



***Heliophysics
Integrated
Observatory***

Project No.: 238969
Call: FP7-INFRA-2008-2

**HFC
Guide Technique**

<i>Title:</i>	HFC Guide Technique
<i>Document No.:</i>	
<i>Date:</i>	5 Juin 2013
<i>Editor:</i>	X. Bonnin , Paris Observatory (OBSPARIS)
<i>Contributors:</i>	C. Renié, J. Abouardham, N. Fuller (OBSPARIS)
<i>Distribution:</i>	Project

Version	Date	Released by	Detail
0.1	2011-03-14	X.Bonnin	Draft
1.0	2011-06-28	X.Bonnin	Ajout de l'utilisation du serveur d'application Apache Tomcat
1.1	2011-04-25	X.Bonnin	Précision sur la configuration de log4j
1.2	2012-06-13	X.Bonnin	Description du compte helio sur siolino. Détails des scripts
2.0	2013-06-05	X.Bonnin	Maj de l'arborescence sur voparis-helio, et ajout des informations sur le site miroir du HFC
2.1	2014-04-15	X.Bonnin	Maj des informations relatives aux machines de production voparis-helio et tycho.

Table des matières

Liste des acronymes.....	4
Références utiles.....	5
Introduction.....	6
I. Organisation du service.....	6
1) Architecture générale.....	6
2) Organisation des serveurs.....	7
II. Les bases de données.....	7
1) Description.....	7
2) Installation.....	8
3) Mise à jour de la base hfc1.....	8
4) Connexion aux bases via un client MySQL.....	8
III. L'interface de requête HELIO.....	9
1) Description.....	9
2) Installation et maintenance.....	9
Apache Tomcat.....	9
Déploiement.....	10
Configuration de Tomcat.....	10
Configuration du HQI.....	12
IV. L'interface de requête EPN-TAP.....	14
V. STILTS.....	14
VI. L'interface web.....	14
VII. Autres interfaces.....	15
1) SolarSoft.....	15
2) API IDL pour le HFC.....	15
3) API Java.....	15
4) API Python.....	15
5) HELIO Front End.....	16
VIII. Codes de détection.....	16
1) Au LESIA.....	16
tycho.obspm.fr.....	16
2) Au Trinity College Dublin.....	17
3) FTP.....	18
Annexes.....	19
1) Schéma de la base de données.....	19
2) Outils de gestion.....	19
Dépôts SubVersionN.....	19
Projet Redmine.....	20
Contacts.....	20

Liste des acronymes

API : Application Programming Interface
CSV : Comma-Separated-Value
DIO : Division Information de l'Observatoire
DPAS : Data Provider Access Service
FRC : Feature Recognition Code
HEC : Heliophysics Events Catalogue
HELIO : Heliophysics Integrated Observatory
HFC : Heliophysics Features Catalogue
HQI : Helio Query Interface
IAS : Institut d'Astrophysique Spatiale
ICS : Instrument Coverage Service
ILS : Instrument Location Service
IVOA : International Virtual Observatory Alliance
GUI : Graphical User Interface
LESIA : Laboratoire d'Etudes Spatiales et d'Instrumentation en Astrophysique
MEDOC : Multi Experiment Data & Operation Center
MSSL : Mullard Space Science Laboratory
PID : Process ID
STILTS : Starlink Tables Infrastructure Library Tool Set
SVN : SubVersion
VO : Virtual Observatory
VO-PDC : VoParis-Data Centre
URL : Uniform Resource Locator
WSDL : Web Services Description Language

Références utiles

- RD1 HELIO_HFC_V2.0 : document décrivant l'architecture de la base de données du HFC
- RD2 HELIO_HFC_GUI_V1.0 : document décrivant le GUI du HFC
- RD3 HELIO_OBSPM_S2_005_UM_HFC_User_Guide_v1.1 : guide d'utilisation des interfaces du HFC.
- RD4 HELIO_OBSPARIS_S2_005_TN_HFC_Developers_Guide_v0.1 : guide d'installation du HFC à l'intention des développeurs.
- RD5 HELIO_HFC_webservices_V10.pdf : document décrivant les accès web services du HFC
- RD6 Helio_FullQuery_0.2 : document décrivant l'interface de requête HELIO (HQI)
- RD7 <http://www.helio-vo.eu> : Site web du projet HELIO
- RD8 <http://voparis-helio.obspm.fr/hfc-gui/> : Site web du HFC à l'Observatoire de Paris
- RD9 <http://helio-hfc.ias.u-psud.fr/> : Site web du HFC à l'IAS
- RD10 <http://voparis-helio.obspm.fr/hfc-hqi/HelioService> : Site web listant les fichiers wsdl de l'interface de requête du HELIO du HFC
- RD11 <http://www.star.bris.ac.uk/~mbt/stilts/> : Site de l'outil STILTS
- RD12 <http://dio.obspm.fr/Calcul/tycho/> : Site web décrivant la grappe de calcul Tycho
- RD13 <http://vo.obspm.fr/about/vopdc> : Site web du VO-Paris Data Centre
- RD14 <http://helio-vo.svn.sourceforge.net/> : adresse du dépôt SVN du projet HELIO
- RD15 <https://version-lesia.obspm.fr/repos/helio/> : adresse du dépôt SVN du projet HELIO au LESIA
- RD16 <http://www.ivoa.net/Documents/> : Documentation sur les standards et interfaces de l'IVOA.

Introduction

Ce document décrit en détail l'organisation des différentes composantes du Heliophysics Feature Catalogue (HFC) hébergé à l'Observatoire de Paris.

Le HFC est un service de l'Observatoire Virtuel HELIO [RD6] chargé de fournir des informations sur des structures solaires et héliosphériques, observées par divers instruments depuis le sol ou l'espace. Les données relatives aux structures sont produites en amont par des codes de détection et de suivi automatiques¹.

On trouvera en annexe de ce document une liste des personnes impliquées dans le projet avec leur adresse email, un schéma de la base de données du HFC, ainsi qu'une liste des outils de gestion utilisés pour la maintenance et les développements du service.

I. Organisation du service

1) Architecture générale

La Figure 1 présente l'organisation simplifiée de la chaîne de traitement constituant le HFC, on y retrouve :

- une **base de données** (database) où sont stockées les informations relatives aux structures.
- des **codes de reconnaissance automatique** (« Feature Recognition Code », FRC, en anglais) utilisés en amont afin de détecter les structures sur les observations, puis d'en extraire des paramètres scientifiques pertinents (tels que la date et l'heure d'observation, la position, les contours, ou bien encore l'intensité moyenne).
- Des **interfaces utilisateurs** permettant d'accéder aux données sauvegardées dans la base. Quatre types d'interface sont actuellement disponibles : une page web ou GUI [RD8], un service web dédié appelé HQI [RD6], et une interface EPN-TAP. A cela s'ajoute également une interface STILTS [RD11] réservée exclusivement aux tests en interne.

¹ La liste complète des données disponibles, ainsi que les intervalles de temps couverts, peuvent être consultés sur le site web du HFC [RD8].

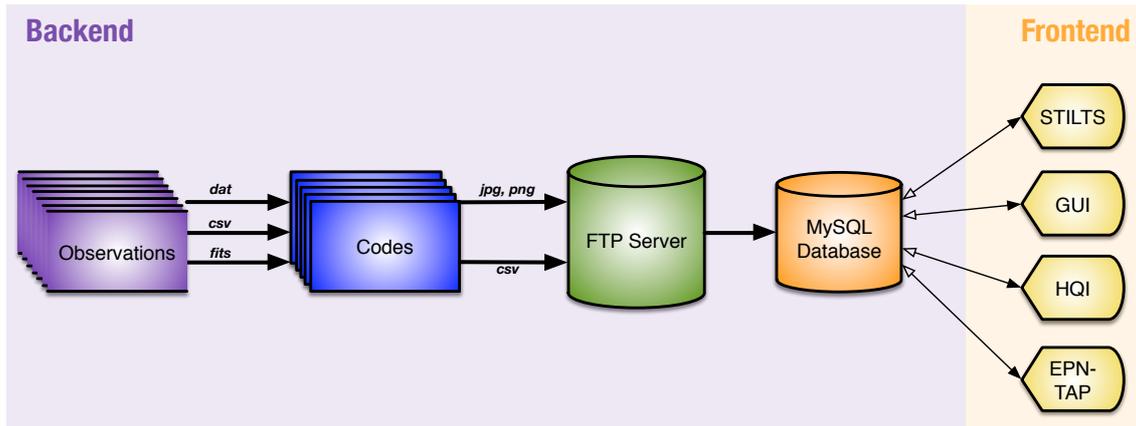


Figure 1 – Architecture générale du HFC

2) Organisation des serveurs

Le HFC s'appuie actuellement sur 5 serveurs de type Linux pour fonctionner :

- ***voparis-helio.obspm.fr*** : cette machine est hébergée et administrée par le VO-PDC [RD13]. Elle accueille les différentes interfaces du HFC (interface EPN-TAP exceptée), ainsi que les routines d'insertion des données des codes de détection dans la base de données. Elle gère aussi la production des données du code TRACKRS. C'est également le seul point d'accès à la base de données publique du HFC via un client mysql (voir chapitre suivant).
- ***voparis-mysql5-paris.obspm.fr*** : c'est la machine utilisée pour héberger les bases de données du HFC. Elle est également administrée par le VO-PDC, et n'est accessible en écriture via un client mysql que depuis la machine *voparis-helio*.
- ***tycho.obspm.fr*** : cette machine gérée par la DIO est le point d'accès à la grappe de calcul de l'Observatoire de Paris. Un compte *helio* a été spécialement créé pour y installer et exécuter une grande partie des codes de détection sur la grappe.
- ***ftpbase2000.obspm.fr*** : c'est le serveur FTP utilisé par le HFC pour stocker les fichiers produits par les codes de détection. En pratique les routines d'insertion viennent directement lire les fichiers sur ce serveur pour remplir la base de données.

Chacun des éléments constituant le HFC sont décrits en détails dans les chapitres suivants.

II. Les bases de données

1) Description

Le HFC donne accès à une base de données où sont répertoriées des informations relatives à des structures solaires et héliosphériques. Il existe actuellement quatre versions MySQL de la base de données du HFC, réparties sur deux machines de l'Observatoire de Paris :

- Une version test *hfc1test* installée sur la machine *voparis-mysql5-paris.obspm.fr*. Le contenu de cette version test est accessible depuis internet via les interfaces de développement <http://voparis-helio.obspm.fr/hfc-gui-dev/> et <http://voparis-helio.obspm.fr/helio-hfc-dev/HelioService>.
- Une version stable *hfc1* également installée sur la machine *voparis-mysql5-paris.obspm.fr*. Cette version de production est le point d'accès des utilisateurs au contenu du catalogue via les différentes interfaces publiques. **Cette base de données doit donc toujours rester opérationnelle, et en aucun cas servir à des tests ou développements.**

☛ Sauf mention explicite, l'ensemble des procédures relatives aux interfaces et bases de données décrites par la suite concerne respectivement les machines *voparis-helio* et *voparis-mysql5-paris* uniquement.

2) Installation

Pour créer les tables et vues de la dernière version de la base de données *hfc1*, on pourra utiliser le script :

```
/home/helio/hfc/prod/database/software/hfc_tables/sql/create_hfc1.sql
```

3) Mise à jour de la base hfc1

La base *hfc1* est mise à jour de manière journalière en exécutant le script */home/helio/hfc/prod/tools/cron/hfc_insert_frc.voparis.sh* à l'aide de **cron**. Ce script se charge de lancer l'insertion des dernières données de détection disponibles sur le serveur *ftpbass2000*, de mettre à jour la table DATE_STAT dans la foulée.

4) Connexion aux bases depuis un client MySQL

Pour des raisons de sécurité, excepté via les interfaces utilisateur dédiées, il n'est pas possible d'interroger les bases de données depuis Internet en utilisant un client MySQL, seul une connexion depuis l'Intranet de l'Observatoire de Paris-Meudon est autorisée. L'accès en lecture seule aux bases *hfc1tmp* et *hfc1* de la machine *voparis-mysql5-paris* peut se faire depuis la machine *voparis-helio* en utilisant les paramètres de connexion suivants :

```
Host = voparis-mysql5-paris.obspm.fr  
User = guest  
Password = guest
```

Exemple : pour se connecter à la base *hfc1test*, depuis le prompt entrer :

```
➤ mysql --host=helio-fc1.obspm.fr --user=guest --password=guest hfc1test
```

☛ Pour un obtenir les paramètres de connexion en lecture et en écriture, voir avec Christian Renié ou Xavier Bonnin.

III.L'interface de requête HELIO

1) Description

L'interface de requête HELIO (ou HQI) est une application Java développée par le MSSL dans le cadre du projet HELIO. Elle offre un accès standardisé aux différents services HELIO hébergeant des bases de données de type SQL. Comme toute servlet Java, elle peut être déployée dans n'importe quel conteneur de servlets ou serveur d'applications Java, tels que Apache Tomcat ou bien encore GlassFish. (Les développeurs HELIO recommandent d'ailleurs l'utilisation de l'un ou l'autre de ces deux logiciels respectivement dans leur version 6.0.xx et 3.x).

Dans le cas du HFC, une version 6.0 du conteneur Apache Tomcat est installée sur *voparis-helio.obspm.fr* dans le répertoire */usr/local/apache-tomcat-6.0*. Tomcat héberge le HQI, dont les fichiers sources sont sauvegardés dans le dossier */home/helio/hqi/webapp*. Les fichiers archivés (.jar) et sources de l'interface sont également téléchargeables depuis le dépôt SVN du projet [RD14].

2) Installation et maintenance

Apache Tomcat

La maintenance de Tomcat sur *voparis-helio.obspm.fr* est assurée par la DIO. Néanmoins, il est, sous certaines conditions, possible de faire une demande afin d'obtenir les droits d'accès à certaines fonctionnalités telles que la gestion du serveur Tomcat (e.g., démarrage/arrêt, déploiement de servlets, etc.).

Le lancement/le redémarrage/l'arrêt de Tomcat se fait via la commande :

```
/usr/local/etc/rc.d/tomcat6 start/restart/stop
```

On peut vérifier rapidement que Tomcat fonctionne correctement, en ouvrant depuis l'Intranet de l'Observatoire à la page d'accueil de son interface graphique:

<http://voparis-helio.obspm.fr:8082/>

Si tout va bien, la page visible sur la figure 2 devrait s'afficher dans le navigateur web.

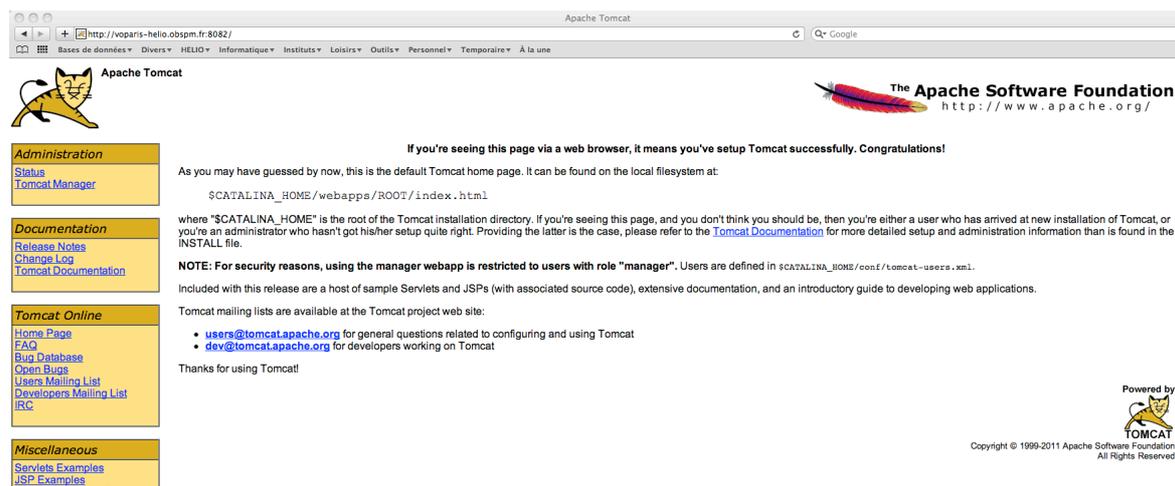


Figure 2: Page d'accueil de Tomcat

La gestion des servlets déployés sur Tomcat peut se faire facilement depuis l'interface graphique, en se rendant sur la page administrateur via le lien **Tomcat Manager** du menu de gauche. (L'accès à cette page est sécurisé, voir avec Christian Renié ou Xavier Bonnin pour plus d'information). Depuis cette page, il est possible de déployer/démarrer/arrêter/supprimer des servlets.

➡ Pour plus d'informations sur le fonctionnement d'Apache Tomcat, consulter la documentation en ligne accessible depuis le site <http://tomcat.apache.org/>.

Déploiement

Une archive war de la dernière version du HQI utilisée par Tomcat est disponible dans le répertoire `/home/helio/hqi/archives/war/releases/current`.

Le fichier texte `/home/helio/hqi/release.txt` fournit également des informations concernant le déploiement des différentes versions de l'interface².

Le déploiement de l'archive war peut se faire via le menu du Tomcat Manager, qui installera par défaut la servlet dans le dossier `/usr/local/apache-tomcat-6.0/webapps`. Néanmoins, afin d'éviter d'intervenir le plus possible directement dans le dossier de Tomcat, l'archive est déployée dans le dossier `/home/helio/hqi/webapp`; la machine `voparis-helio.obspm.fr` accueillant également depuis mars 2013 une copie des services ICS, ILS, et DPAS d'HELIO accessibles via le HQI, et chacun de ces services (HFC y compris et DPAS excepté) utilisant la même version de l'interface.

Configuration de Tomcat

Pour indiquer à Tomcat où trouver l'interface, il faut produire un fichier de configuration au format xml pour chaque service (DPAS excepté), qui devra être sauvegardé dans le sous-dossier `/usr/local/apache-tomcat-6.0/conf/Catalina/localhost`.

² Pour obtenir la version la plus récente du HQI, on pourra se rendre sur le dépôt SVN du projet [RD14], ou bien contacter Kevin Benson du MSSL.

HFC : guide technique Version 2.1

Dans le cas du HFC, le fichier en question s'appelle *hfc-hqi.xml*, et contient les lignes suivantes :

```
<?xml version="1.0"?>
```

```
<Context path="/hfc-hqi" docBase="/home/helio/hqi/webapp" debug="0"  
privileged="true">
```

```
<Environment description="Property Context file for Helio HQI service. Describes db  
connections and how to query tables"
```

```
name="property/context" override="false" type="java.lang.String"  
value="/home/helio/hqi/hfc/hfc-hqi-property.txt" />
```

```
</Context>
```

Les options indiquées en bleues sont les plus importantes :

- `path="/hfc-hqi"` renseigne sur le chemin relatif de l'URL utilisée par l'application.
- `docBase="/home/helio/hqi/webapp"` renseigne le chemin d'accès au dossier contenant la servlet concernée (ici le HQI).
- `value="/home/helio/hqi/hfc/hfc-hqi-property.txt"` renseigne sur le chemin d'accès au fichier de configuration utilisé par le HQI pour accéder à la bdd SQL (voir chapitre suivant)

Si le HQI est correctement déployée et configuré, sa page d'accueil (cf. figure 3) doit être visible à l'adresse³ :

<http://voparis-helio.obspm.fr:80/hqi-hfc>

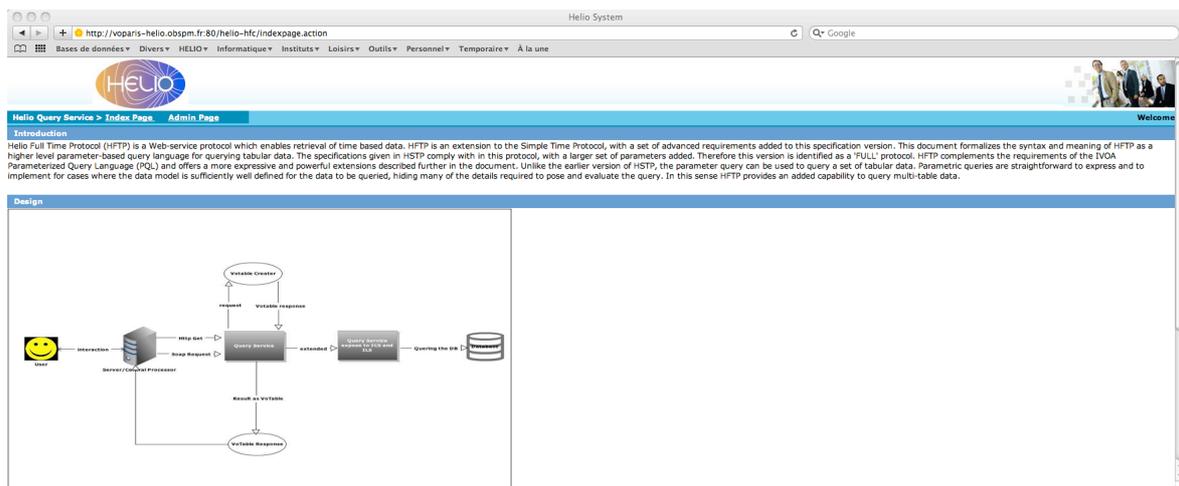


Figure 3: Page d'accueil du HQI.

³ Ici cas du HFC (pour les autres services, l'URL sera différente).

 Pour des raisons historiques le HFC est également accessible depuis la page <http://voparis-helio.obsppm.fr:80/helio-hfc>. Il faut donc s'assurer que des fichiers *helio-hfc.xml* et *helio-hfc-property.txt* existent aussi pour cet accès.

Notes :

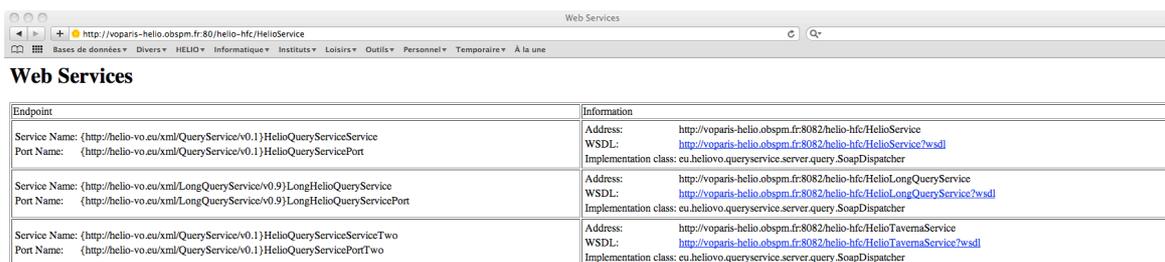
- Les numéros de port sont définis dans Tomcat, et peuvent être modifiés en éditant les fichiers de configuration dédiés. Il est néanmoins prévu d'accéder au HQI via le port 80 (le port 8080 étant réservé à STILTS ; cf. chapitre sur STILTS).
- Il est conseillé de redémarrer Tomcat après chaque installation ou mise à jour du HQI.

Il est également utile de vérifier que les services web proposés par le HQI fonctionnent correctement en se rendant à l'adresse :

<http://voparis-helio.obsppm.fr:80/hfc-hqi/HelioService>

, qui doit normalement ressembler à la page illustrée sur la figure 4. Cette page fournit la liste des fichiers WSDL décrivant les méthodes de requêtes possibles (<http://www.w3.org/TR/wsdl>).

 Le HQI utilise une classe java log4j qui écrit le fichier log *queryservice.log* reportant l'activité de l'interface (e.g., entrées/sorties, messages d'erreur, etc.). Avant de tester des requêtes, mieux vaut d'abord s'assurer que le chemin vers le dossier d'écriture de ce fichier log est correct, sous peine de plantage de Tomcat. Pour cela il faut éditer la ligne « **log4j.appender.LOGFILE.File=** » du fichier **/home/helio/hqi/webapp/WEB-INF/classes/log4j.properties** (qui doit normalement pointer vers le dossier */usr/local/apache-tomcat-6.0/logs/hfc-hqi.log*). Dans tous les cas si le moindre problème survient, on peut toujours se reporter aux fichiers logs de tomcat (disponible dans le dossier */usr/local/apache-tomcat-6.0/logs*).



Endpoint	Information
Service Name: {http://helio-vo.eu/xml/QueryService/v0.1}HelioQueryServiceService Port Name: {http://helio-vo.eu/xml/QueryService/v0.1}HelioQueryServicePort	Address: http://voparis-helio.obsppm.fr:8082/helio-hfc/HelioService WSDL: http://voparis-helio.obsppm.fr:8082/helio-hfc/HelioService?wsdl Implementation class: eu.heliovo.queryservice.server.query.SoapDispatcher
Service Name: {http://helio-vo.eu/xml/LongQueryService/v0.9}LongHelioQueryService Port Name: {http://helio-vo.eu/xml/LongQueryService/v0.9}LongHelioQueryServicePort	Address: http://voparis-helio.obsppm.fr:8082/helio-hfc/HelioLongQueryService WSDL: http://voparis-helio.obsppm.fr:8082/helio-hfc/HelioLongQueryService?wsdl Implementation class: eu.heliovo.queryservice.server.query.SoapDispatcher
Service Name: {http://helio-vo.eu/xml/QueryService/v0.1}HelioQueryServiceServiceTwo Port Name: {http://helio-vo.eu/xml/QueryService/v0.1}HelioQueryServicePortTwo	Address: http://voparis-helio.obsppm.fr:8082/helio-hfc/HelioTavernaService WSDL: http://voparis-helio.obsppm.fr:8082/helio-hfc/HelioTavernaService?wsdl Implementation class: eu.heliovo.queryservice.server.query.SoapDispatcher

Figure 4: Page des services web du HQI.

Configuration du HQI

Afin de s'assurer que le HQI se connecte à la bonne base de données, et renvoie les champs demandés lors des requêtes, il est nécessaire d'éditer un fichier de configuration (dit « property file » en anglais). Ce fichier ASCII contient en en-tête les informations pour se connecter à la base (typiquement le pilote jdbc, le nom d'utilisateur, et le mot de

pas correspondant), puis les noms des tables et des colonnes que l'on souhaite rendre accessible depuis l'interface (avec optionnellement la description de ces champs, ainsi que les UCD et Utype qui leur sont associés).

Le HQI propose une interface graphique pour générer ce fichier (se rendre à la page d'accueil puis cliquer sur le lien « Admin page » du menu, cf. figure 3). On peut également l'écrire directement pour peu que l'on respecte la syntaxe.

☛ Un script python **make_hqi_property.py** permet de générer rapidement ce fichier. Il est disponible dans le dossier */home/helio/hqi/tools/property_file*. Pour fonctionner correctement ce script requiert la lecture de deux fichiers **hfc-hqi-jdbc.txt** et **hfc-hqi-constraints.txt** qui contiennent respectivement les informations de connexion à la base de données (nom de la base, login, password, pilote jdbc, etc.), et les informations sur les éventuelles contraintes relatives à certains champs des tables, contraintes qui peuvent être utilisées lors des requêtes (e.g., contraintes temporelles, contraintes sur le nombre maximum de lignes renvoyées par l'interface). Si le premier fichier doit être obligatoirement fourni, le second est optionnel. On trouvera une copie de ces fichiers dans le dossier */home/helio/hqi/hfc/config*.

Une fois le fichier créé, il est recommandé de relancer ensuite l'application à partir de la page d'administration de Tomcat (à l'aide du bouton « reload »).

Il est possible de vérifier le bon fonctionnement de l'interface en effectuant une requête depuis un logiciel client de type SOAP, ou bien tout simplement à partir de n'importe quel navigateur web, en appelant une des méthodes que propose le HQI, HelioQueryService, via une requête de type HTTP-GET telle que :

http://voparis-helio.obspm.fr:80/hfc-hqi/HelioQueryService?FROM=FRC_INFO

Dans l'exemple ci-dessus, on récupère les informations stockées dans la table FRC_INFO du HFC, en appelant la servlet HelioQueryService. Si la requête aboutie, le navigateur devrait retourner un fichier au format XML, une votable, tel que défini par l'IVOA [RD16] contenant les champs et les valeurs de la table FRC_INFO.

Il est également possible d'interroger des vues (si elles ont été indiquées dans le fichier de configuration), et d'ajouter des contraintes sur certains paramètres tels que l'intervalle d'observation par exemple :

http://voparis-helio.obspm.fr/hfc-hqi/HelioQueryService?FROM=VIEW_AR_HQI&STARTTIME=2001-01-24T00:00:00&ENDTIME=2001-01-26T22:45:30

On effectue ici une requête sur la vue VIEW_AR_HQI - qui regroupe les informations sur les régions actives - afin d'obtenir les données du catalogue entre le 24 janvier 2001 à 00:00:00, et le 26 janvier 2001 à 22:45:30, en ajoutant pour cela les mots-clés STARTTIME et ENDTIME à l'URL de la requête.

☛ Le HQI autorise l'utilisation des langages de requête PQL et ADQL au travers des interfaces HTTP et TAP. Consulter [RD5], [RD6], et [RD16] pour plus de détails.

IV. L'interface de requête EPN-TAP

To be done.

V. STILTS

Le logiciel STILTS est également installée sur la machine *voparis-helio* dans le répertoire */usr/local/stilts/* (une copie est également disponible dans le répertoire */home/helio/tools/stilts/*). Une fois démarrée en mode serveur, cet outil permet d'effectuer des requêtes à l'aide d'une syntaxe SQL sur les bdd. **En pratique STILTS ne constitue pas une interface d'accès « officiel » au HFC, son utilisation doit donc se limiter aux personnes travaillant dans le cadre du projet.**

Pour démarrer le mode serveur, exécuter la commande suivante depuis la machine *voparis-helio.obspm.fr* :

```
java -classpath stilts.jar:/usr/local/stilts/mysql-connector-java-5.0.6-bin.jar -  
Djdbc.drivers=com.mysql.jdbc.Driver uk.ac.starlink.ttools.Stilts server  
tasks="sqlclient tcopy" port=8080 &
```

L'arrêt du serveur STILTS se fait brutalement en utilisant la commande kill sous unix (on pourra utiliser la commande unix « ps aux » pour identifier le pid du serveur).

Pour vérifier que STILTS fonctionne correctement, il est possible de tester une requête MySQL depuis n'importe quel navigateur web, en accédant au serveur *voparis-helio.obspm.fr* via le port 8080, par exemple :

<http://voparis-helio.obspm.fr:8080/stilts/task/sqlclient?db=jdbc:mysql://voparis-mysql5-paris.obspm.fr:3306/hfc1&user=guest&password=guest&sql=SHOW%20TABLES>

On demande ici à STILTS de nous afficher la liste des tables et vues disponibles dans la base *hfc1* hébergée sur *voparis-mysql5-paris.obspm.fr*, à l'aide de la requête SQL « SHOW TABLES ».

☛ On peut remarquer que les champs renseignés dans le lien sont typiquement : l'adresse et le numéro de port du serveur hébergeant STILTS, *voparis-helio.obspm.fr* :8080, l'adresse jdbc de la bdd, *jdbc:mysql://voparis-mysql5-paris.obspm.fr* :3306/hfc1, le nom et le mot de passe de l'utilisateur, *guest/guest*, et la commande sql, *SHOW TABLES*.

Pour plus d'information sur STILTS, visiter [RD11].

VI. L'interface web

L'interface web du HFC (ou GUI) est accessible à l'adresse :

<http://voparis-helio.obspm.fr/hfc-gui/>

L'utilisation du GUI est décrite en détail dans la référence [RD3].

Les fichiers sources sont sauvegardés dans le dossier `/usr/local/www/apache22/data/hfc-gui` du serveur HTTP Apache qui gère la page web. L'ensemble des fichiers sources peut être également récupéré depuis les dépôts SVN du projet et du LESIA.

Une version de développement du GUI est également hébergée sur la machine *voparis-helio*, et interroge la base *hfc1test* sur *voparis-mysql5-paris*. Elle est accessible depuis l'adresse :

<http://voparis-helio.obspm.fr/hfc-gui-dev/>

VII. Autres interfaces

1) SolarSoft

Le logiciel IDL SolarSoft (SSW) fournit également une bibliothèque de routines permettant d'utiliser HELIO, et plus particulièrement le HFC. Cette bibliothèque (« package »), qui se trouve dans le sous-dossier `$SSW/vobs/helio` (pour installer le package SSW/HELIO, consulter http://www.lmsal.com/solarsoft/ssw_install_howto.html ou bien http://www.helio-vo.eu/documents/help/ssw/helio_installing_software.html), est maintenue à jour par Bob Bentley. L'ensemble des programmes relatifs au HFC est disponible dans le dossier `/home/helio/hfc/interfaces/api/idl/ssw/`.

2) API IDL pour le HFC

Indépendamment du package SSW-HELIO, un client IDL, téléchargeable depuis la page web du HFC, permet d'interroger le catalogue via des requêtes simplifiées.

La dernière version du code source est disponible dans le dossier `/hfc/trunk/interfaces/api/idl/hfc_idl_client` du dépôt SVN du LESIA.

3) API Java

HELIO possède sa propre API java développée par Marco Soldati du FHNW permettant d'interroger les différents services via le HQI.

4) API Python

Une simple application Python permet de visualiser les images et données du HFC. Cette application, encore en cours de développement, est stockée sur le dépôt SVN dans le dossier `/hfc/trunk/interfaces/api/python/hfcviewer`. Elle a été testée sous Python 2.7.1.

La liste suivante référence les nom et version des modules python nécessaires pour exécuter le programme :

- Matplotlib 1.2.1
- Scipy 0.12.0

HFC : guide technique
Version 2.1

- PIL 1.1.7
- Numpy 0.1

Pour lancer l'application, exécuter le script *hfcViewer.py* du sous-dossier *hfcViewer/*.

5) HELIO Front End

HELIO possède son propre portail d'accès, le HELIO Front End (HFE), accessible depuis l'adresse :

<http://hfe.helio-vo.eu>

Le HFC n'est pas (encore ?!) intégré à ce portail. (On pourra trouver plus de détails dans le document HELIO_HFC-HFE_connection.pdf).

VIII. Codes de détection

1) Au LESIA

tycho.obspm.fr

Le compte helio du serveur *tycho.obspm.fr* héberge et exécute sur la grappe de calcul de l'Observatoire de Paris [RD12], la quasi-totalité des codes de reconnaissance du HFC gérés par le LESIA. L'utilisation de la grappe de calcul permet une gestion optimale de l'exécution des codes, en assurant une disponibilité quasi-continue de serveurs de calcul, tout en assurant une sauvegarde régulière des fichiers et données. Elle permet aussi notamment de réduire significativement le temps de calcul de certains codes de détection, en parallélisant l'exécution des jobs sur la grille.

Les codes suivant sont actuellement installés sur le serveur :

- **MDISS** : Code de détection des tâches solaires sur les images SoHO/MDI.
- **SDOSS** : Code de détection des tâches solaires sur les images SDO/HMI.
- **RABAT2** : Code de détection des sursauts radio solaires de type II sur les spectres dynamiques produits par Wind/Waves/Rad2 et STEREO/Waves/HFR. Ce code n'est pas encore validé !
- **RABAT3** : Code de détection des sursauts radio solaires de type III sur les spectres dynamiques produits par Wind/Waves/Rad2, STEREO/Waves/HFR, et le réseau décimétrique de Nançay.
- **SPOCA** : Code de détection des régions actives et des trous coronaux sur les images SoHO/EIT (171, 195) et SDO/AIA (171, 193).
- **TRACKFIL** : Code de suivi (tracking) des filaments solaires détectés
- **NRH2D** : Code de détection des sources radio sur les observations 2D du Radio Héliographe de Nancay, NRH (images obtenues par déconvolution des données interférométriques).
- **TRACKRS** : Code de tracking des sources radio détectées par NRH2D (code déplacé sur *voparis-helio* pour le moment).
- **NOAA** : Code d'identification des régions NOAA avec les tâches solaires et régions actives.

La Figure 5 présente l'arborescence des dossiers et sous-dossiers principaux accessibles depuis le compte *helio* sur *tycho.obspm.fr*.

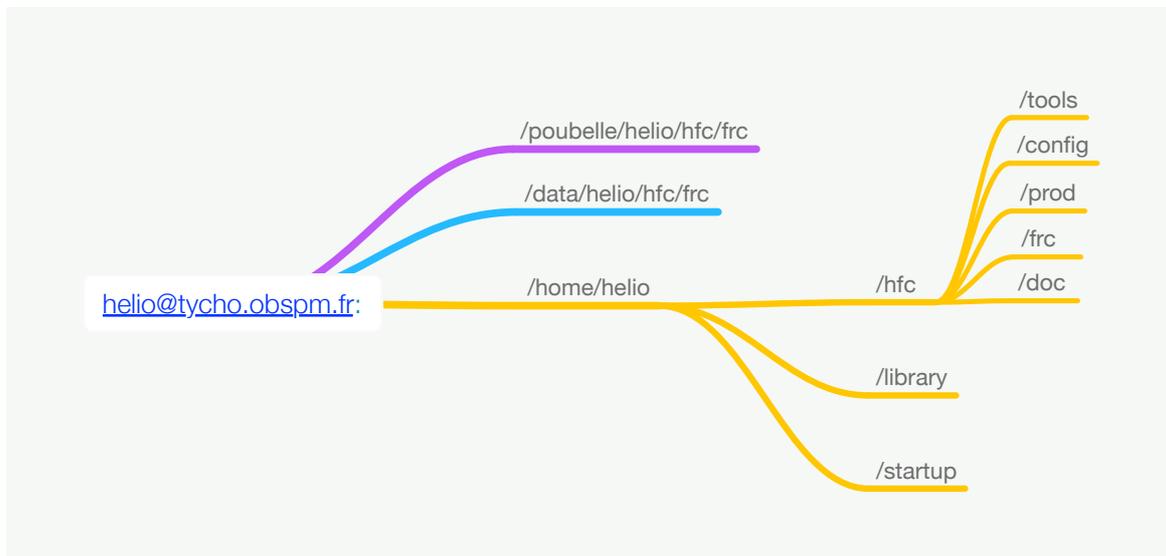


Figure 5. Arborescence du compte *helio* sur *tycho.obspm.fr*.

A noter que :

- Les dossiers */poubelle/helio/hfc/frc* et */data/helio/hfc/frc* accueillent les sous-dossiers contenant respectivement les fichiers de données à traiter par les codes (e.g., images au format fits, fichiers binaires ou ascii, etc.), et les fichiers produits par les codes.
- Le dossier */home/helio/hfc/frc* regroupe les codes sources des programmes de détection (un sous-dossier par code).
- Le dossier */home/helio/hfc/prod* contient les scripts et programmes permettant de gérer l'exécution automatique des codes, ainsi que la copie des fichiers produits sur le serveur ftp de Bass2000.
- Le dossier */home/helio/library* contient des bibliothèques de routines additionnelles (en IDL, Python, etc.) susceptibles d'être appelées par certains codes.
- Le dossier */home/helio/startup* contient les scripts et programmes prises en compte lors de la connexion au compte.

Le LESIA gère par ailleurs les codes SoSoft et SoSoPro, dédiés respectivement à la détection des filaments solaires et des protubérances sur les images du Soleil en Halpha. produites par le Spectro-héliographe de Meudon. Pour plus d'information sur ces deux codes, contacter Nicolas Fuller.

2) Au Trinity College Dublin

Le Trinity College of Dublin (TCD) fournit les données des codes suivants:

- SMART : Code de détection des régions actives sur les images SoHO/MDI et SoHO/EIT.

- CHARM : Code de détection des trous coronaux sur les images de SoHO/EIT.

L'ensemble des fichiers csv produits est déposé sur le serveur ftp [ftp ftpbass2000.obspm.fr](ftp://ftp.bass2000.obspm.fr).
Pour plus d'information sur ces codes, contacter David Perez-Suarez.



La production des fichiers pour SMART et CHARM n'est plus assurée par le TCD depuis la fin du projet HELIO (c.-à-d., fin 2012).

3) **FTP**

L'ensemble des fichiers produits par les codes de détection sont stockés sur le serveur ftp [ftp *ftp*bass2000.obspm.fr](ftp://ftp.bass2000.obspm.fr), qui sert de source principale pour l'insertion des données dans les bdds du HFC.

Les fichiers au format csv contenant les paramètres des structures détectées, ainsi que des informations sur le code, l'observation, et l'instrument utilisés sont disponibles à l'adresse :

<ftp://ftp.bass2000.obspm.fr/pub/helio/>

Les imageries (quicklooks) produites par les codes sont copiées dans le dossier :

<ftp://ftp.bass2000.obspm.fr/temp/qlk/>

Annexes

1) Schéma de la base de données

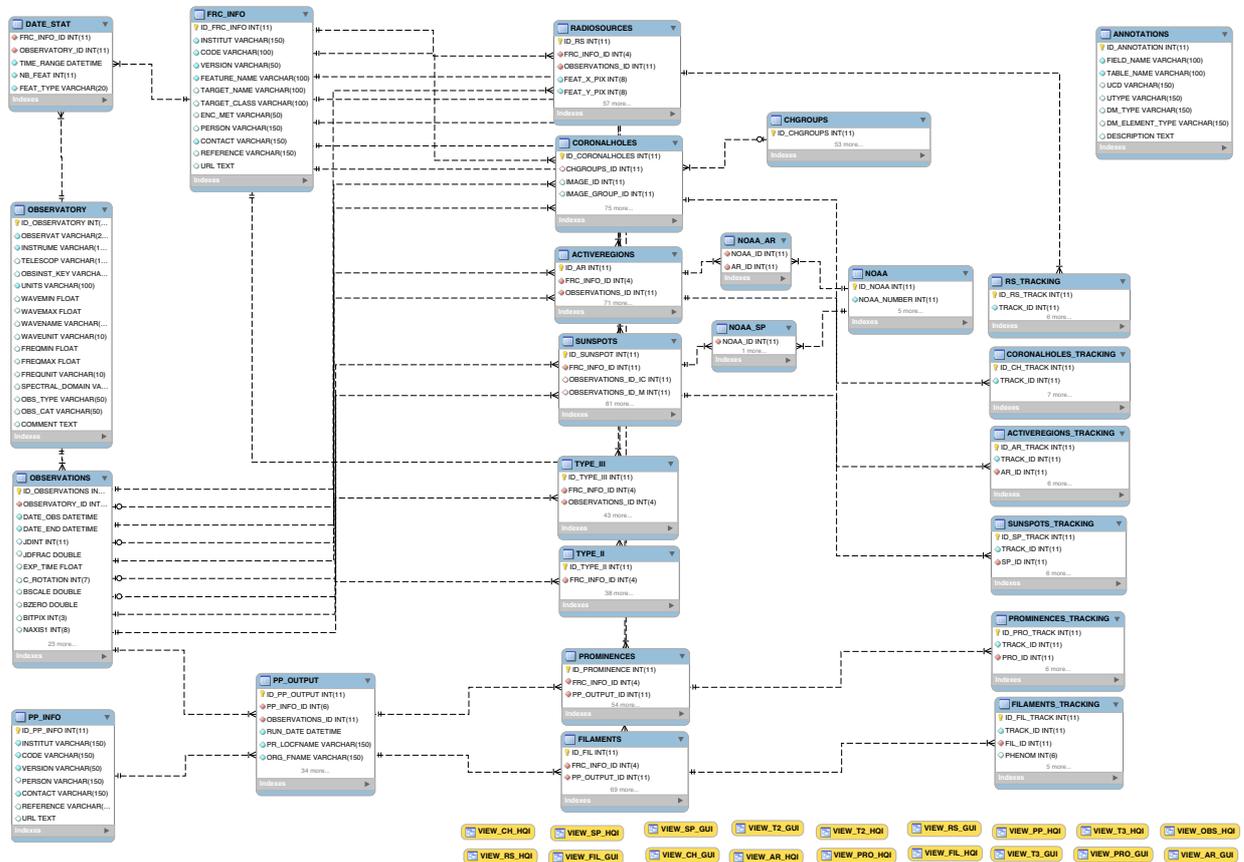


Figure 6. Schéma de la base hfc1 sur voparis-mysql5-paris

2) Outils de gestion

Plusieurs outils collaboratifs existent pour gérer le développement et la maintenance du HFC, n'hésitez pas à la utiliser !

Dépôts SubVersion

Le développement et la maintenance du HFC s'appuient sur deux dépôts SubVersion (SVN) :

- Au niveau LESIA - <https://version-lesia.obspm.fr/repos/helio/>
- Au niveau HELIO - <http://sourceforge.net/projects/helio-vo/>

Le dossier /hfc du dépôt du LESIA regroupe l'ensemble des fichiers, programmes, et codes sources permettant de maintenir et faire tourner l'ensemble de la chaîne de production du HFC. Le dossier /hqi est son équivalent pour tout ce qui concerne l'interface de requête HELIO.

HFC : guide technique
Version 2.1

Chaque dossier propose 3 sous-dossiers tels que recommandés pour la gestion d'un dépôt SVN :

- /trunk - Dossier de développement principale
- /tags - Contient l'historique des versions du service (dont la dernière version doit correspondre à la version stable de production.)
- /branches – Peut être utilisé pour réaliser des développements parallèles à la branche principale du trunk.

 Avant de démarrer tout développement, on s'assurera d'avoir la dernière version disponible du dépôt SVN (récupérable via la commande « svn update »), et l'on pensera à mettre à jour le dépôt avant de modifier les fichiers et programmes de production.

Projet Redmine

Une page Redmine regroupant un certain nombre d'informations et de documents sur HELIO et le HFC est accessible depuis le site des projets Redmine du LESIA :

<https://projets-lesia.obspm.fr/projects/helio-features>

Contacts

Jean Abouharham – *Responsable scientifique de HELIO au LESIA* :
jean.abouharham@obspm.fr

Christian Renié – *Responsable technique de HELIO au LESIA* :
christian.renie@obspm.fr

Xavier Bonnin – *Développement et maintenance de la base de données, des interfaces, et des codes de détection* :
xavier.bonnin@obspm.fr

Renaud Romagnan – *Développement et maintenance des codes SoSoFT et SoSoPro, et des interfaces utilisateurs* :
renaud.romagnan@obspm.fr

DIO – *Division Informatique de l'Observatoire de Paris* :
admin.dio@obspm.fr

David Perez-Suarez – *Développement et maintenance des codes SMART et CHARM* :
dps.helio@gmail.com

Kevin Besson – *Développement et maintenance du HQI, et des services HEC, ILS, HRS, et ICS hébergés au MSSL* :
kmb@mssl.ucl.ac.uk

Bob Bentley – *Principal Responsable du projet HELIO* :
rdb@mssl.ucl.ac.uk

