XWNATIS

TypeScript

Michael

XWNATIS

XDNATIS

Compétence visée :

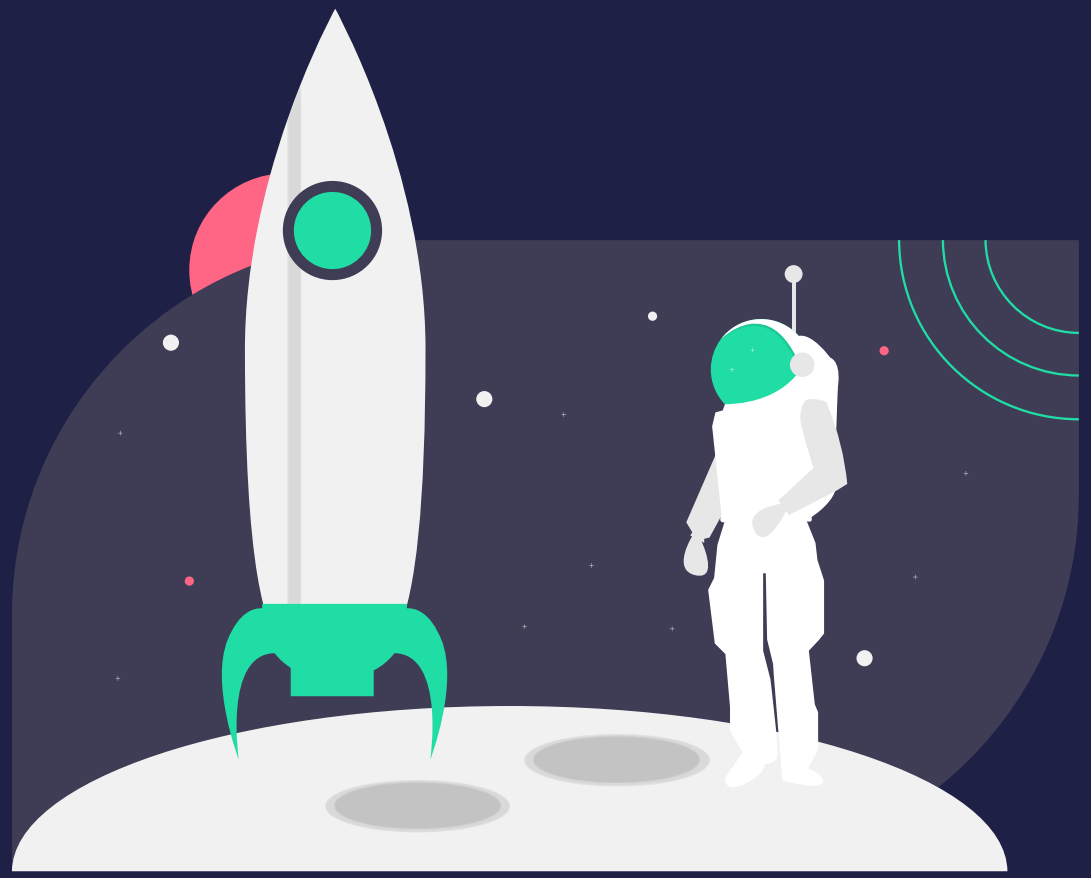
Comprendre l'articulation de TypeScript autour du JavaScript

XWIKI NATIS

1. Le typage statique
2. La transpilation
3. Inconvénients
4. Débugging et source maps

XWIKI SAS

XWIKI NATIS



XWIKI SAS

XEBIA NATIS

LE TYPAGE STATIQUE

XNATIS

En termes simples, TypeScript est un langage de programmation open source développé par Microsoft qui compile en JavaScript.

Depuis sa sortie en 2012, le langage est resté en développement actif et continue de gagner en popularité chaque année.

XWIKI NATIS

La prise en charge du typage fort ne fait pas partie de la norme ECMAScript et ne sera probablement jamais due à la nature interprétée au lieu de la nature compilée de JavaScript.

XDNATIS

L'augmentation clé que TypeScript apporte à JavaScript est un système de types (d'où le nom "TypeScript").

XEBIA NATIS

Le système de types de TypeScript est incroyablement riche et comprend : des interfaces, des énumérations, des types hybrides, des génériques, des types d'union/intersection, des modificateurs d'accès et bien plus encore.

XEBIA NATIS

Le site officiel de TypeScript donne un aperçu de ces fonctionnalités. Le système de typage de Typescript est comparable à la plupart des autres langages typés comme le C# par exemple.

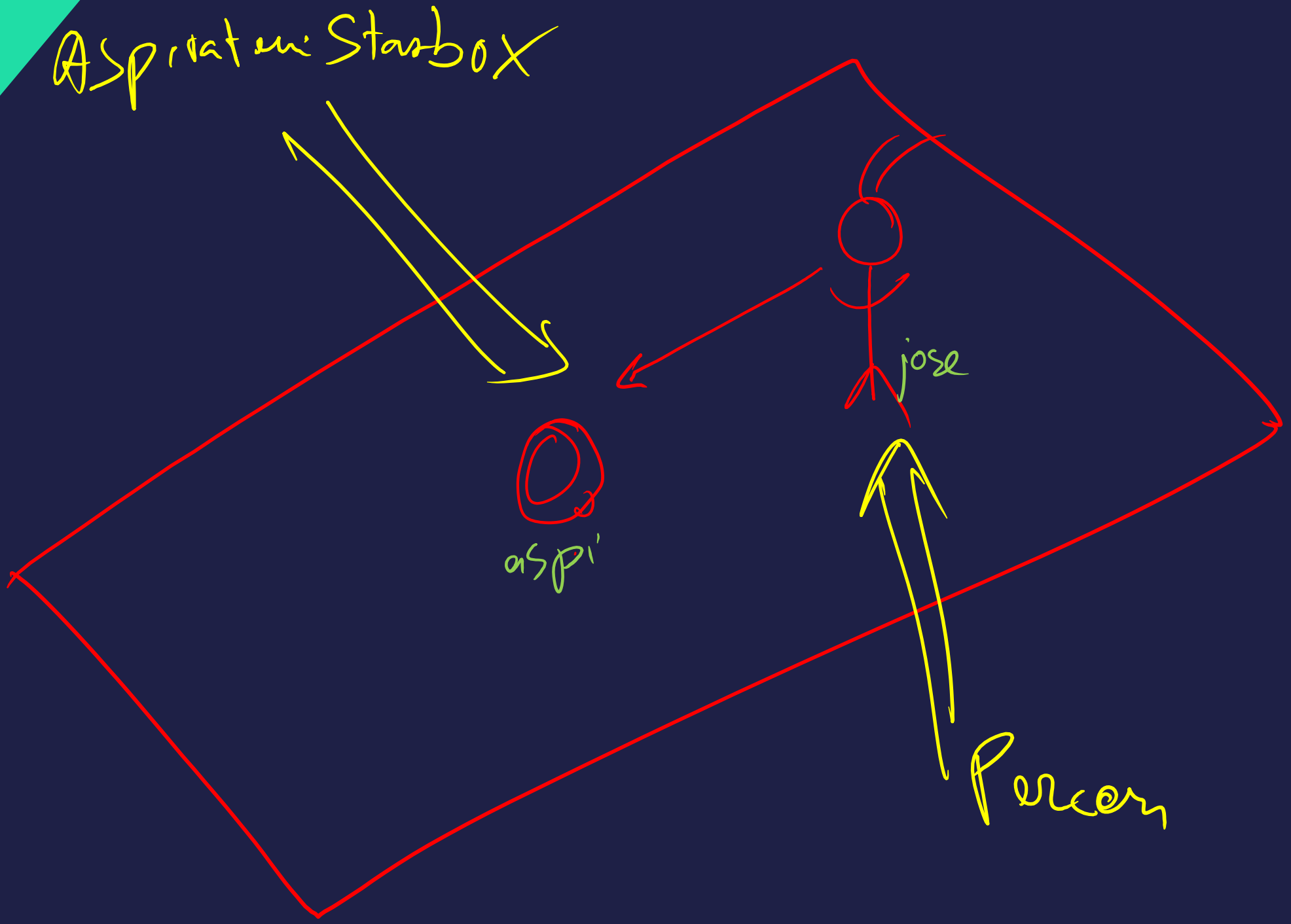
XEBIANATIS

1. Ajouter une sémantique d'architecture au niveau du code
2. Permettre le type checking pour vérifier la cohérence et par conséquent, anticiper les crashs en production
3. Faciliter la compréhension du code

XEBNATIS

```
interface Electromenager {
    nettoyer(): void;
}

abstract class Aspirateur implements Electromenager {
    protected abstract aspirer(): void;

    public nettoyer(): void {
        this.aspirer();
    }
}

class ApirateurStarbox extends Aspirateur {
    protected aspirer(): void {
        console.log('Aspire')
    }
}

const aspi = new ApirateurStarbox();
aspi.nettoyer();
```

XWIKI NATIS



XEBIA NATIS

RUNTIME : interaction entre objets

Public	Privé
public	private protected

STATIC : organisation des classes (héritage)

XWIKI NATIS

TRANSPILATION

XEBIANATIS

Les transpilers sont également connus sous le nom de compilateurs source-à-source. Donc, essentiellement, ils sont un sous-ensemble de compilateurs qui prennent un fichier de code source et le convertissent en un autre fichier de code source dans une autre langue ou une version différente de la même langue.

XWIKI NATIS

La sortie est généralement compréhensible par un humain. Cette sortie doit encore passer par un compilateur ou un interpréteur pour pouvoir s'exécuter sur la machine.

XKNATIS

Le compilateur TypeScript prend en charge la compilation incrémentielle (option --watch du compilateur) afin que toutes les modifications ultérieures puissent être compilées plus rapidement.

XEBIA NATIS

La configuration du transpiler est consignée dans un fichier tsconfig.json

projet > typescript > tsconfig.json > {} compilerOptions

```jsonc
{
  "compilerOptions": {
    /* Basic Options */
    // "incremental": true,
    "target": "es6",
    "module": "es2015",
    "lib": ["ES2015", "DOM"],
    // "allowJs": true,
    // "checkJs": true,
    // "jsx": "preserve",
    // "declaration": true,
    // "declarationMap": true,
    // "sourceMap": true,
    // "outFile": "./",
    "outDir": "./dist",
    // "rootDir": "./",
    // "composite": true,
    // "tsBuildInfoFile": "./",
    // "removeComments": true,
    // "noEmit": true,
    // "importHelpers": true,
    // "downlevelIteration": true,
    // "isolatedModules": true,
```

XWIKI NATIS

Chaque ligne de la configuration permet d’activer ou de désactiver un comportement du transpiler.

XWIKI SAS

XWIKI NATIS

INCONVENIENTS

XWIKI NATIS

La configuration nécessaire peut être assez compliquée à comprendre ou a démystifier ! ☹

XEBIA NATIS

Cependant, ce n'est pas strictement un inconvénient de TypeScript, mais plutôt un inconvénient qui découle de son interopérabilité totale avec JavaScript, ce qui lui-même laisse encore plus de place aux complications.

XEBIA NATIS

Un autre argument contre TypeScript est qu'il nécessite une compilation, contrairement à JavaScript.

XEBIA NATIS

Pour utiliser TypeScript, vous avez besoin d'un processus de construction pour compiler en code JavaScript.

Le processus de construction dépend de la taille de votre projet.

XONATIS

XWIKI NATIS

SOURCE MAP

XWIKI NATIS

Un débugueur (de l'anglais debugger) ou débogueur (de la francisation bogue) est un logiciel qui aide un développeur à analyser les bugs d'un programme.

XWIKI NATIS

Pour cela, il permet d'exécuter le programme pas-à-pas — c'est-à-dire le plus souvent ligne par ligne —, d'afficher la valeur des variables à tout moment et de mettre en place des points d'arrêt sur des conditions ou sur des lignes du programme.

XEBIA NATIS

Le compilateur TypeScript peut intégrer les informations de carte source dans les fichiers .js générés ou créer des fichiers .map distincts.

XEBIA NATIS

Les informations de carte source peuvent être utilisées par des utilitaires de débogage tels que Chrome DevTools et d'autres IDE pour relier les lignes du JavaScript à celles qui les ont générées dans le TypeScript.

XEBIA NATIS

Pour aller plus loin

https://www.typescriptlang.org/