



PHP *Avancé*

NAMESPACE (*PHP 5.3*)

PROBLEMATIQUE

- **Utilisation de bibliothèques**
applications existantes, frameworks,
- **Collision des identificateurs**
constantes, classes, fonctions
- **Nommage**
 - *prefixe : mysql_xxxx*
 - *longueur des identificateurs*
pas de possibilité d'alias

SOLUTION

- **Qualification des définitions (constante, classe, méthode ou fonction)**
regroupement (≈ dossier et fichiers)
- **Espace de nom (namespace)**
 - *espace de nom global*
 - *espace de nom courant*
 - *sous espaces de noms*
- **Référence à un élément**
 - *absolue*
 - *relative*
- **Existe dans d'autres langages**
 - C++ (namespace)
 - Java (package)
 - .NET (namespace)

DEFINITION D'UN ESPACE DE NOM

(V1)

```
<?php
namespace Scolarite;

const ETUDIANT_RECU = 1;
class Etudiant
{
    /* ... */
}
function evaluer ()
{
    /* ... */
}
?>
```

- **Création de** la constante : *Scolarite\ETUDIANT_RECU*,
la classe : *Scolarite\Etudiant*,
la fonction : *Scolarite\evaluer()*

DEFINITION D'UN ESPACE DE NOM

(V2)

```
<?php
namespace Scholarite
{
    const ETUDIANT_RECU = 1;
    class Etudiant
    {
        /* ... */
    }
    function evaluer ()
    {
        /* ... */
    }
}
?>
```

CONTRAINTES

- **Première instruction du fichier**
avant même code html
- **Scindable sur plusieurs fichiers**
- **Antislash (\) et non slash (/)**
- **Pas de point (.)**

SOUS ESPACE DE NOM

(V1)

```
<?php
namespace Sclarite\Metier;

const ETUDIANT_RECU = 1;
class Etudiant
{
    /* ... */
}
function evaluer ()
{
    /* ... */
}
?>
```

- **Création de** la constante : `Sclarite\Metier\ETUDIANT_RECU`,
la classe : `Sclarite\Metier\Etudiant`,
la fonction : `Sclarite\Metier\evaluer()`

ESPACE DE NOM GLOBAL

(V1)

```
<?php  
  
const ETUDIANT_RECU = 1;  
class Etudiant  
{  
    /* ... */  
}  
  
function evaluer()  
{  
    /* ... */  
}  
  
?>
```


ESPACE DE NOM GLOBAL

(V2)

```
<?php
namespace
{
    const ETUDIANT_RECU = 1;
    class Etudiant
    {
        /* ... */
    }
    function evaluer ()
    {
        /* ... */
    }
}
?>
```

UTILISATION

(Fichier *Etudiant.class.php*)

```
<?php
namespace Scolarite\Metier;

const ETUDIANT_RECU = 1;
class Etudiant
{
    /* ... */
}

function evaluer ()
{
    /* ... */
}

?>
```

UTILISATION

(Fichier index.php)

```
<?php
namespace Sclarite;
require_once 'Etudiant.class.php';

/* Redéfinition dans l'espace de nom Sclarité */
const ETUDIANT_RECU = 2;
class Etudiant
{
    /* ... */
}
function evaluer()
{
    /* ... */
}
...

```

UTILISATION

(suite du fichier index.php)

/ on est dans l'espace de nom **Scolarite** */*

/ Référence non qualifiée (nom relatif) */*

```
evaluer ();  
echo ETUDIANT_RECU;  
$obj1 = new Etudiant();
```

Devient

```
Scolarite\evaluer();  
echo Scolarite\ ETUDIANT_RECU;  
$obj1 = new Scolarite\Etudiant();
```

/ Référence qualifiée (chemin relatif) */*

```
Metier\evaluer ();  
echo Metier\ETUDIANT_RECU;  
$obj2 = new Metier\Etudiant();
```

Devient

```
Scolarite\Metier\evaluer();  
echo Scolarite\ Metier\ETUDIANT_RECU;  
$obj2 = new Scolarite\Metier\Etudiant();
```

/ Référence absolue (chemin absolu) */*

```
\Scolarite\Metier\evaluer ();  
echo \Scolarite\Metier\ETUDIANT_RECU;  
$obj3 = new \Scolarite\Metier\Etudiant();
```

Devient

```
Scolarite\Metier\evaluer();  
echo Scolarite\ Metier\ETUDIANT_RECU;  
$obj2 = new Scolarite\Metier\Etudiant();
```

?>

RETOUR A L'ESPACE DE NOM GLOBAL

- **Hors espace de nom**
 - *déclarations rattachées à l'espace de nom global*
 - *appel relatifs à l'espace de nom global*
- **Dans un espace de nom**
(*rappel : déclarations rattachées à l'espace de nom courant*)
 - *appel préfixé d'un *
 - *noms non qualifiés non résolus dans l'espace de nom courant*

RETOUR A L'ESPACE DE NOM GLOBAL

```
<?php
namespace
{
    function calcul($val)
    {
        return $val + $val;
    }
    function afficher($mesg)
    {
        echo "<h2>" . $mesg . "</h2>";
    }
}
namespace Sclarite
{
    function calcul($val)
    {
        return $val * $val;
    }

    afficher(calcul(12));
    afficher(\calcul(12));
}
?>
```

Affiche 24

Constante `__NAMESPACE__`

- **Chaîne renvoyant l'espace de nom courant**
structure sociale caractérisée par une circulation d'informations (informations produits, clients, fournisseurs, ...)
- **Vide pour l'espace de nom global**
- **Equivalent de `self::` pour la class**

Constante `__NAMESPACE__`

```
<?php
namespace Scholarite;

function get($nomClasse)
{
    $a = __NAMESPACE__ . '\\'. $ nomClasse;
    return new $a;
}

echo "<h1>Espace de nom courant : " . __NAMESPACE__ . "</h1>";

$obj = get ("Etudiant");
```


Commande *namespace*

- **Référence à l'espace de nom courant**
- **Utilisation**
 - *namespace\constante*
 - *namespace\fonction*
 - *namespace\class*
- **Utilité**

accès à une ressource (constante, fonction, classe) masquée par un use

Commande *namespace*

```
<?php
namespace Sclarite;

class Etudiant
{ ... }

$obj = new namespace\Etudiant();
```

Alias

- **Raccourcis**
opérateur use
- **Importation :** **use** *ressource*;
 ressource : chemin absolu
- **Alias :** **use** *ressource as raccourci*;
 raccourci : chaîne de caractères (alias)
 - *alias de nom de classe*
 - *alias de nom d'interface*
 - *alias d'espace de nom*
 - *alias de fonction (PHP 5.6)*
 - *alias de constante (PHP 5.6)*

Alias

```
<?php
namespace Projet;
use Scholarite\Metier\Etudiant as Etu;

use Scholarite\Metier;                                // Equivalent à: use Scholarite\Metier as Metier

use ArrayObject;                                       // importation d'une classe globale

use function Scholarite\Metier\evaluer;                // importation d'une fonction (PHP 5.6+)

use function Scholarite\Metier\evaluer as eval;         // alias d'une fonction (PHP 5.6+)

use const Scholarite\Metier\ETUDIANT_RECUC;            // importation d'une constante (PHP 5.6+)

$obj = new namespace\Etu;                                // instantie un objet de la classe Projet\Etu
$obj = new Etu;                                           // instantie un objet de la classe Scholarite\Metier\Etudiant
Metier\evaluer();                                         // appelle la fonction Scholarite\Metier\evaluer
$a = new ArrayObject(array(1));                          // instantie un objet de la classe ArrayObject
// Sans l'instruction "use ArrayObject", référence à la classe Projet\ArrayObject

eval();                                                   // Appel la fonction Scholarite\Metier\evaluer
echo ETUDIANT_RECUC;                                     // affiche la valeur de Scholarite\Metier\ETUDIANT_RECUC
?>
```

Règles de résolution

Rappels

- **Nom non qualifié**
ne contient aucun anti-slash
ex: calcul
- **Nom qualifié**
contient au moins un anti-slash
ex : Metier\calcul
- **Nom absolu**
commence par un anti-slash
ex : \Scolarite\Metier\calcul

Règles de résolution

Dans l'ordre :

1. **Noms absolus** : Résolution des appels (fonction, class, constante) à la compilation
`$e=new \Scolarite\Metier\Etudiant();` → classe Scolarite\Metier\Etudiant
2. **Noms non absolus** : Utilisation des règles d'alias
`use Scolarite\Metier;`
`$e=new Etudiant ();` → classe Scolarite\Metier\Etudiant
3. **Noms qualifiés non résolus** : Préfixage par l'espace de nom courant
`namespace Scolarite\Metier;`
`$e=new Etudiant ();` → classe Scolarite\Metier\Etudiant
4. **Classes non qualifiées** : Application des règles d'alias à la compilation
`use Scolarite\Metier\Etudiant as Etudiant;`
`$e=new Etudiant ();` → new Scolarite\Metier\Etudiant();
5. **Appels des fct non qualifiées** : Recherchés dans l'espace de nom courant, puis global à la compilation
`namespace Scolarite\Metier;`
`calcul ();` → recherche Scolarite\Metier\calcul(), puis fct globale calcul()
6. **Appels des classes qualifiées ou non (i.e. non absolues)** : recherchés dans l'espace de nom courant, puis global à l'exécution
`namespace Scolarite\Metier;`
`$e=new Etudiant ();` → recherche classe Scolarite\Metier\Etudiant , puis autoload pour Scolarite\Metier\Etudiant

