

PSB 2022

Cours Javascript : cours : <https://github.com/ystroppa/PSB2020>

Sujet à valider ensemble

Yvan Stroppa

IR CNRS Linguistique

Yvan.stroppa1@univ-orleans.fr

Projet : 2 à 3 étudiants

Rapport

Cahier de charges : expression de besoins

Analyse et conception

Explications et illustrations

Phase des Tests (vérif sur tous les navigateurs)

Conclusion

Programme à livrer

Dépôt de doc....

<https://github.com/ystroppa/PSB2020>

Accompagn
ement
technique si
besoin

langages

C
C++
Fortran
Java

compilés

Python
Perl
Shell

interprétés

Html : langage tagué

JSON : description de données
XML

Javascript : interprété

Langage SQL

Bases de données

Mysql

MariaDB

Postgresql

Oracle

DB2

Sql server

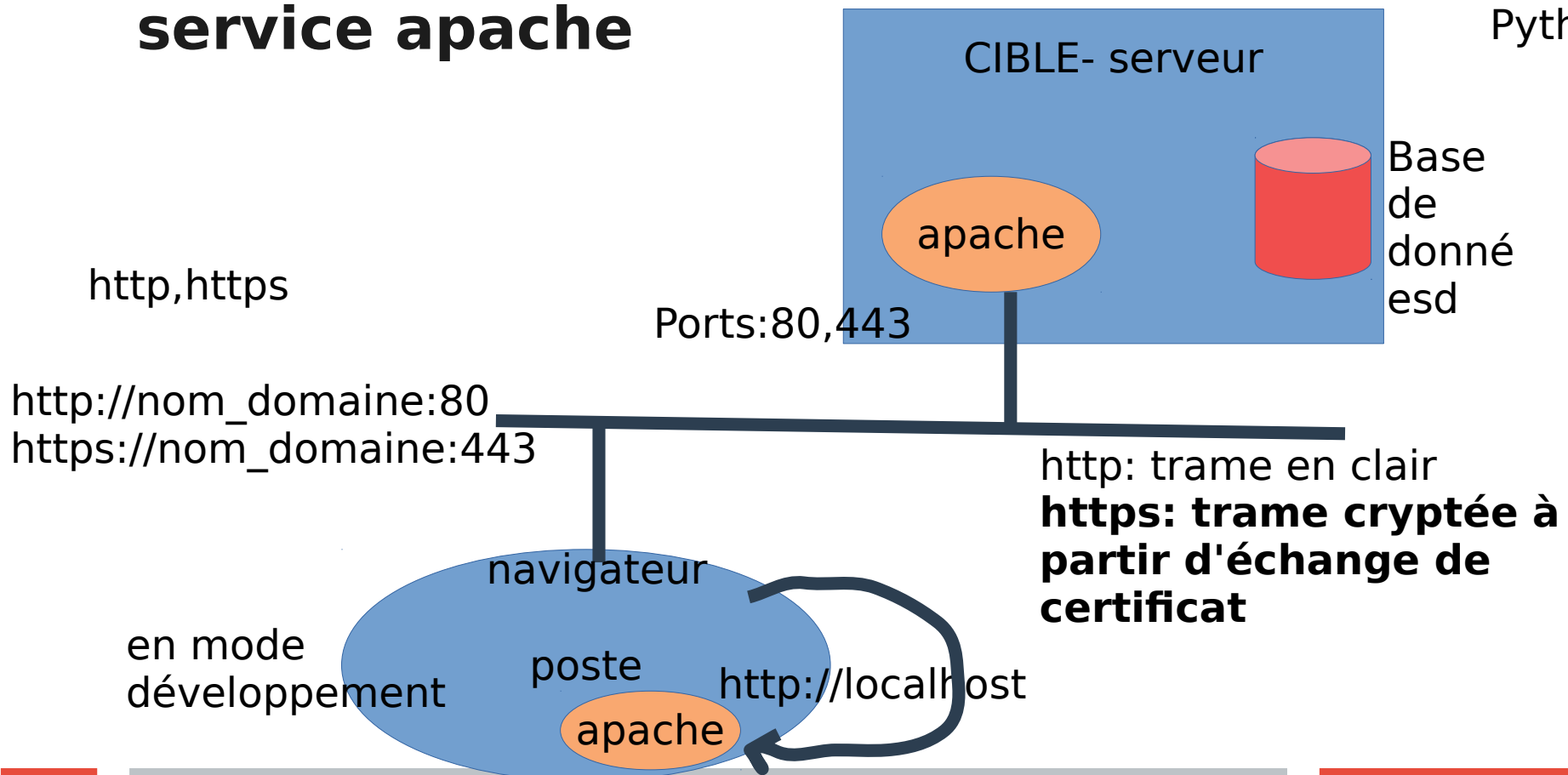
92

Services web / Architecture

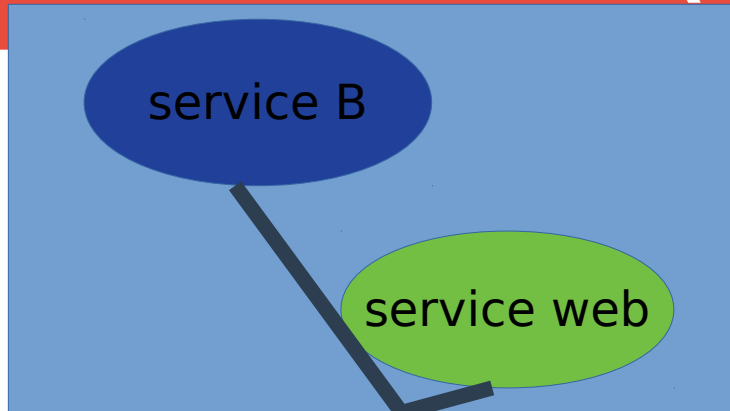
Service web : service apache

type de solution

- Apache
- IIS
- Tomcat
- Node
- Python



Architecture (suite)



serveur/machine distante chez un hébergeur

Service web : (http ou https)

service réseau : écoute des sollicitations sur le réseau à partir des ports 80 ou 443

contenir un ensemble d'éléments

développeur ==> (fichiers --- html --- css --- js --- autres)

network

trame/
demande

psbedu.org

1/ requête : url

portable

Navigateur qui interprète : important

- pas la maîtrise de son évolution
- plusieurs versions de navigateur (Chrome, Firefox, Safari, Opera, Edge)
- pas de maîtrise de la mise à jour (nouvelle version diffusée par l'éditeur)
- peut nécessiter des adaptations

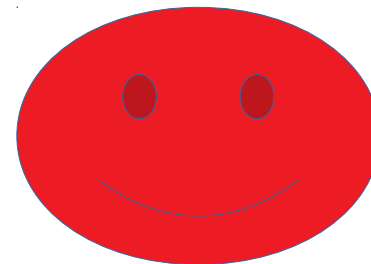


2/ retour du service web
index.html + css (visuel)
+ js + png + audio +
(dynamique /comportement)



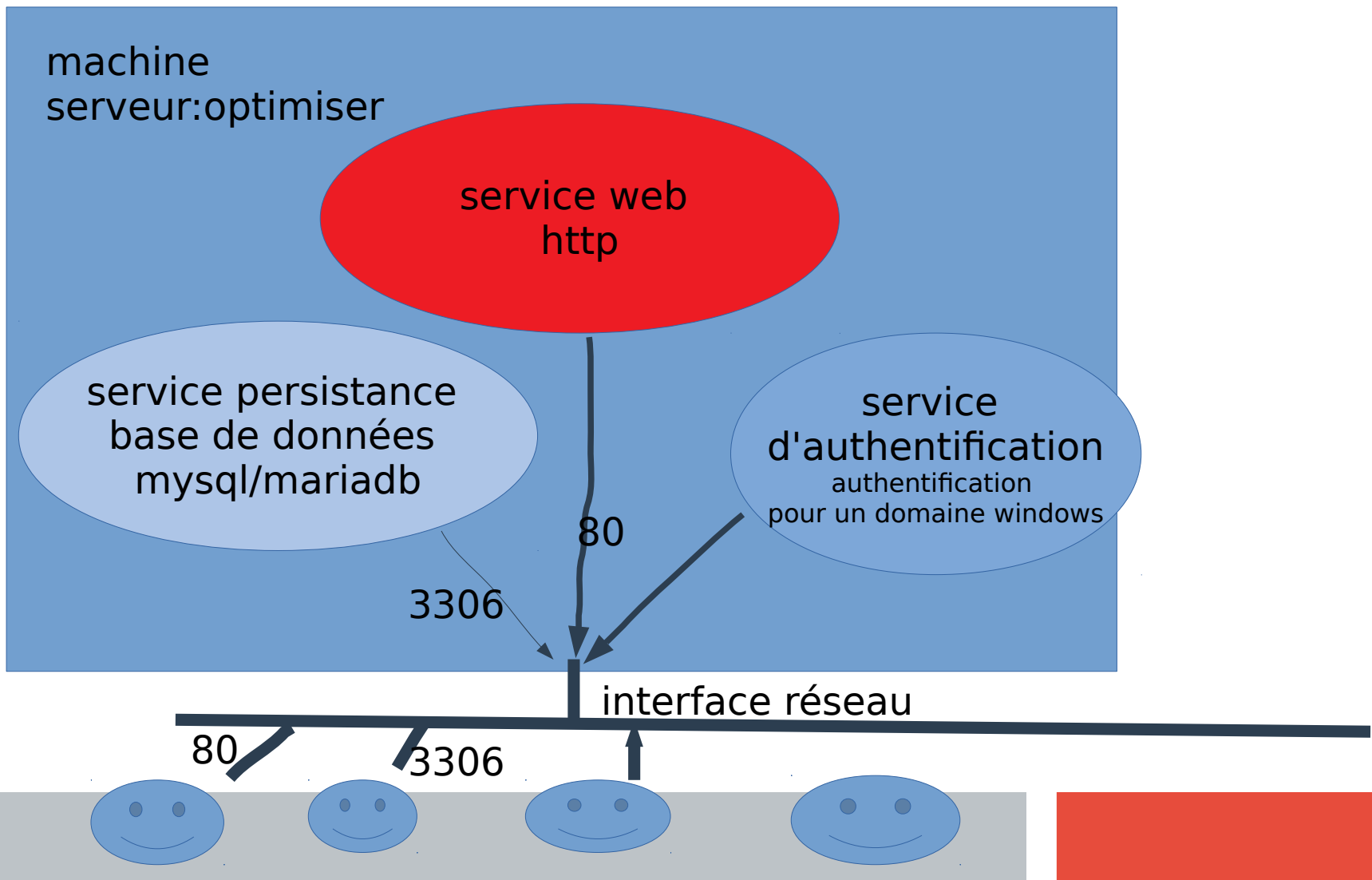
protocole

Dans le cadre d'une
communication entre
deux parties :
Protocole d'échange et
communication
http/https



règles

services



Application WEB : hébergée sur une machine (serveur ou en dev sur votre portable) accessible par la protocole http

Elle est composée :

HTML (documents ou fichiers html) inclusions : script de css
structure de vos pages web

CSS (fichiers de style)

js (fichiers de scripting en javascript)

la gestion des interactions et événementiel (dynamique sur le poste client (dans le navigateur))

autres ingrédients :

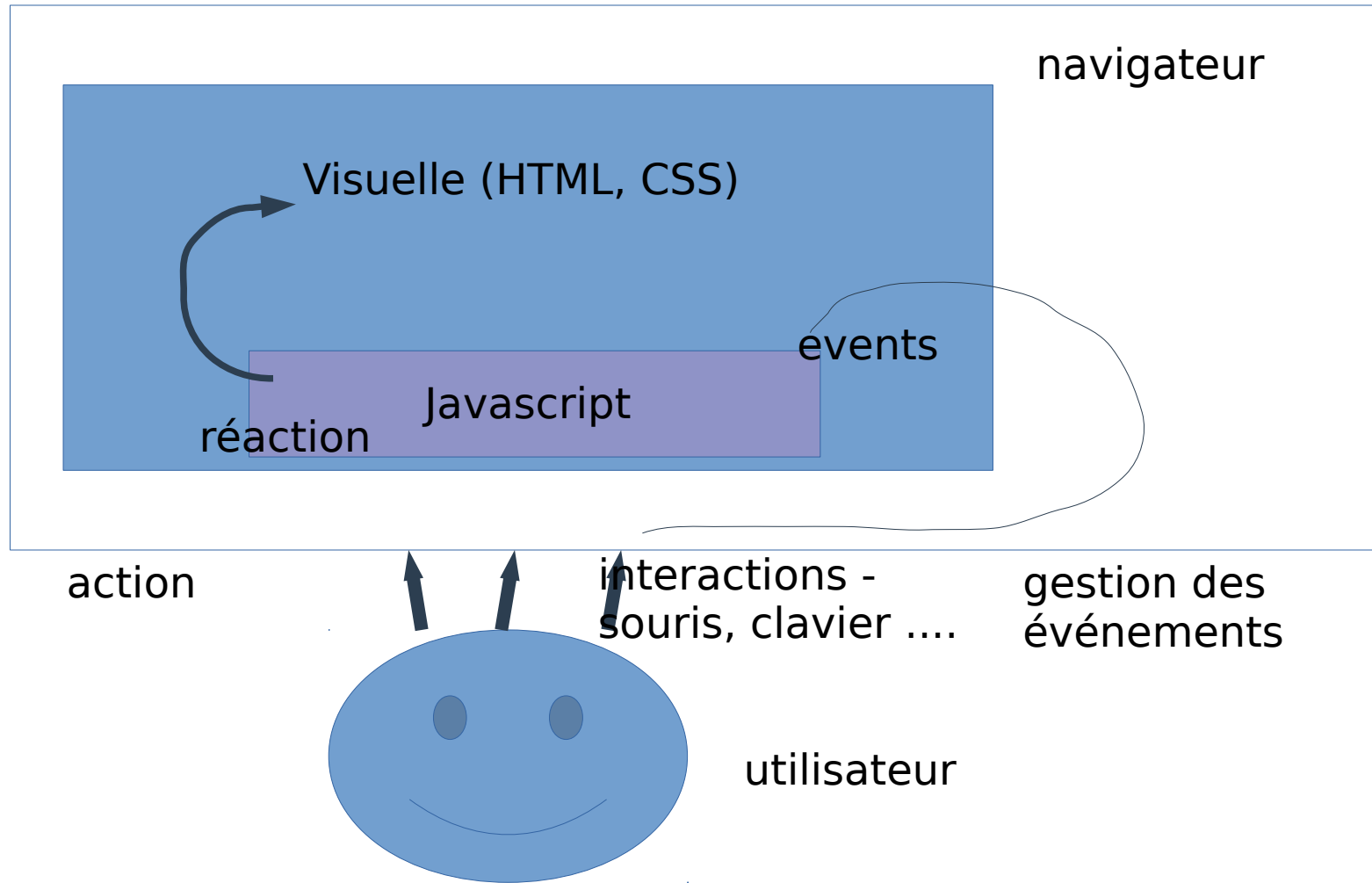
fichiers images

fichiers audios

fichiers vidéos

navigateur qui interprète
les tags HTML (doc html)
Les styles
interprète les scripts JS

Application - web



À installer sur vos machines

Nodejs (sur tous les systèmes)

Git (dispo sur toutes les machines)

Installer plusieurs navigateurs pour des tests

Langage informatique

A quoi sert un langage de programmation ?

Programmer des traitements en utilisant des séquences d'instructions du langage.

Que peut-on trouver dans un langage et de quoi a t-on besoin ?

Variables (de différentes natures)

jeu d'instructions

- Blocs d'instructions
- Modulaires

syntaxe à respecter
mots clés (spécifique au langage)

Algorithmie

Permet de décrire les étapes de son traitement :
utiliser les commentaires pour décrire les étapes
// première étape : récupérer les valeurs saisies
// deuxième étape : vérifier leur contenu
//

données en
entrée

décrire le contenu
de cette boîte

sorties

Langage informatique : Variable

Notion de variable :

Permet de stocker une valeur au cours de l'exécution du programme dans le but de pouvoir la reprendre, la modifier ou la détruire. La variable n'existe que lorsque le programme s'exécute.

Utilisation d'une variable :

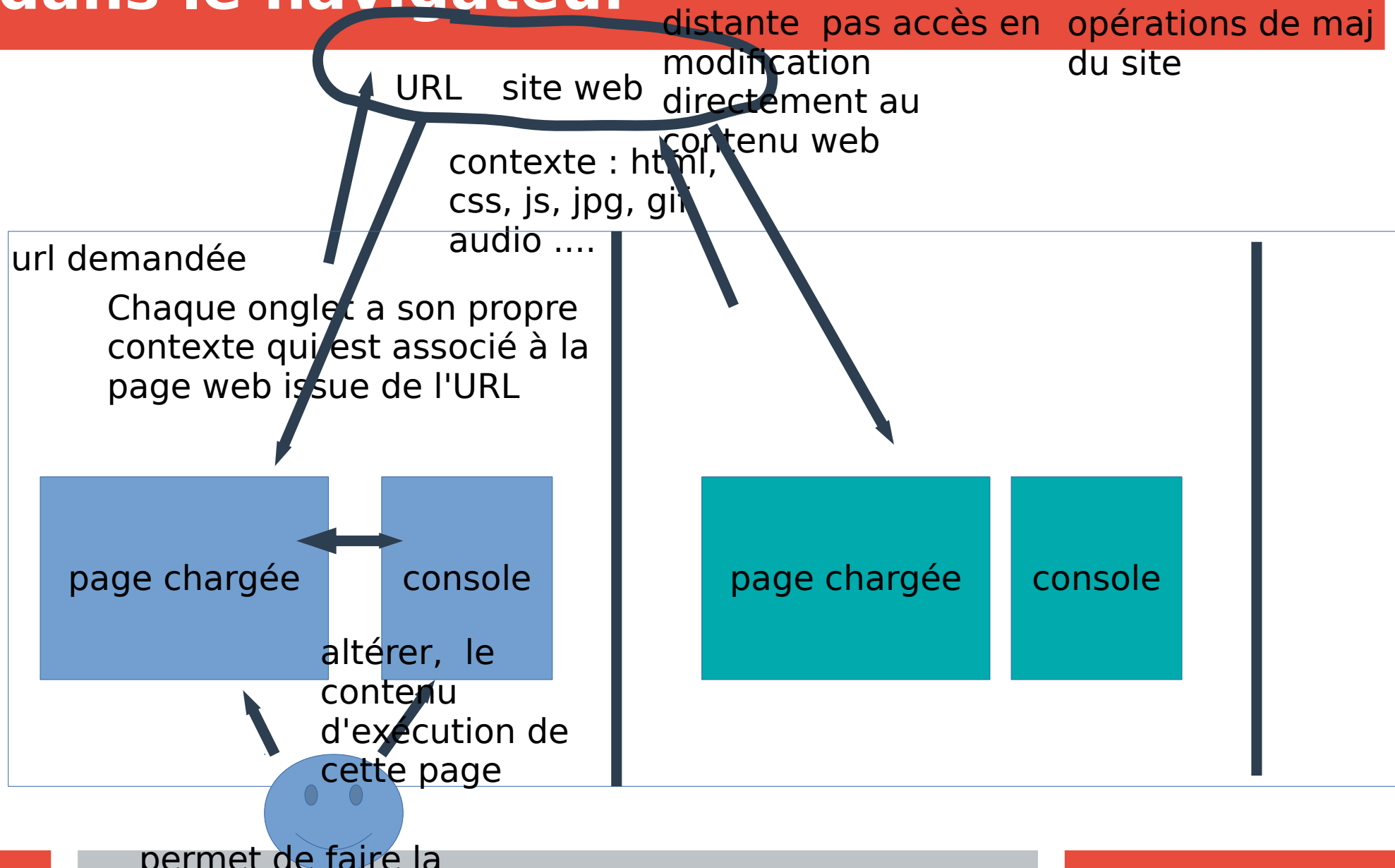
Allocation : déclaration explicite ou implicite

Utilisation : `variable1="voiture"`

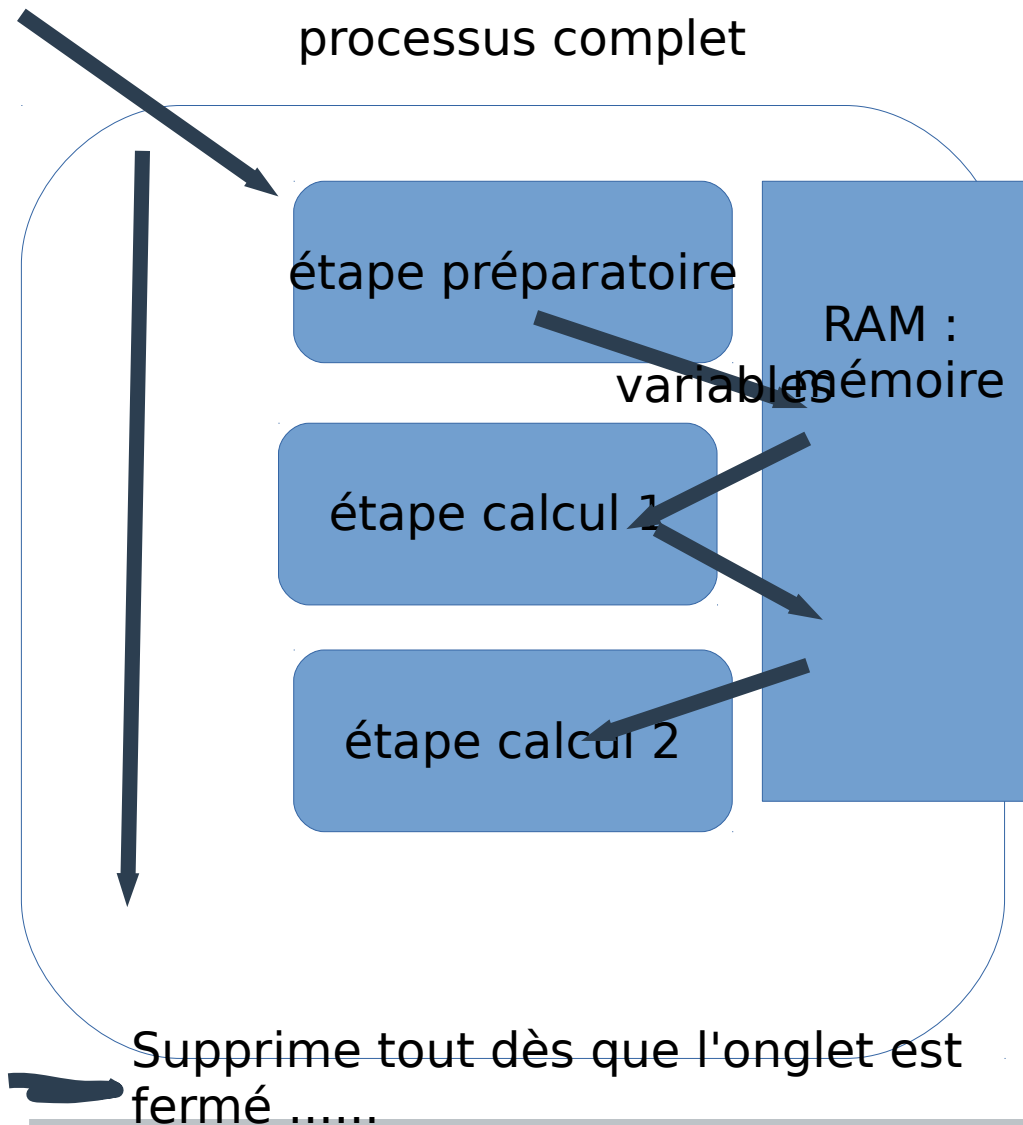
Désallocation : nécessaire dans certains langages de type c,c++, en Java il y a un mécanisme de libération des variables mémoires (Garbage Collector), en Javascript notion de delete, elles sont généralement déclarée en local.

dans le navigateur

nécessite un accès
particulier pour
effectuer des
opérations de maj
du site



traitement en étape



Définition et controle de type

Dans Javascript comme dans Python

Un déclaration implicite du type des variables

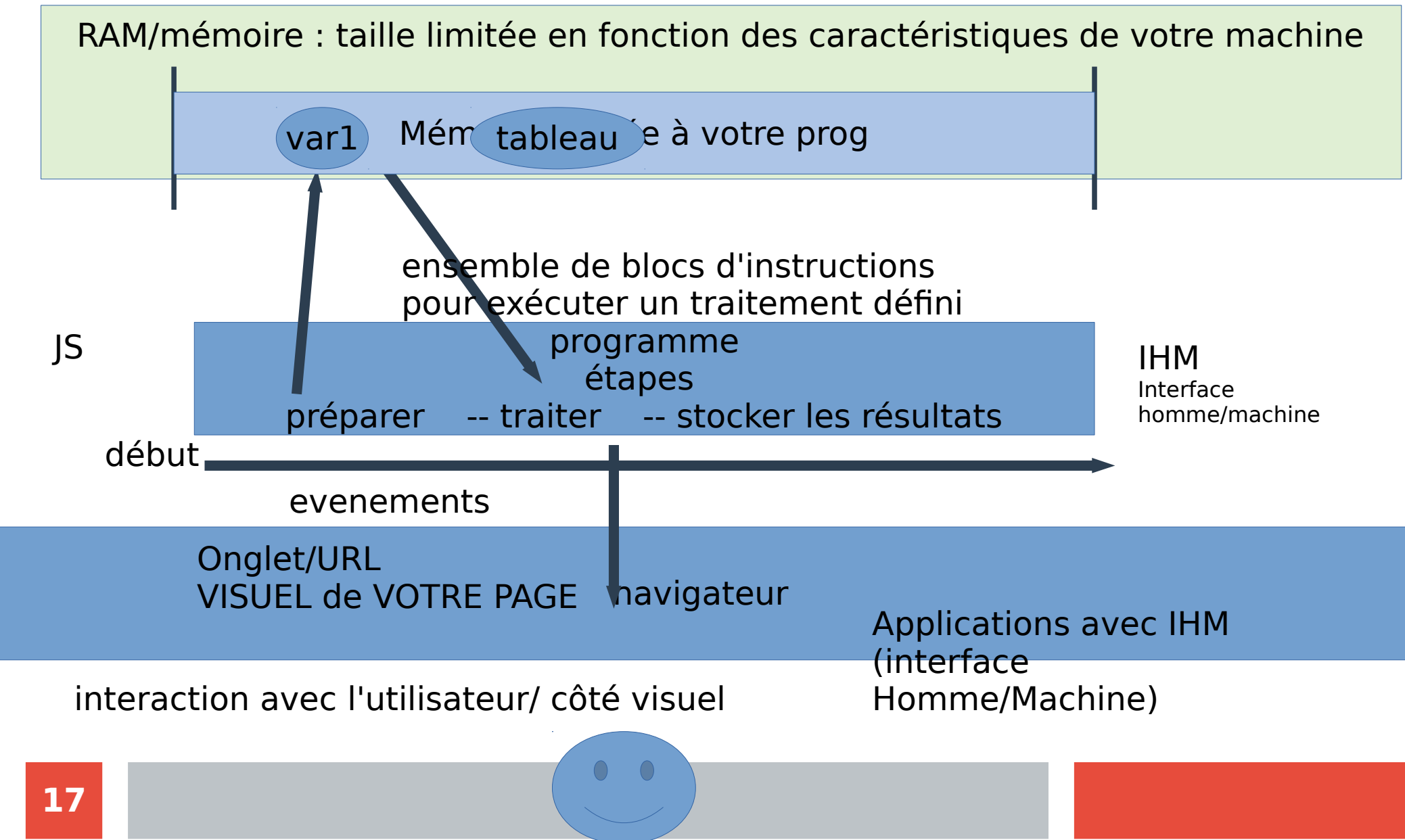
Il n'y a pas de contrôle implicite des types

Dans d'autres langages de type Java, C, C++, typescript

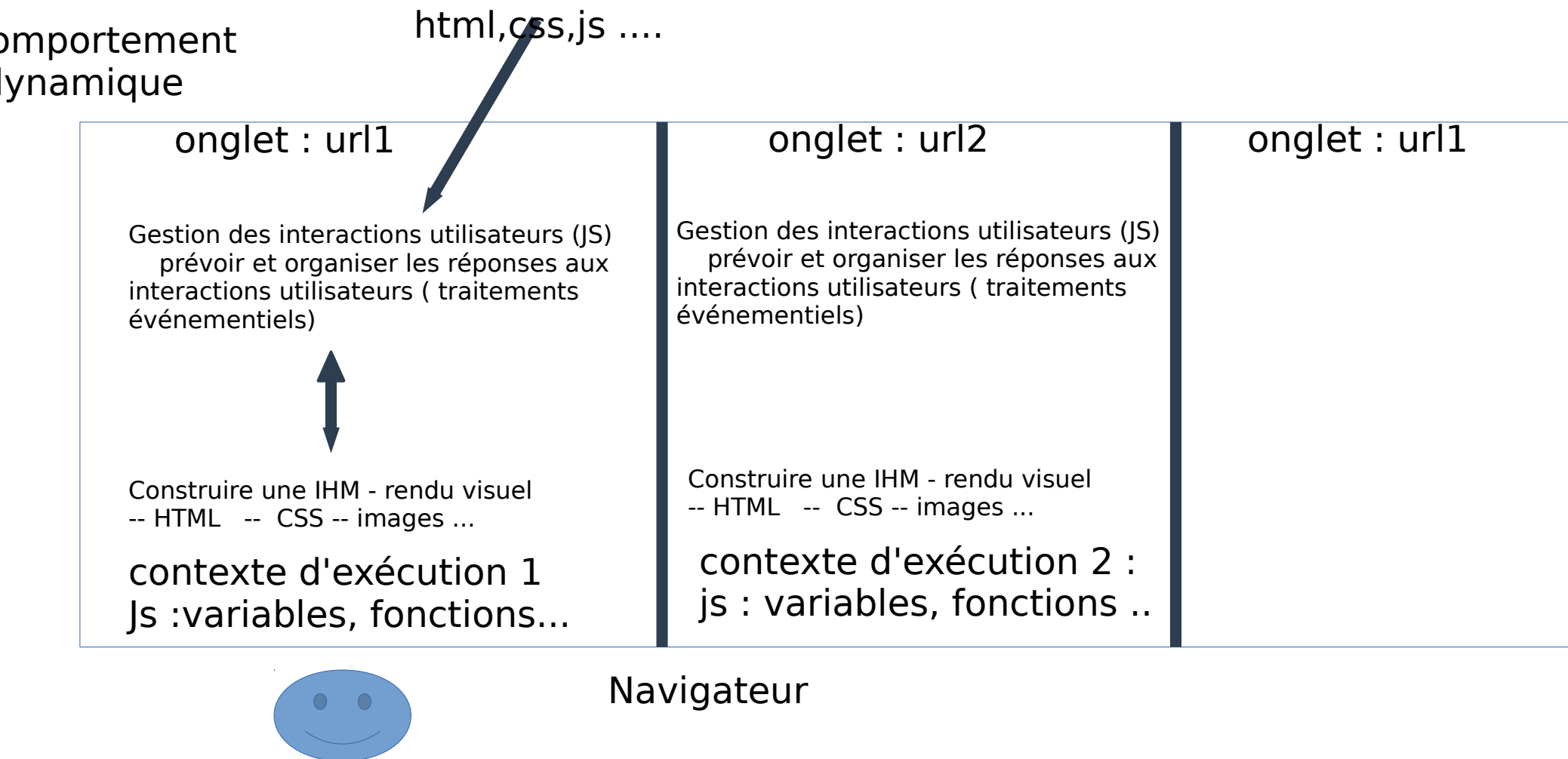
Déclaration explicite des types `public int age ;`

Contrôle des types au moment des affectations

Langage informatique : Variable



contexte technique de votre application web



structure d'une page HTML

<!DOCTYPE html>

<html>

<head>

<!-- déclarations de meta-données utilisés par les moteurs de recherche -->

<!-- inclusions de fichiers css, js --> <script src="http..."></script>

<!-- blocs style ou js -->

<style> </style>

<script> </script>

</head>

<body>

<h1> salut à tous </h1>

</body>

</html>

permet d'indiquer au navigateur la version html utilisée (version 5)

description des tags issue de XML

Mécanisme de type

**Déclaration d'une variable : pas de type
à l'aide de trois mots clés
var ou let ou const**

**Lors de l'affectation : défini un contenu
le contexte associe à la variable un type
(String, Number, Date, Array , Object)
Chaque type == définit un Objet (une classe)
qui possède des attributs et des fonctions
(méthodes prédéfinies)**

Différents types d'application

application en mode batch (lot)

input -- exécute -- output

fichier -- prog -- fichier résultat

Applications avec IHM

+ à gérer l'interaction avec l'utilisateur (prévoir anticiper orienter le comportement de l'utilisateur)

applications en standalone (installer sous système d'exploitation)

applications légères/ en ligne : elles s'appuient sur le navigateur (les applications Online qui sont les plus utilisées)

Déclaration des variables et types de variables

Dans certains langages vous avez obligation de déclarer et de typer vos variables (c,c++,Fortran, Java ...) ce sont les langages fortement typés. (voir des exemples)

Dans d'autres, il y a un typage implicite, ce qui veut dire c'est en fonction de la valeur que l'on stocke que le système associe le type à votre variable (e : python, javascript, R, Matlab ...)

Structure de variables

Construction

Etre capable de les définir (de les construire)

Manipulation

Etre capable d'extraire les informations

Etre capable de les modifier ou de la compléter

Blocs d'instructions

affichage dans la console (afficher différentes valeurs dans la console)

blocs d'instructions conditionnels

condition : si valeur faire

```
if (condition) {  
  } else {  
  }
```

```
if (condition1 && condition2)    ET  
{
```

```
}
```

```
if (condition1 || condition2) {    OU
```

```
}
```


blocs conditionnels

notion de switch (valeur)

```
switch (valeur) {  
    case 1 :  
        //instructions pour cas 1  
        break ;  
    case 2 :  
        //instructions pour cas 2  
        break ;  
    case 3 :  
        //instructions pour cas 3  
        break ;  
  
    default :  
        break ;  
  
}
```

blocs itératifs : faire des boucles

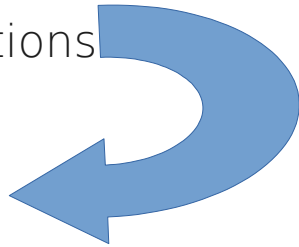
bloc de type for

on commence à début et on finit à fin (itération explicite)

```
for (let i=0 ; i< 10 ; i++) {
```

// blocs d'instructions

```
}
```



```
while (condition) {
```

```
}
```



ne pas omettre la condition de sortie sinon boucle infinie

$i++ \implies i=i+1$


```
i=0 ;  
while (i<100) {  
  // instructions  
  console.log(i) ;  
  i=i+1 ;  
}
```

définition de fonction

Factoriser et réutiliser des blocs d'instructions

Evite d'avoir de la redondance/duplication de blocs d'instructions

difficilement réutilisation
et modifiable



Instructions
bloc 1
instruction

instruction ...
bloc1
instruction

"factoriser"

instructions
bloc1() ;
instruction ...
bloc1() ;

Définition d'une fonction
function bloc1() {
instruction 1
.....
}

réutilisation -- test unitaire
(dans le cadre de la
validation)

Javascript et Html

Enjeux et interactions

Comment se passe les échanges entre les deux parties d'une même application

la partie HTML : permet de décrire le visuel (IHM) par l'intermédiaire de tag

`<div>`, `<p>`, `<li,ul,ol>` `<article>` `` `<form>` ``

Chaque tag peut posséder des attributs qui vont permettre de lui apporter des caractéristiques supplémentaires :

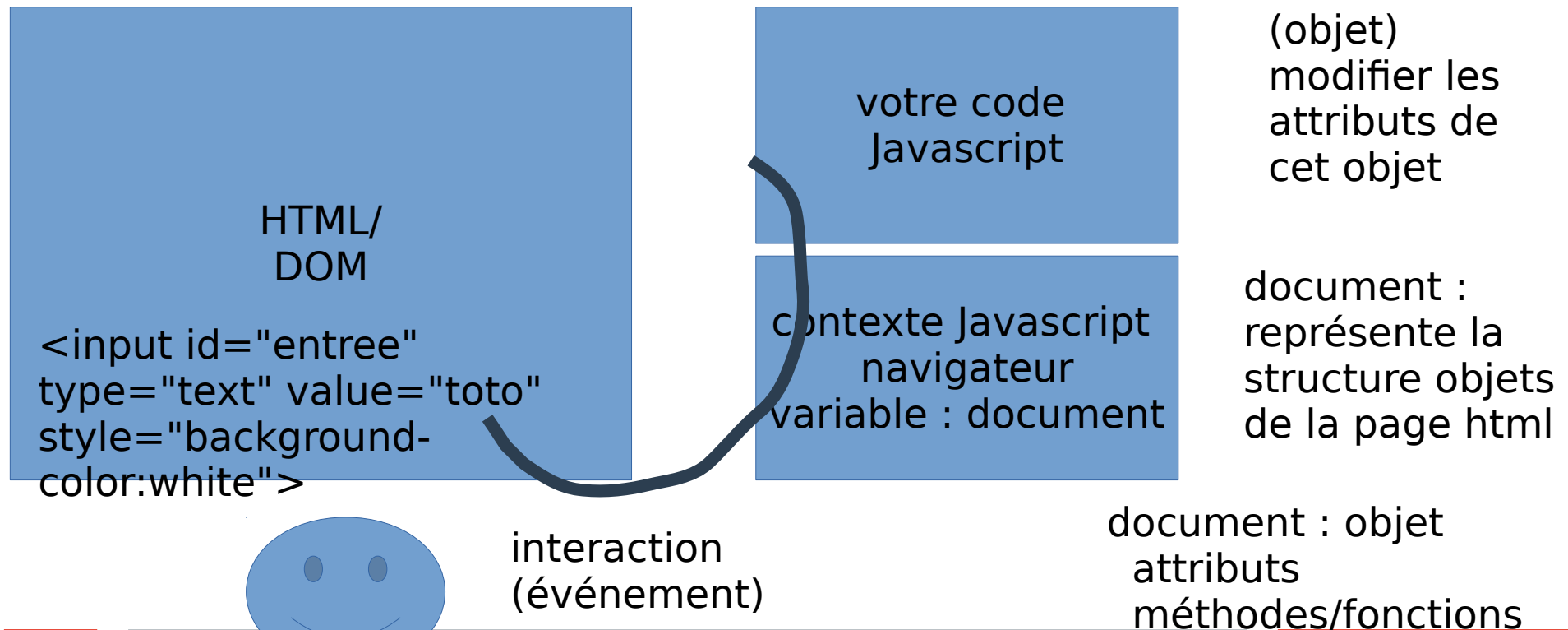
de type style, contenu, identification, ... pour récupérer ensuite à partir de javascript les valeurs on pourra utiliser les méthodes associées à document de type `document.getElementById("identification")`.

structure HTML

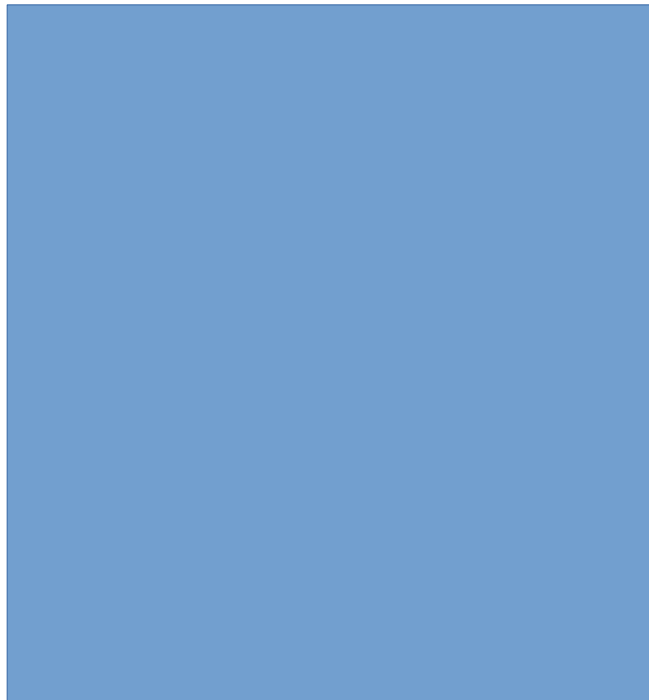
HTML est associé une DOM (Document Object Model) -- s'appuie sur des tags

Les tags `<body>` `<div ...>` `<p>` `<input ...>`

Structure l'information à partir des tags



structure DOM



identifier l'objet sur
lequel on veut travailler
effectuer les opérations
sur cet objet



- extraction
- modification
- suppression

- ajout d'enfants

action utilisateur --> réaction de l'application



Rappel évaluation projet et organisation

Evaluation des projets

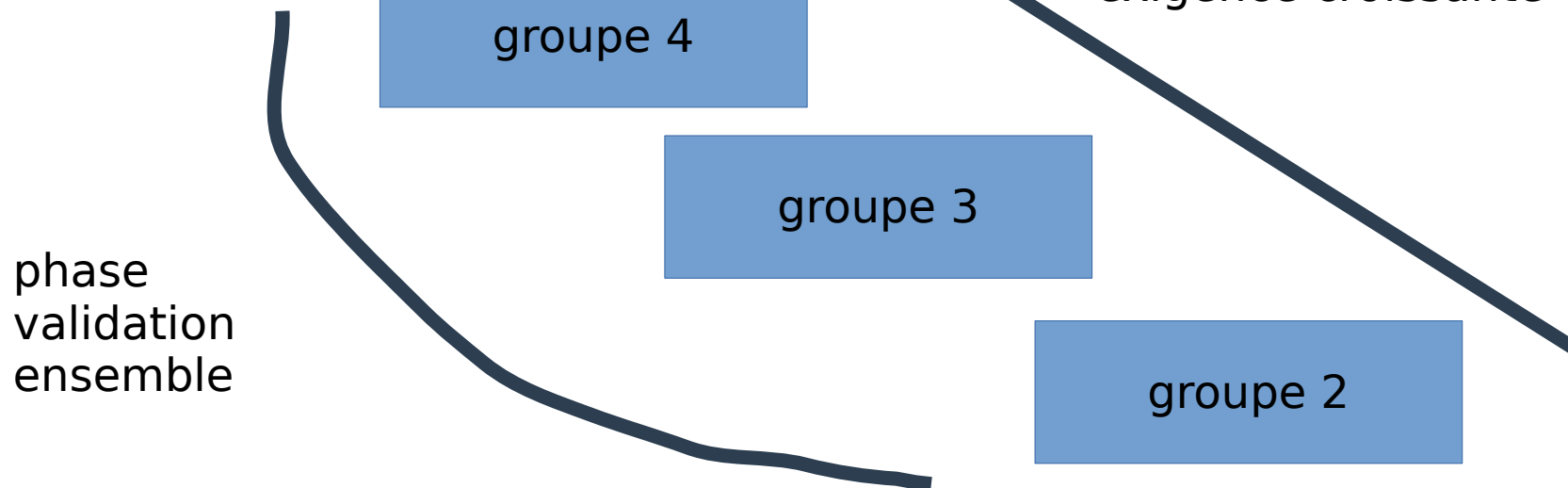
deux phases :

Rapport

Test de la réalisation

quantité / complexité

exigence croissante



Réalisation d'un projet

Vue globale -- description la plus précise possible des fonctionnalités

Découper en lots : organisation/répartition

Description fonctionnelle et ses interfaces avec les autres

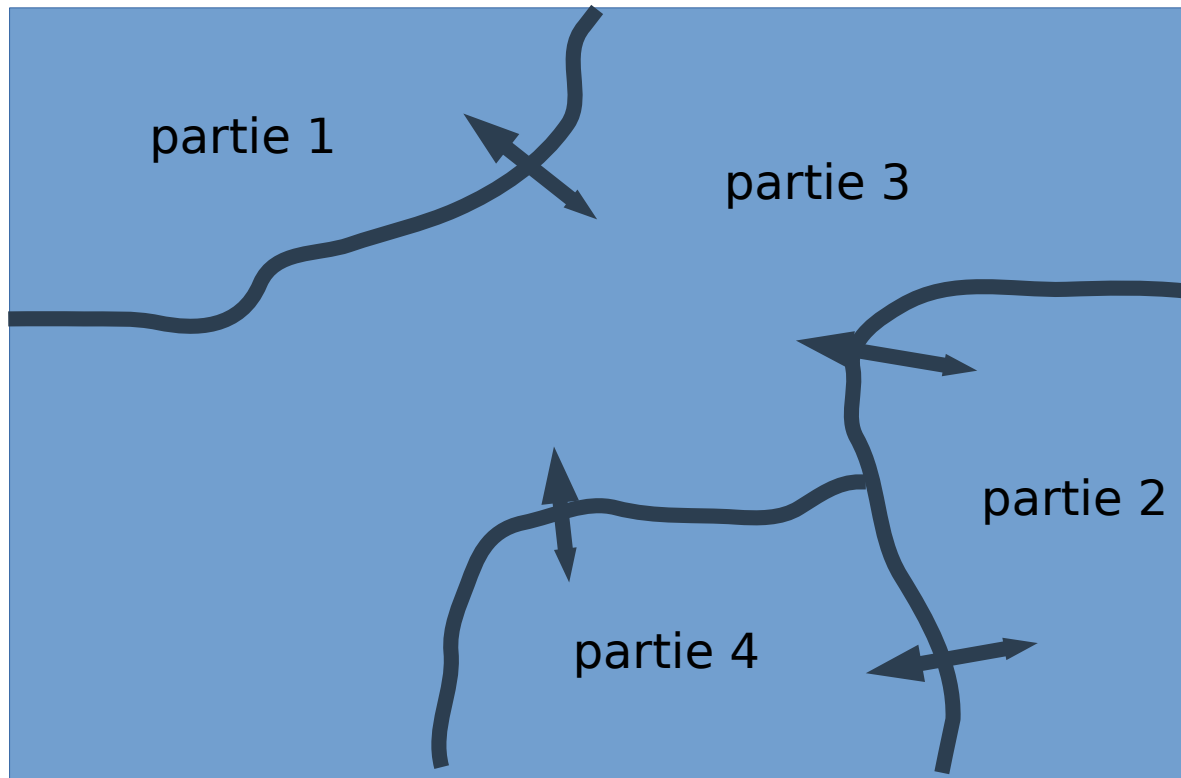
Développer les lots /mettre au point /tester

de façon unitaire (javascript -- modulaire -- fichiers dédiés js) --
vérifier compatibilité pour toutes les navigateurs

Phase d'intégration qui consiste à ramener les lots dans leur contexte cible d'exécution

Tests globaux

problème



Déléguer et répartir
travail en groupe

décomposition en partie /lot
décrire les interfaces entre les
différentes parties