

# Outils collaboratifs

Responsable : E. Poisson Caillault

ULCO  
Université du Littoral Côte d'Opale  
Bureau : Calais, A214.

intervenants : É. Poisson (A214)  
mails : nom@univ-littoral.fr

2025

## 1 Historique

- Monde de l'Entreprise
- ... Pour le développement

## 2 Git

## 3 GitHub

# Dates charnières

Étapes clés de l'évolution des outils collaboratifs :

- ...- 1970 Avant le numérique
- 1980-1990 Émergence des ordinateurs
- 1990 - Essor d'Internet (années 1990-2000)
- 2000 - Passage au cloud (années 2000-2010)
- 2010 - Collaboration en temps réel
- 2020 - Ère post-pandémie / télétravail
- 2025 - Génératifs, immersifs

# Prémices de la collaboration avant le numérique

La collaboration à cette époque reposait sur des outils traditionnels comme :

- Le courrier postal
- Le téléphone
- Les réunions en présentiel
- Les documents étaient rédigés à la main ou à la machine à écrire, puis partagés physiquement.

Exemple d'outil pionnier :  
La télécopie via le fax-similé.

# Les premiers outils numériques (années 1980-1990)

1980-1990 : la période dorée du fax

L'émergence des ordinateurs personnels a marqué une révolution dans les outils collaboratifs.

## Avancées-clés :

- E-mail (années 1970-80) : L'une des premières innovations majeures, permettant de communiquer à distance en temps quasi réel.
- Groupware/Collecticiel (fin des années 1980) : Des logiciels permettant aux équipes de partager des documents et des calendriers.

### Exemple d'outil pionnier :

Lotus Notes (1989) combine messagerie électronique, bases de données partagées et calendriers collaboratifs.

# L'essor d'Internet (années 1990-2000)

Avec l'arrivée d'Internet, de nouveaux outils collaboratifs en ligne ont vu le jour, permettant de travailler à distance sans limite géographique.

## Avancées-clés :

- Messagerie instantanée (ICQ, MSN Messenger)
- Forums et wikis : Création de plateformes permettant de partager des connaissances (comme Wikipedia en 2001).
- Outils de gestion de projets : ex. Microsoft Project

1991 à 2004, le nombre d'ordinateurs personnels dans le monde est passé de 130 millions à 775 millions.

# Le passage au cloud (années 2000-2010)

L'essor du cloud computing a permis aux outils collaboratifs de devenir accessibles depuis n'importe quel appareil connecté à Internet.

## Avancées-clés :

- Google Docs (2006) : Révolutionnant la collaboration en ligne en permettant de coéditer des documents en temps réel.
- Outils de stockage cloud : Lancement de services comme Dropbox (2007) et Google Drive (2012).
- Réseaux sociaux d'entreprise : Des plateformes comme Yammer (2008) ont permis une communication interne fluide au sein des organisations.

# La collaboration en temps réel (années 2010-2020)

La tendance est passée des outils de stockage vers des plateformes de collaboration en temps réel, plus intégrées et interactives.

## Avancées-clés :

- Slack (2013) : Une plateforme de messagerie instantanée conçue pour les équipes professionnelles.
- Microsoft Teams (2017) : Une plateforme unifiant la messagerie, les appels vidéo, le partage de fichiers et la gestion de projets.
- Trello (2011) et Asana (2012) : Des outils de gestion de tâches et de projets collaboratifs.

# L'ère post-pandémie et le télétravail (2020 et au-delà)

La pandémie de COVID-19 a accéléré l'adoption massive des outils collaboratifs. Le travail à distance est devenu la norme pour de nombreuses entreprises.

## Avancées-clés :

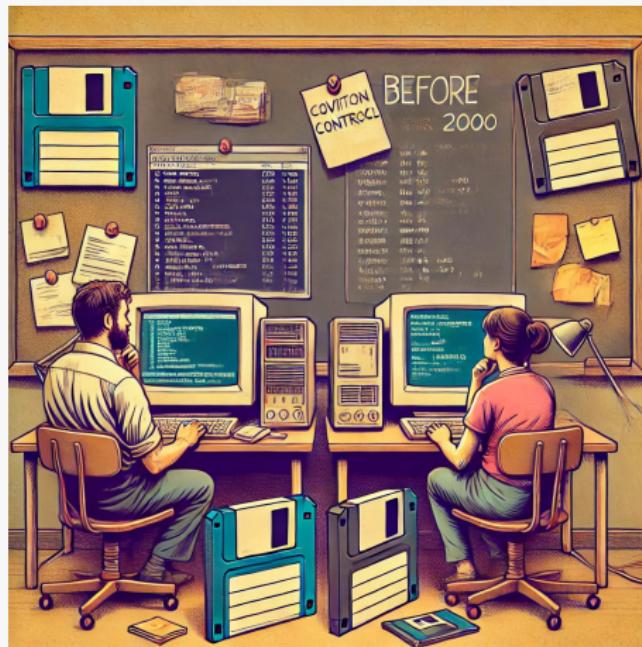
- Zoom : Explosion de la visioconférence pour maintenir la communication d'équipe.
- Miro : Outil de tableau blanc numérique interactif.
- Notion : Plateforme tout-en-un combinant la prise de notes, la gestion de projets et la collaboration en temps réel.
- IA intégrée : début sommaire des fonctionnalités d'intelligence artificielle pour automatiser certaines tâches (comme les résumés automatiques ou la prise de notes, traduction ou correction de code).

# Tendances futures (2025 et au-delà)

Les outils collaboratifs continueront d'évoluer avec :

- L'intégration de l'intelligence artificielle
- La réalité augmentée et virtuelle (VR/AR) pour des réunions immersives.
- Des plateformes unifiées combinant tous les aspects de la collaboration (messagerie, gestion de projet, visioconférence).

# Pour le développement



Outils collaboratifs avant 2000

# Premiers langages...1940-1980

## Les débuts de la programmation (1940-1960)

- **1949 : Assembleur** — Premier langage de bas niveau.
- **1957 : FORTRAN** — Premier langage de haut niveau.
- **1958 : LISP** — Utilisé en intelligence artificielle.
- **1959 : COBOL** — Conçu pour les applications métiers.

## Langages (1960-1980)

- **Pascal (1970)** — Langage pour l'enseignement de la programmation.
- **C (1972)** — Langage généraliste, utilisé pour UNIX.
- **Smalltalk (1980)** — Pionnier de la programmation orientée objet.

# Premiers outils de développement

Apparition des éditeurs de texte et IDE (1960-1980)

- **TECO (1962)** — Premier éditeur de texte pour la programmation.
- **Emacs (1976)** — Éditeur extensible.
- **Vi (1976)** — Éditeur populaire sous UNIX.

# Montée en puissance des IDE et gestion de versions (1980-2000)

## Langages

- **C++ (1983)** — Extension du langage C.
- **Python (1991)** — Langage interprété et lisible.
- **Java (1995)** — Langage portable pour les applications web.
- **PHP (1995)** — Pour le développement web dynamique.
- **JavaScript (1995)** — Langage pour le web.

## Outils de développement

- **Turbo Pascal (1983)** — Un des premiers IDE.
- **Visual Studio (1997)** — IDE complet de Microsoft.
- **Eclipse (2001)** — IDE open-source populaire.

# Montée en puissance des IDE et gestion de versions (1980-2000)

- **CVS (1990) — Concurrent Versions System**
- **Subversion (SVN) (2000) — Remplaçant de CVS.**

CVS, ou Système de Contrôle de Version, est un outil utilisé pour suivre les modifications apportées à des fichiers, généralement dans le cadre de projets de développement logiciel.

Il permet à plusieurs personnes de travailler sur les mêmes fichiers en même temps, en enregistrant chaque modification et en permettant de revenir à une version précédente si nécessaire. Cela aide à éviter les conflits, à suivre l'historique des changements et à gérer les versions du projet de manière efficace.

# L'ère des frameworks et outils collaboratifs (2000-2010)

## Langages

- **C# (2000)** — Langage de Microsoft.
- **Ruby on Rails (2005)** — Framework web.
- **Django (2005)** — Framework pour Python.
- **AngularJS (2010)** — Framework JavaScript.

## Outils de développement

- **Git (2005)** — Système de gestion de versions décentralisé.
- **GitHub (2008)** — Plateforme de collaboration autour de Git.
- **PyCharm (2010)** — IDE pour Python.

# L'essor des outils modernes et du cloud (2010-2025)

- **React (2013)** — Bibliothèque JavaScript.
- **Vue.js (2014)** — Framework léger et flexible.
- **Kubernetes (2014)** — Outil d'orchestration de conteneurs.
- **Docker (2013)** — Plateforme de conteneurisation.

# Qu'est-ce que Docker ?

Docker est une plateforme permettant de créer, déployer et exécuter des applications dans des conteneurs.

- Un **conteneur** est une unité légère, portable et indépendante qui contient tout le nécessaire pour exécuter une application : le code, les bibliothèques, les outils système, etc.
- **Différence avec une machine virtuelle** : Contrairement aux machines virtuelles, les conteneurs partagent le noyau du système d'exploitation, ce qui les rend plus légers et plus rapides.
- **Avantages :**
  - Facilité de déploiement.
  - Portabilité entre différents environnements.
  - Réduction des problèmes liés aux différences de configuration.
- Utilisé dans le développement moderne pour déployer des applications de manière cohérente dans des environnements de cloud, de test ou de production.

# Outils de développement modernes

- **Visual Studio Code (2015)** — Éditeur de texte populaire.
- **IntelliJ IDEA** — IDE puissant pour Java.
- **GitLab** — Alternative à GitHub.
- **Jenkins** — Serveur d'intégration continue.

# L'intégration de l'IA (2020 et au-delà)

## Génération de code

- **GitHub Copilot (2021)** — Assistant de codage basé sur l'IA développé par GitHub et OpenAI. Il propose des suggestions de code en temps réel et peut aider à écrire des fonctions entières, générer du code à partir de commentaires, ou suggérer des corrections. basé sur **OpenAI Codex** capable de générer du code à partir de simples instructions en langage naturel.  
Langages supportés : nombreux langages comme Python, JavaScript, Ruby, Java, etc.
- **Replit Ghostwriter** — Replit est une plateforme en ligne qui permet de créer et exécuter des projets de programmation en utilisant une interface web. Elle propose également une fonctionnalité d'IA qui aide les utilisateurs à écrire et corriger du code. Assistant IA pour la programmation.

# L'intégration de l'IA (2020 et au-delà)

## Autocomplétion

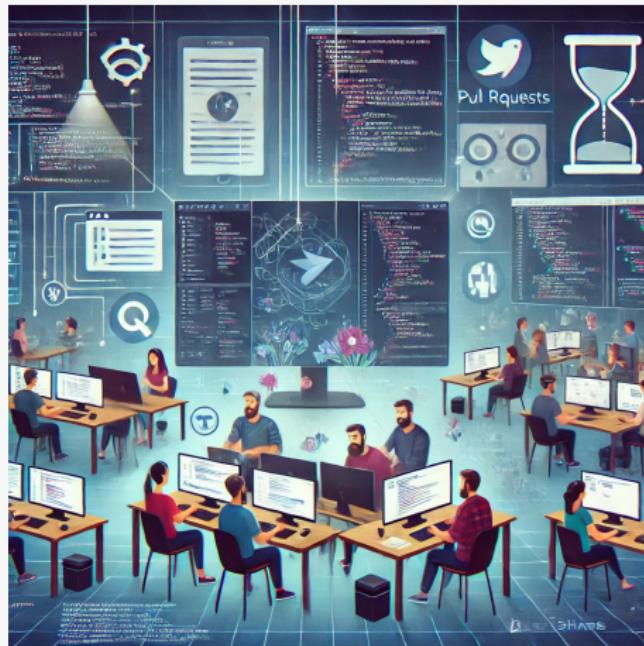
- **Tabnine** — Autocomplétion intelligente, aide à automatiser l'écriture de code et accélère le processus de développement. Il peut être intégré à des IDE populaires comme VS Code, IntelliJ, et autres.  
Langages supportés : Java, Python, JavaScript, TypeScript, C++, et d'autres.
- **Codota**, plateforme d'IA pour l'autocomplétion du code qui aide les développeurs à écrire des lignes de code plus rapidement et avec moins d'erreurs. Elle est principalement utilisée pour Java, Kotlin, et d'autres langages populaires.
- **Kite**, utilise l'IA pour la complétion automatique de code, avec des suggestions intelligentes pendant que vous tapez. Il est utilisé pour améliorer la productivité des développeurs, surtout dans le cadre du développement Python, Go.

# Traduction de langages

Outils de traduction de code (transpileurs - compilateur source à source) :

- **Source Level Translator (SLT)**, conversion de code écrit dans un langage de programmation vers un autre. Pas directement un produit basé sur l'IA, il utilise des algorithmes pour effectuer cette tâche.  
Langages supportés : Convertit généralement entre des langages comme C, C++, Java, Python, etc.
- **Transcrypt (2018)**, compilateur Python qui permet de traduire du code Python en JavaScript. Ce type de traduction est utile pour les projets qui nécessitent une intégration entre les deux langages.
- **Babel (2015)**, un transpileur JavaScript populaire, utilisé pour convertir du code moderne JavaScript (ES6+) en une version compatible avec des navigateurs plus anciens. Bien qu'il ne soit pas basé sur l'IA, il fait partie des outils utilisés pour la conversion du code.  
Langages supportés : JavaScript, JSX (React).
- **P2J (2008)** transpilation Python en JavaScript.

# Systèmes de gestion Git



je Git, tu GitHub, iel GitLab...

