

Principes d'utilisation des systèmes de gestion de bases de données

Rappels : Internet, HTML, PHP

M1, Informatique
Emmanuel Waller, LRI, Orsay

Rappels : internet, HTML, PHP

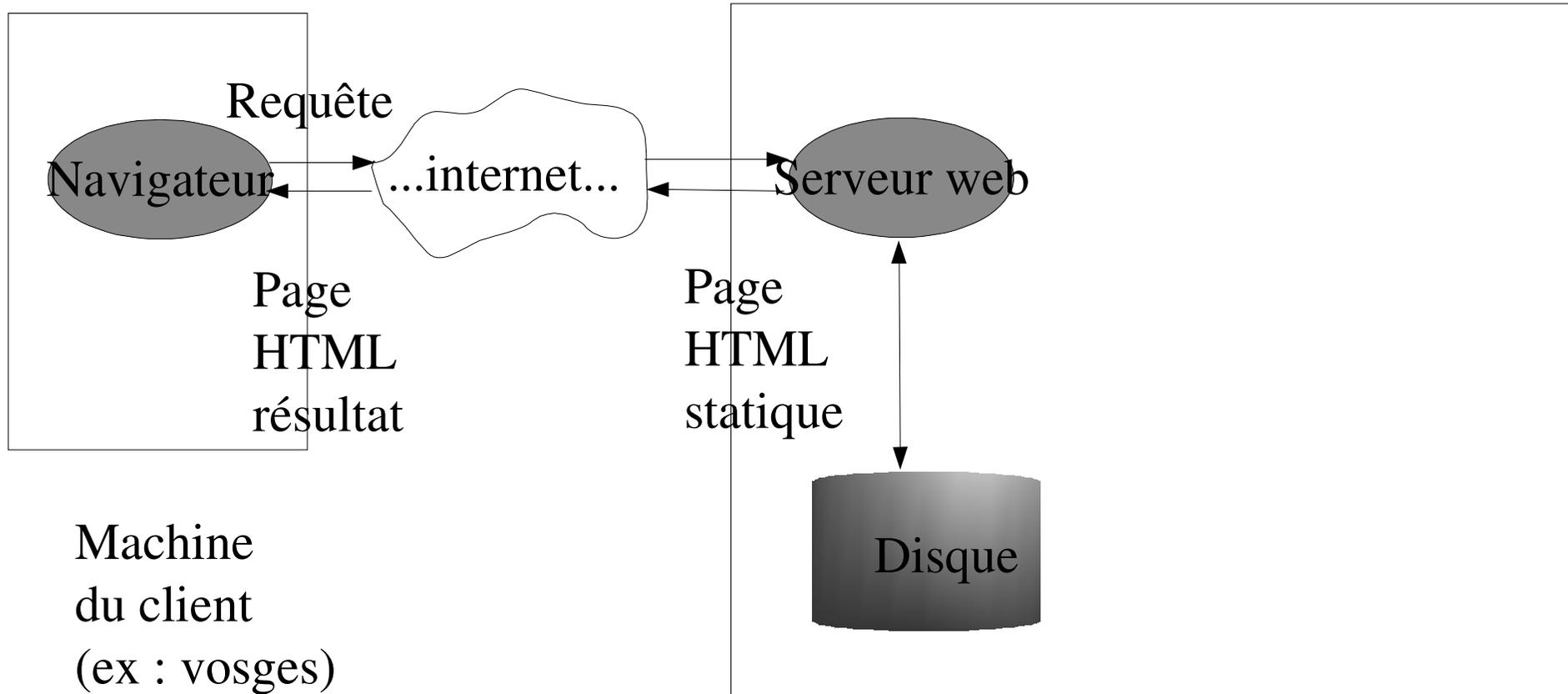
- Clients et serveurs web
- HTML
- PHP

Clients et serveurs web

principe

- Machine stocke fichiers
serveur (logiciel) sur cette machine les fournit à l'extérieur sur demande
- Réseau (internet) relie toutes les machines (physique, adressage, etc.)
- Client (ex : navigateur) lance requête sur réseau =
 - Adresse serveur
 - Nom (chemin) fichier
- Serveur renvoie le fichier demandé
- Le navigateur l'affiche (possible cas plus général)

Architecture



Machine
du serveur web :
ssh1

démonstration

- Exemples ie2
- toto.txt
- View page source
- Erreurs possibles ?
 - Coquille requête
 - Contenu : indifférent

Rappels : internet, HTML, PHP

- Clients et serveurs web
- HTML
- PHP

HTML

- permet de présenter du texte de manière attrayante
 - présentation (couleurs, polices, etc.) définie par des «balises»
- Liens hypertexte
- Saisie de données et actions (mail, exécution d'un programme – ex : php)

Principe (intuition)

- Fichier texte (comme Ex.java) contenant un « programme HTML »
- Un navigateur interprète (évalue) ce programme :
 - Le résultat est l'affichage sur l'écran de « texte » présenté comme décrit dans la page

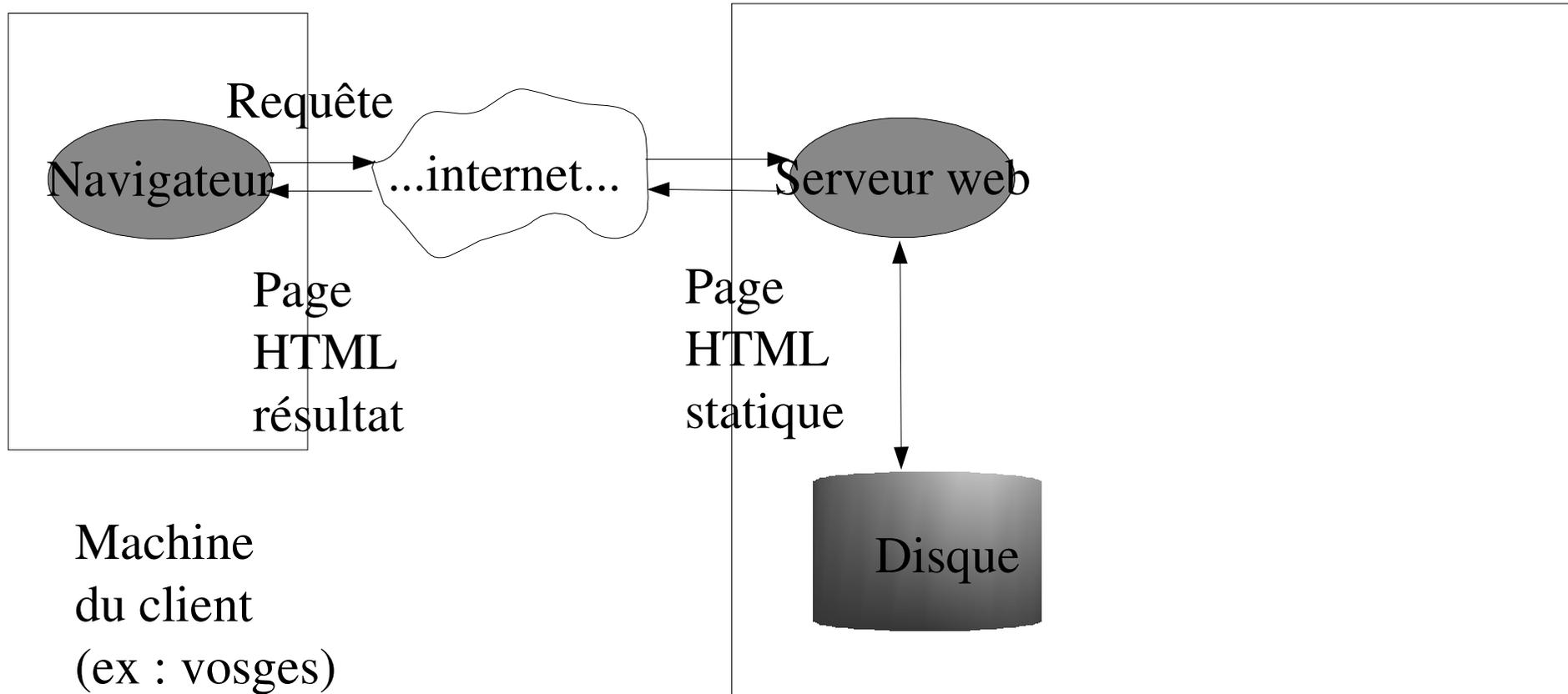
Exemple : première page en HTML

```
<html>  
  <body bgcolor=green>  
    <h2>ma page</h2><br>  
    <blink>bonjour</blink><br>  
    figurez-vous que ce matin en cours...  
  </body>  
</html>
```

démonstration

- <http://www.lri.fr/~waller/cours/xsgbd/cours1/ex1.html>
- view page source
- Erreurs possibles ?
 - Coquille requête
 - Contenu : HTML mal formé

Architecture : la même



Machine
du serveur web :
ssh1

exemple : saisie de données et déclenchement de programme

```
<html>
```

```
<body>
```

```
<form method="post" action="ex3.php">
```

Entrez un kilometrage :

```
<input type="text" name="x">
```

```
<input type="submit" value="declencher">
```

```
</form>
```

```
</body>
```

```
</html>
```

démonstration

- <http://www.lri.fr/~waller/cours/xsgbd/cours1/ex1.html>
- view page source
- cliquer sur « déclencher »
déclenche l'exécution du programme ex3.php (vu ult.)
- Erreurs possibles ? Les mêmes

Autres exemples

- Page du cours
- Autres utilisés dans suite du cours

Rappels : internet, HTML, PHP

- Clients et serveurs web
- HTML
- PHP

PHP

- est un langage généraliste (objets, fichiers, exceptions, bibliothèques, etc.)
- Il s'intègre avec HTML

PHP : c'est quoi ?

- PHP (« Personal Home Page »)
- 1994
- Logiciel libre
- S'intègre avec HTML
- Pour programmation côté serveur : interprété par le serveur (ex : Apache)
- Destiné à la production de pages HTML générées dynamiquement

principe

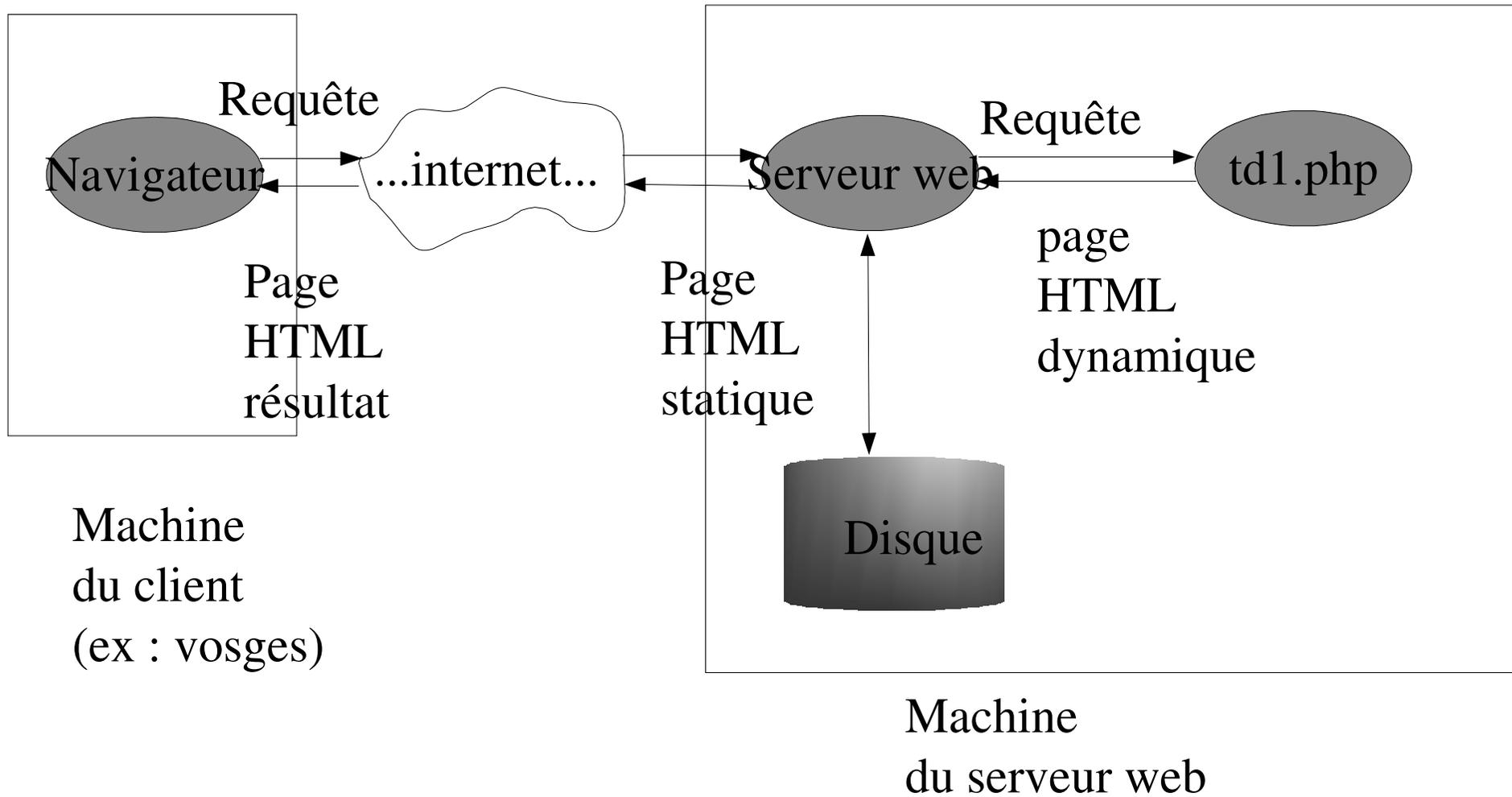
- Fichier texte (comme Ex.java) contenant un programme PHP
- L'interpréteur PHP :
 - Évalue le programme (dont effets de bord)
 - Renvoie le résultat
 - Ce résultat est un fichier texte (éphémère)
- Ce résultat peut donc être affiché par un navigateur
- Le fichier résultat est construit lors de l'exécution, il est donc appelé « page web dynamique »

Exemple : premier programme PHP

```
<?php  
    echo "Bonjour Monde !";  
?>
```

- Démonstration + view page source
- Rem : aucun intérêt car pas dynamique
- Rem : pas de HTML
- Erreurs possibles ? Les mêmes

Architecture

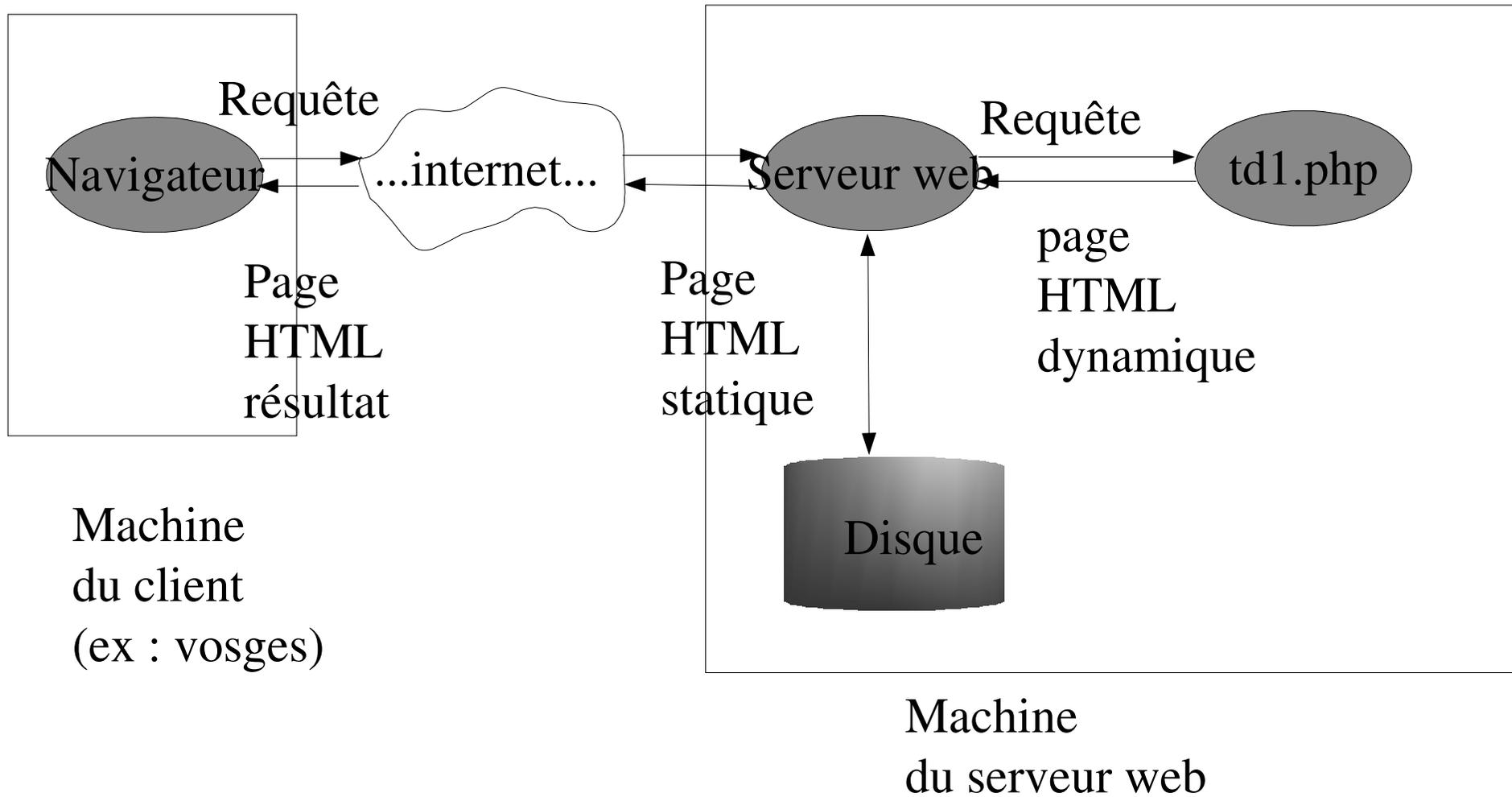


Exemple : dynamique

```
<?php  
    $x = 7;  
    echo $x;  
?>
```

- Démonstration + view page source
- Rem : page web dynamique
- Rem : pas de HTML
- Erreurs possibles ? Les mêmes

Architecture



Exemple : paramètre

```
<?php
    $x = $_REQUEST['y'];
    echo $x;
?>
```

- Démonstration : passage de y dans l'URL + view page source
- appelé avec y valant 7, affiche : 7
- le programme ex3.php a reçu variable y en param.
- Rem : pas de HTML
- Erreurs possibles ? Les mêmes + essayer : pas de y

paramètres depuis formulaire

- ex1.html : formulaire :
 - action='ex2.php'
 - input : name='y'
- Démonstration : appel depuis formulaire HTML
+ view page source
- Erreurs possibles ? Les mêmes

Exemple plus complexe

- Afficher une phrase contenant des entiers résultant de calculs
- Plusieurs parties PHP dans le fichier
- Rem : ne dépend pas de HTML

```
<html><body>
```

Voici

```
<?php
```

```
    $x = 4;
```

```
    echo $x;
```

```
?>
```

```
<br> puis
```

```
<?php
```

```
    echo(« <h2> »);
```

```
    echo($x + 1);
```

```
    echo(« </h2> »);
```

```
?>
```

```
!
```

```
</body></html>
```

Principe d'exécution d'un fichier PHP

- Requête : `http://www.lri.fr/~waller/ex1.php`
- Le préprocesseur PHP procède comme suit
- Parcours le fichier
- Chaque fragment de code PHP est :
 - Reconnu
 - Évalué
 - Remplacé par son résultat (= tout ce qu'il a écrit sur sa sortie standard : echo)
- Le fichier temporaire ainsi construit est envoyé au client
- Le client :
 - Affiche le fichier dans fenêtre navigateur
 - Si pas HTML correct : il y affiche une erreur

Fichier résultat

- Construction au tableau
- Premier fragment PHP : 4
- Deuxième fragment PHP : `<h2>5</h2>`
- Fichier construit (sauf sauts ligne):

```
<html><body>
```

```
Voici 4 <br> puis <h2>5</h2> !
```

```
</body></html>
```

- Affichage navigateur (avec le « 5 » grand) :

Voici 4

puis 5 !

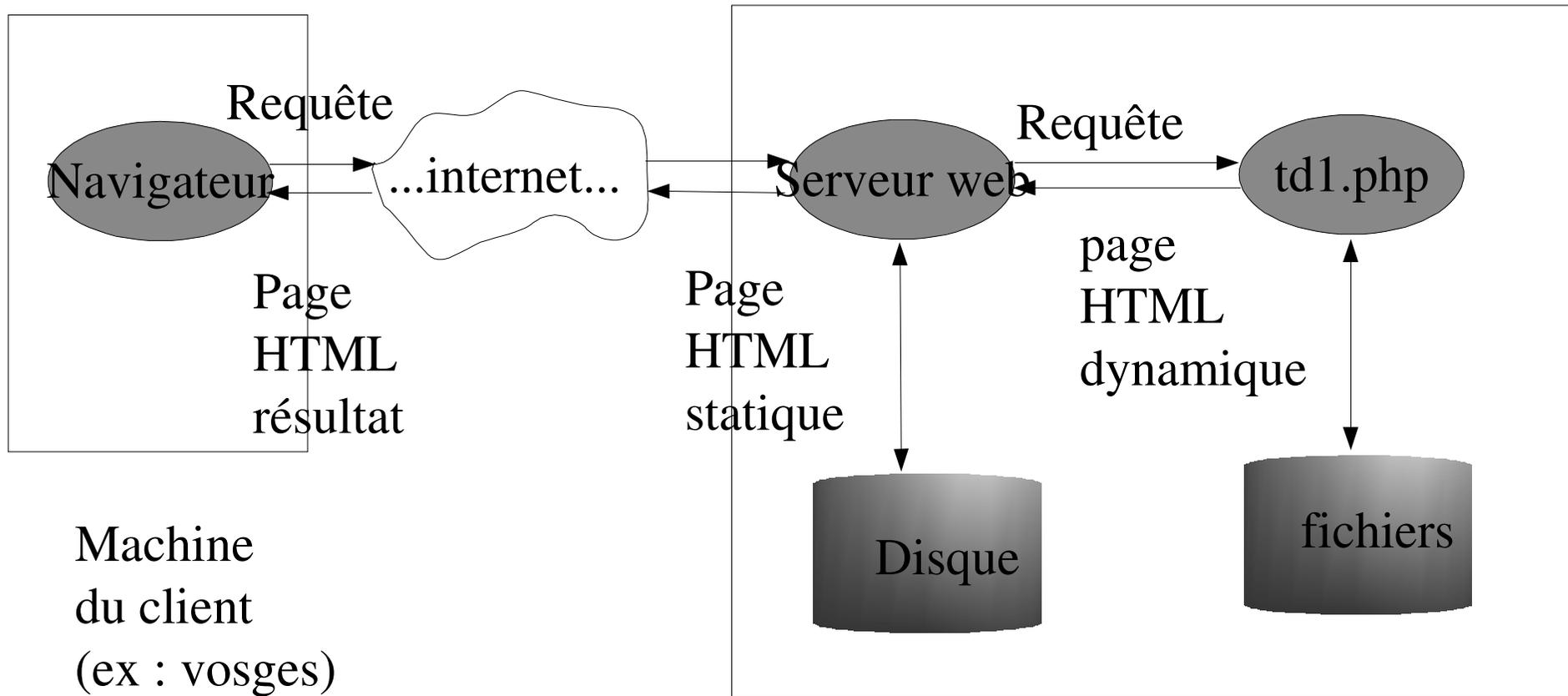
- Démonstration + view page source

Le langage

- appel depuis formulaire/url récupération des paramètres
- bloc, if, for, while, fonctions
- Variables : non déclarées, non typées : `$x = 7`
- Tableaux : taille dynamique : `$y[0]=3; $y[1]=7;`
 - directement : `$t[0]=4; echo $t[$i];`
- Exceptions, objets, fichiers, bibliothèques, etc.
- Fichiers séparés : `include « titi.php »`

```
if ($x == 7) { /* pas de booléens : 0, etc. */
    $x++;
    echo « x= » . $x;
}
else echo « ok »;
for ($i=0; $i<7; $i++)
    echo $i;
function p($x) {
    $y=$x+1;
    return $y;
}
function p($z) {
    echo $z-1;
}
$x=f(3); echo f(7); p(4);
```

Architecture



Machines
des serveurs web et
fichiers (2 machines)

Rappels : internet, HTML, PHP

- Clients et serveurs web
- HTML
- PHP