DESS

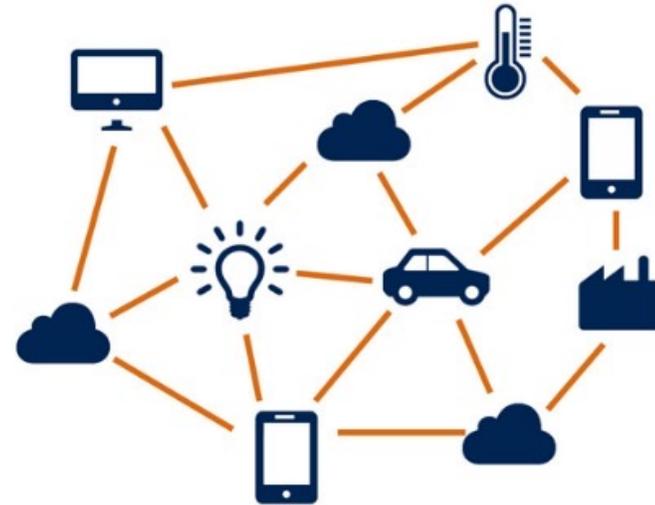


En résumé

Explosion des usages Internet

Répartition des systèmes et des équipes

Accroissement de la complexité



Le logiciel mange le monde

2010 - maintenant



Mark Andreessen
founder of Netscape,
renowned Venture Capitalist
Andreessen-Horowitz

Software is eating the
world, in all sectors

In the future every
company will become a
software company

Depuis environ 10 ans

Méthode

Passage à l'échelle, Concept de Produit, Accelerate

Analyse

Scenario et/ou propriétés

Conception

Convergence des langages, MicroService,

Production et maintenance

Virtualisation, Chaine d'intégration

En France

Création de l'INS2I 2010

Lycée (NSI), Collège (Scratch)

