

Interrogation de registry via PHP

- Code client fourni par le nvo
- Tutorial PHP de leur école d'été
- Accès aux registries du NVO via SOAP
- Utilisation de la librairie NuSOAP de PHP

- La classe VORegistry :
- Simple à utiliser :
- Il suffit de configurer le chemin d'accès au fichier wsdl du serveur
- Valeurs par défaut :
- ```
var $VORegistryWSDLList = array
("http://nvo.stsci.edu/VORegistry/registry.asmx?WSDL", "http://voservices.net/Registry/registry.asmx?WSDL"); // mirror site
```

- On peut spécifier d'autres registries dans le constructeur
- Méthodes :
- fonction QueryVOResource (\$predicate) :  
requête basée sur un prédicat

ResourceType like 'CONE' and contentlevel  
like '%research%'

- fonction QueryResource(\$predicate) :  
similaire à QueryVOResource mais retourne  
un format spécifique

- function KeywordSearch (\$swords, \$andKeys) : recherche par mots clés
- function DumpVOResources () : retourne toutes les VORresources d'un registry
- function getRevisions () : teste que les requêtes SOAP sont disponibles sur le serveur

# Format des réponses

- Les méthodes KeywordSearch et QueryVOResource retournent des VOResource
- Format défini par l'IVOA pour décrire les ressources contenues dans les registries
  - <http://ivoa.net/Documents/WD/ReR/VOResource-20061107.html>
- Les résultats sont dans un tableau associatif qui suit le schema XML de VOResource

- Tableau de 2 éléments :
  - Resources
  - Attributes
- Resources contient la liste des Ressources résultats
- Pour chaque résultat on a un tableau associatif :
  - Ex :
    - Title => String
    - shortName => String
    - Identifier => String
    - Curation => array
    - ...

# Inconvénients

- N'utilise pas des méthodes standards définies par l'IVOA
- Utilise une interface propriétaire définie par STSCI
- Peu pérenne
- Une interface commune aux registries a été définie
  - <http://ivoa.net/Documents/latest/RegistryInterface.html>
- Elle est à présent en version 1.00
- Elle définit des méthodes Search et KeywordSearch