

# **Client TAP**

## **Retour d'Expérience**

C.Chauvin, M.Servillat,  
P. Le Sidaner  
Observatoire de Paris

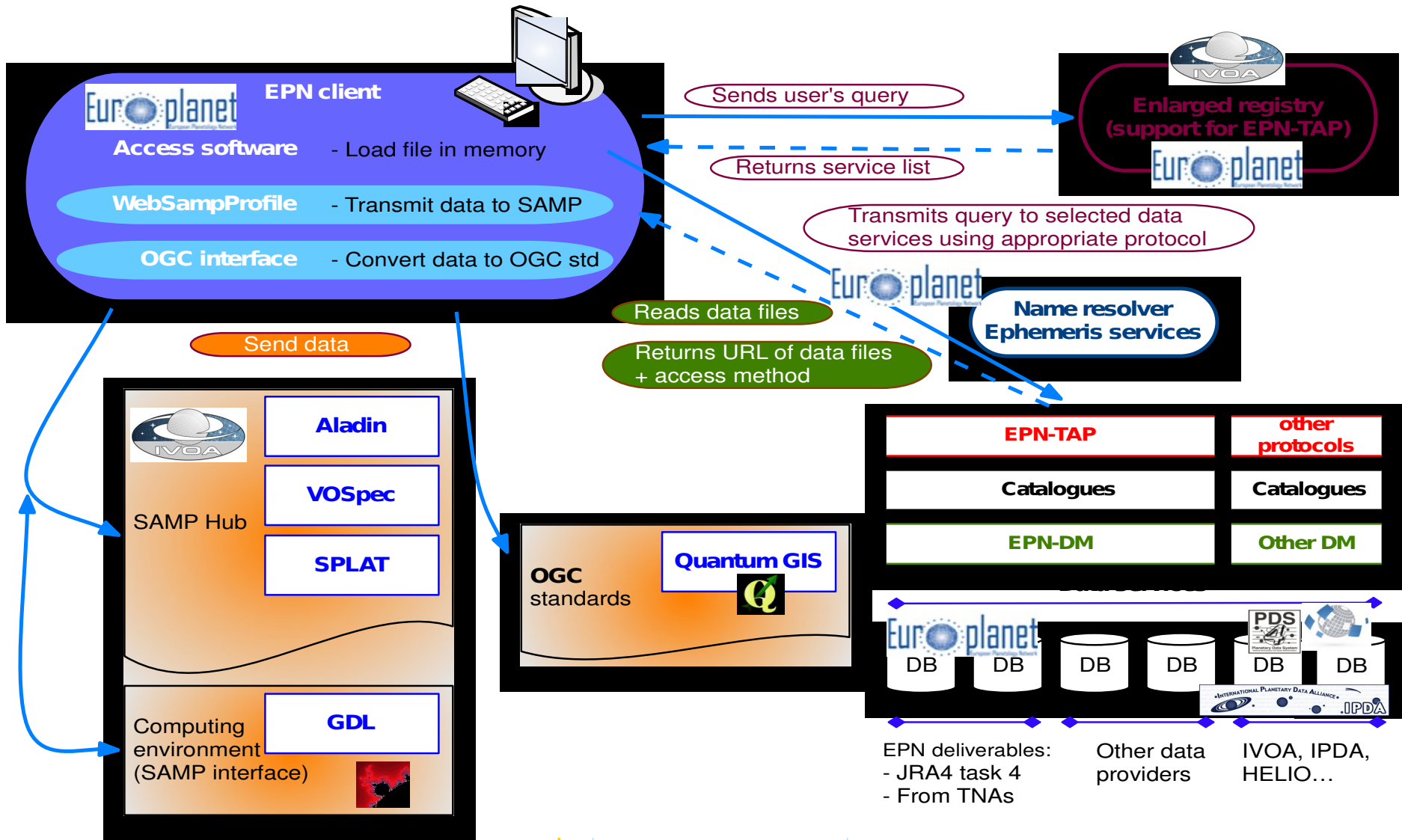
## Data Scope :

- **Databases produced by various work packages during the Europlanet program**
- **Space borne data from planetary missions (PSA)**
- **Databases and tools related to participants of the EPN/IDIS (GhoSST, AMDA, SSODnet)**
- **Big data repositories including Planetary Science data (the ESO and HST archives)**
- **Data sets directly published in a compliant form by data providers**
- **External data provider including observational data derived from space missions or ground-based telescopes, but also reference data acquired in the laboratory and simulations**

## Aim of EPN-TAP

- **Facilitate searches in big archives**
- **Solution to specific complexity of Planetary Science (measurement types and coordinate frames)**
- **On-line data access and visualization possible**
- **Allow small data providers to share their data in an interoperable environment with minimum effort**

# EPN-TAP Architecture



## Réutilisation du code Vespa dans le CTA Web Client

- **Serveurs en python 2**
- **Utilisation du framework Django pour gérer simplement :**
  - Logs**
  - Sécurité**
  - Sessions**
  - Templates**
  - Authentification**
  - URL**
  - Formulaires**
- **Astropy pour lire les VOTable**
- **jQuery**

- **Éviter à l'utilisateur de taper une requête ADQL...**
- **Indiquer à l'utilisateur les champs de la base qu'il peut interroger (champs communs pour Vespa)**
- **Ajouter des fonctionnalités dynamiques:**
  - Plusieurs types de coordonnées (céleste, cylindrique, sphérique, etc.)**
  - Différentes manières de saisir des dates dans un intervalle (inclusion, intersection, delta)**
  - NameResolver (Simbad, SsODNet)**
- **Dans Vespa, générer un formulaire pour chaque service**

- **SsODNet : API REST de l'IMCCE**
- **(extra)solar system object names**
- **Autocomplete**
- **<http://voparis-registry.obspm.fr/ssodnet/1/autocomplete?q=venu>**
- **`{"count":2,"hits":  
[{"name":"Venus","type":"Planet","id":"Venus","parent":"Sun","aliases":["2"]},  
{"name":"Venusia","type":"Asteroid","id":"Venusia","parent":"Sun","aliases":["499","1902KX ","1950 CE"]}],  
"total":2}`**

- **Découvre les services qui nous intéressent**
- **[http://voparis-registry.obspm.fr\(..\)/voresources/search?  
keywords=datamodel:epncore&max=100](http://voparis-registry.obspm.fr(..)/voresources/search?keywords=datamodel:epncore&max=100)**
- **Fournit des informations sur chaque service**

**schema**

**table**

**access\_url**

**title**

**description**

**creator, contributors**

**reference\_url**



- Différents serveurs TAP, actuellement Dachs et VODance
  - Une requête par service : `SELECT * FROM {service_id} WHERE ...`
  - Idéalement, clause `WHERE` commune à tous les services mais...
  - Si un service n'a pas renseigné un paramètre, on ne veut pas forcément l'exclure des résultats
  - Solution envisagée : Pouvoir pour chaque service, définir des colonnes qui puissent être à `NULL` sans les exclure du résultat
  - Exemple de requête : `SELECT * FROM apis.epn_core WHERE target_name = "Jupiter"`  
Si le provider a dans sa base `jupiter` → exclut des résultats
- Comparaison case insensitive ajoutée par l'IVOA, implémentée dans Dachs mais pas dans VODance...

- **L'utilisateur ne veut pas voir de VOTable**

→ **Affichage d'un tableau avec un minimum de fonctionnalités (tri, show/hide colonnes) library dataTables.js pour Vespa, bootstrap pour CTA web client**

- **Sur Vespa, affichage d'une preview**

- **On ne récupère pas directement la donnée mais une URL, l'utilisateur peut télécharger la donnée ou l'envoyer par SAMP**

- **Utilisation de la library samp.js avec différents mtype :**

- image fits**

- spectres**

- tables**

- cdf**

- cube Virtis**

- **H2020 démarre à la fin du mois → nouveaux développements à venir sur Vespa**
- **Traiter de grosses quantités de données :**
  - Utiliser `votable.js` pour pouvoir parser des `VOTable` dynamiquement
  - Combiner de la pagination avec des requêtes `ADQL`
- **Idéalement, afficher les résultats de tous les services dans un seul tableau mais...**  
**Comment gérer les grosses quantités de données sachant qu'elles sont dans des bases différentes ?**
- **Intégrer les données du PSA en résolvant leur problème de prise en compte de colonnes non renseignées**
- **Permettre à l'utilisateur de récupérer une requête `ADQL` générée pour pouvoir l'envoyer simplement dans un champ texte.**