# Introduction du nouveau Centre de Données Biomedicales Décrypthon

**Hoan Nguyen — Anne Friedrich — Guillaume Berthommier — Laetitia Poidevin — Luc Moulinier — Raymond Ripp — Olivier Poch**

*Laboratoire de BioInformatique et Génomique Intégrative, UMR 7104*
*Institut de Génétique et de Biologie Moléculaire et Cellulaire*
*1 rue Laurent Fries, 67404 Illkirch – Strasbourg France*

*{nguyen,friedrich, berthomg,poidevin,moumou,ripp, poch}@igbmc.fr*

ABSTRACT. *This paper presents the design and the implementation of the new high performance biomedical data center of the Décrypthon computing grid which provides a strong potential for calculation and storage to high trhoughput biological applications and projects. In order to efficiently share the biological data required by the application, the Décrypthon data center is integrated in the computing grid to provide local databases of nucleotide, genomic and proteomic sequences. In addition, the access to methods for heterogeneous and distributeddata, and treatment of joint queries, analysis and visualization are provided. A new system of data integration, called BIRD (for Biological Integration and Retrieval of Data), considered as the core of the Décrypthon data center, was developed to locally integrate very large genomic, proteomic and transcriptomic datasets. BIRD also provides an engine and a high level query language allowing the biologist to extract pertinent information.*

MOTS-CLÉS : *base de données biologiques, intégration des données, bioinformatique, langage de requêtes biologique, BIRD-QL, grille Décrypthon.*

KEYWORDS: *biological database, data integration, bioinformatics, biological query language, BIRD-QL, Décrypthon Grid.*

## 1. Introduction

Depuis 2001, trois acteurs majeurs dans leur domaine, l'AFM, Association Française contre les Myopathies, le CNRS, organisme de recherche et IBM, société d'informatique, ont entamé une collaboration autour du Programme Décrypthon afin de développer une plate-forme de *Grid Computing*. L'objectif est de faciliter les analyses génomiques et protéomiques pour progresser dans la compréhension des maladies génétiques, notamment des maladies neuromusculaires. La grille de calcul Décrypthon est constituée des supercalculateurs des centres universitaires de Bordeaux 1, Lille 1 (USTL) et Pierre et Marie Curie (Paris 6 Jussieu). L'ensemble de ces ressources est connecté par le réseau à haut débit RENATER 2 (Réseau National de Télécommunications pour la Technologie, l'Enseignement et la Recherche) et comprend plus de 100 processeurs Power 4 ou 5 d'IBM.

Le développement de la grille Décrypthon fait écho à l'émergence de nouvelles technologies à haut débit en biologie, regroupées sous le terme générique de « génomique fonctionnelle » et recouvrant les notions de transcriptomique, protéomique, interactomique, métabolomique... Ces méthodes ont abouti à des taux de production et d'hétérogénéité des données générées qui dépassent largement les capacités d'analyse et d'expertise humaine ainsi que les possibilités de traitement des plus puissants ordinateurs. Dans ce contexte, des développements sont nécessaires pour, d'une part, assurer le déploiement d'un système capable de gérer de gros volumes de données hétérogènes et de traiter rapidement des requêtes croisées entre différentes sources de données et d'autre part, créer des systèmes d'extraction de connaissances efficaces et pertinents capables de traiter les données fortement bruitées de la génomique fonctionnelle. Un tel système d'extraction de connaissances recouvre un processus itératif piloté par les connaissances elles mêmes incluant de nombreuses étapes de génération, épuration, validation, comparaison, analyse et représentation des données aboutissant à une nouvelle connaissance susceptible de relancer l'ensemble du processus. Dans une telle approche, l'intégration des données biologiques dans un système adapté représente un défi majeur puisqu'il sous-tend l'ensemble des inférences biologiques à la base des hypothèses qui seront ultérieurement testées expérimentalement.

Les analyses bioinformatiques actuelles nécessitent l'accès à un grand nombre de banques de données dont les tailles interdisent un transfert en chaque nœud d'une grille de calcul. Dès lors, il a fallu développer une infrastructure permettant d'intégrer les banques de données, accessibles tant par les nœuds de calcul que par les utilisateurs de la grille. De plus, la plupart des applications de calculs biologiques sont amenées à traiter des données (séquences nucléotidiques ou protéiques, transcriptomes, etc.) dont les formats sont de type « fichier plat ». Cette manière d'exploiter les données fait appel à de nombreuses opérations répétées de transferts des fichiers originaux qui consomment énormément de temps dans une grille de calculs.

Actuellement, plusieurs approches et systèmes d'intégration (Olund *et al*., 2007 ; Hernandez et Kambhampati, 2004) sont disponibles pour créer et déployer facilement des bases de données hétérogènes. Ces approches sont brièvement classifiées en trois types : fédération, entrepôt de données et indexation. Dans l'approche fédérative, BioBank (Olund *et al.,* 2007 ; Muilu *et al.,* 2007) est un projet emblématique pour fédérer des bases de données sous une nouvelle architecture et offrir des services assurant la gestion des données biomédicales en utilisant la plate-forme Websphere Information Integrator -WII (Haas *et al..,* 2001). WII a été mise au point pour relever le défi de l'intégration et de l'analyse des données scientifiques des divers domaines des Sciences du vivant. WII permet la création aisée d'une base de données virtuelle (schéma virtuel) à partir de Web Service et de bases de données relationnelles à distance. Cette technologie est totalement adaptée à la fusion de bases de données cliniques à distance avec des bases de données relationelles locales. Dans l'approche dite d'entrepôt de données, BioWarehouse (Lee *et al.,* 2006) est un outil récent permettant la création d'un entrepôt de données biologiques. Cet outil est capable d'intégrer de multiples sources d'informations provenant des génomes, des ressources métaboliques ou d'enzymes dans une base relationnelle gérée par Oracle/MySQL. Cependant, la complexité et l'inter-dépendance du modèle de données rendent délicate l'intégration de nouvelles sources de données. Finalement, dans le cadre de l'approche dite d'indexation, le Sequence Retrieval System-SRS est largement utilisé pour l'indexation des banques biologiques. SRS est un système d'indexation en texte entier des fichiers en « format plat ». Toutefois, SRS n'est pas un système relationnel permettant l'opération de mise à jour fondée sur les transactions atomiques. Il ne convient pas bien à la gestion du contenu scientifique et de l'extraction direct des connaissances dans une base de données relationnelle (Bertone *et al.,* 2001).

Dans ce contexte, le Laboratoire de Bioinformatique et Génomique Intégratives (LBGI) de l'IGBMC de Strasbourg travaille à la conception d'un nouveau système d'information bio-médical, nommé BIRD - Biological Integration and Retrieval of Data, fédérant des données généralistes (séquences, structures, fonction, évolution, etc.) et des données spécialisées de la biologie à haut débit (protéomique, transcriptomique, interactomique, phénomique, etc.). BIRD intègre efficacement les données hétérogènes dans une base de données relationnelle en utilisant des modèles de données configurables et des règles de configuration. Il s'est équipé d'un moteur de requête original afin de répondre aux requêtes de haut niveau. Il utilise WII pour fusionner les bases de données locales et distantes. Ce système BIRD est le cœur du nouveau centre de données de haute performance qui est directement intégré dans la grille de calculs Décrypthon. L'objectif de ces travaux est d'étudier et de décrire le réseau de relations complexes existant entre une ou des mutations et le phénotype des maladies génétiques humaines (par exemple, myopathies, rétinopathies, etc.). Ce centre permet aussi la mutualisation des données nécessaires à la réalisation des différents projets déployés sur la grille Décrypthon ainsi qu'un accès simplifié aux données pour la communauté scientifique.

Cet article s'organise en 4 parties. La section 1 présente la motivation du projet, la section 2 parle de la génération du centre. La conception et l'implémentation de système BIRD seront présentées dans la section 3. Finalement, les conclusions et perspectives seront discutées dans la section 4.

## 2. Centre de données Décrypthon

### 2.1. *Objectif général*

Le centre fournit un environnement efficace pour la recherche biomédicale comprenant une base de données locale de séquences nucléotidiques et protéiques, les méthodes d'accès aux données hétérogènes et distribuées, le traitement des requêtes complexes, l'analyse et la visualisation. Un nouveau système d'intégration, nommé BIRD (voir section 3), considéré comme le noyau du centre de données Décrypthon, a été développé pour intégrer localement un grand volume de données biologiques publiques (des miroirs des banques de données). BIRD est équipé d'un moteur et d'un langage de requête de haut niveau BIRD-QL permettant aux biologistes de rechercher des données et de les extraire sans écrire de programmes.

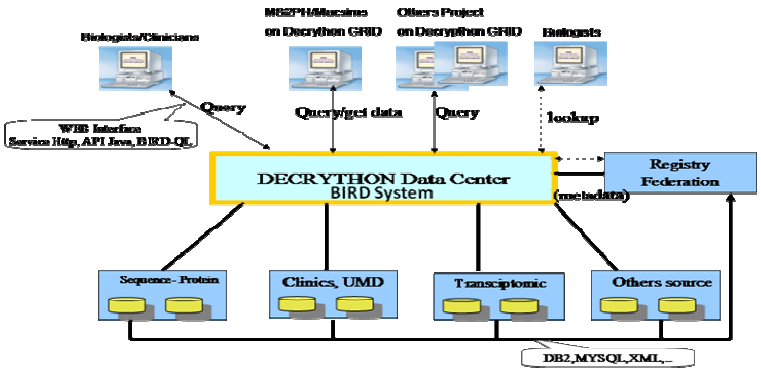


**Figure 1.** *Architecture fonctionnelle du centre de données Décrypthon*

La Figure 1 illustre l'architecture fonctionnelle du centre de données Décrypthon, qui permet de fusionner des données de différentes sources et de partager ces données avec les applications qui tournent sur la grille de calcul Décrypthon et avec la communauté biologique. Le centre Décrypthon partage ses données intégrées avec des applications/clients et des utilisateurs de plusieurs manières : par l'intermédiaire d'une API Java, d'une interface web et du langage BIRD–QL via des services web (HTTP).

### *2.2. Architecture de déploiement pour le Centre de Données Décrypthon*

BIRD qui sera présenté dans la section 3 repose sur une architecture multi-couches contenant plusieurs modules différents. Il utilise DB2 comme unité de stockage et d'indexation de données massives. Il fournit des API Java et des services de requêtes à plusieurs applications de haut niveau. WII est utilisé comme une couche de données virtuelle pour fusionner les données locales et des données cliniques de type UMD (Beroud *et al.*, 2000) à distance. Un module « Registrer » permettra de publier automatiquement des données sous les formats XML et tables relationnelles. Grâce à ce module, il leur sera aussi possible de rechercher des méta-données d'une source de données enregistrée.

Le moteur de requête BIRD-QL est une composanse réutilisable pour les services d'accès aux bases de données comme Serveur Web, API et d'autres modules scientifiques. L'avantage du moteur BIRD-QL est de permettre aux développeurs de haut niveau dans BIRD et aux utilisateurs (biologistes) de développer leurs applications de manière indépendante à la structure du système de bases de données, aux sources de données hétérogènes et aux langages de programmation. La Figure 2 illustre l'architecture adaptée pour le centre de données Décrypthon.

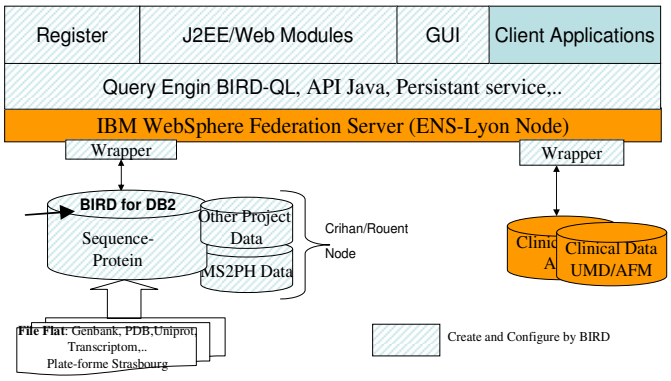


**Figure 2**. *Architecture de déploiement du BIRD au centre de données*

### *2.3. Les contraintes d'intégration des données*

Le centre Décrypthon peut fédérer et fusionner des bases de données locales et distantes de différents types (Table 1) et il est déployé pour répondre à trois contraintes décisives :

– Volume des données : les données biologiques de séquences, de structures et d'alignements représentent environ un Tera-octet. Ce volume double tous les ans. Une procédure de mise à jour hebdomadaire est également intégrée.

– Hétérogénéité des données : modèles tridimensionnels, alignements de séquences, pouvant aller jusqu'à des images de gels ou des articles scientifiques. Les formats de ces données brutes sont très différents, d'où la nécessité de développer des « wrappers » (parseurs) à façon. Les données générées par les projets du programme Décrypthon doivent être stockées et indexées en temps réel dans le centre de données.

– Sécurité et confidentialité : ce centre est amené à intégrer des données phénotypiques provenant de patients. Il est, pour des questions évidentes de confidentialité et d'éthique, hors de question de dupliquer de telles bases ou de faire transiter des données sensibles sur des réseaux. L'information nécessaire au centre Décrypthon reste donc sous le contrôle des utilisateurs propriétaires qui permettent un accès à l'information pertinente et sécurisée dans leur base locale.

| Nom | Type Intégration | Format | Droit utilisateur | Source d'origine |
|---|---|---|---|---|
| Genbank | Schéma local | Flat genbank | Public | NCBI[1] |
| Refseq | Schéma local | Flat Genbank | Public | NCBI |
| Uniprot/uniref | Schéma local | Flat Embl | Public | UniprotKB[2] |
| Pdb | Schéma local | PDP | Public | PDB[3] |
| GO, SO,PSI et Pato Ontology | Schéma local ou Service Ontology Lookup EBI | OBO/OWL | Public | OBO Foundry[4] |
| OMIM | Schéma local | OMIM | Public | NCBI |
| Protein Interaction | Schéma local | PSI/XML | Décrypthon | Calculés par les projets du programme Décrypthon |
| Macsims | Schema local | XML | Décrypthon | Génération par la cascade de logiciels Macsims (Thompson *et al.*, 2006) fonction dans la grille |
| Genome UCSC | Schéma local | Table Relationelle | Public | UCSC[5] |
| Taxonomy | Schéma local | Table Relationelle | Public | NCBI |
| Phénotypes | Schéma virtuel | | Décrypthon | Génération par AFM ou clinique |
| Mutation UMD | Schéma virtuel | UMD | Décrypthon | Clinique |

**Table 1.** *Liste des banques de données hétérogènes intégrées par le centre de données Décrypthon. En général, ces banques de données sont intégrées sous deux types de schéma relationnel : local et virtuel. Les fichiers de données sous le format flat/XML sont*

[1] NCBI : National Center for Biotechnology Information

[2] UniprotKB : Universal Protein Resource

[3] PDB : RCSB Protein Data Bank

[4] OBO Foundry : Open Biomedical Ontologies

[5] UCSC : University of California Santa Cruz Genome Bioinformatics

*convertis en table relationnel. Les données cliniques à distance sont intégrées par le schéma virtuel en utilisant WII. Les fichiers de données biologiques publiques sont mis à jour par la plate-forme Bio-Informatique de Strasbourg(BIPS).*

La plupart des données intégrées au centre sont de type généraliste et peuvent donc être utilisées par l'ensemble des projets Décrypthon. L'accès à certains types de données confidentielles doit être limité aux utilisateurs ayant les droits d'exploitation. Le Table 1 montre en détail les banques de données et les sources de données intégrées par le centre. Ces données sont fournies par la plate-forme bioinformatique de Strasbourg.

### *2.4. Exploitation des données*

Le but de notre projet est de développer des interfaces (machine-machine) entre le centre de données et les applications qui fonctionnent sur la grille. Concrètement, les applications biologiques exploitent les données en utilisant le service de requête BIRD-QL via le protocole http, par exemple les logiciels PipeAlign et MACSIMS (Thompson *et al*., 2006) qui sont installés sur la grille et qui génèrent et gèrent l'information contenu au sein d'alignements multiples de séquences protéiques, exploitent des données du Centre. Toutefois, les interfaces web (homme-machine) contenant beaucoup de fonctionnalités intéressantes sont développées afin de fournir des données et des services de haut-débit à la communauté scientifique des non-experts.

***Fouille et extraction des données dans l'interface machine-machine***

Le moteur BIRD-QL est une fonctionalité de haut niveau permettant aux biologistes de fouiller facilement des données du Centre sans avoir besoin de connaissances en base de données. L'utilisateur peut développer et envoyer des requêtes BIRD-QL (voir l'exemple ci-dessous et section 3) via des commandes http ou des scripts pour rechercher des séquences intéressantes et recevoir des données sous différents formats. Ce moteur est également capable de générer rapidement et à la volée des sous-banques de données selon les besoins des applications.

**MaBIRDQL.txt**

```
ID * DB MyESTMouse, REFSEQ
WH OC Contains “Eukaryote” & not “Metazoa”
WH DR Contains “GO”
WH GENE contains “GF100027”
FD AC, SQ,OS
FM FASTA
```

**Figure 3.** *Exemple d'une requête BIRD-QL*

L'exemple de requête ci-dessus (Figure 3) permet de rechercher dans plusieurs banques (Genbank et RefSeq) des séquences contenant le gene GF100027 et une référence croisée dans la banque Gene Ontology, plusieurs informations incluant le numéro d'accès, la séquence et le nom de l'organisme seront extraites et fournies à un format (FASTA) compatible avec de nombreuses applications de bioinformatique.

***Applications Web***

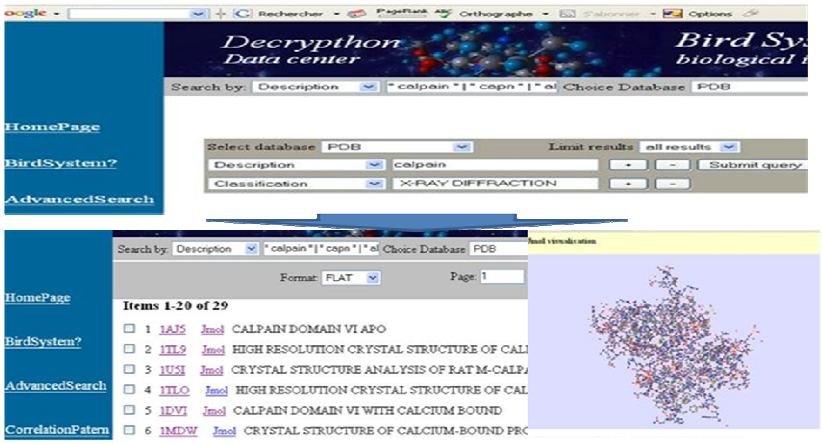


**Figure 4.** *Interface Web permettant de sélectionner et de naviguer dans les données intégrées par BIRD*

Ces interfaces (Figure 4) permettent aux utilisateurs de sélectionner des informations et de naviguer facilement *via* le contenu des bases de données herbégées ou fédérées dans le Centre. L'outil JMOL (http://jmol.sourceforge.net) a été utilisé pour visualiser les structures de trois dimensions.

## 3. Le système BIRD

BIRD est basé sur les idées principales de la configuration automatique et de l'architecture du système SAADA- Système d'Archivage Automatique des Données Astronomiques (Michel *et al.*, 2005 ; Nguyen *et al.*, 2006) et a été développé pour la gestion de données hétérogènes provenant de l'astronomie. Le but de BIRD est de créer une base de données locales prête à l'emploi, à partir de schémas relationnels préexistants qui peuvent servir de base pour configurer de nouvelles bases de données. Un modèle de données configurable générique a été conçu pour chaque

type de bases de données (UniProt, GenBank, PDB, RefSeq, InterPro, etc.). Ce modèle existe physiquement sous forme d'un fichier XML contenant un nombre limité de règles déclaratives permettant de définir le schéma relationnel de la base de données à créer. D'autres modules contiennent les règles pour analyser des fichiers de données, et permettent d'identifier le schéma relationnel auquel elles appartiennent.

BIRD est totalement écrit en Java et utilise les APIs publiques de la plateforme JAVA/J2EE. Les utilisateurs à distance peuvent exploiter les données hébergées par différentes méthodes, comme une interface Web, une API Java, ou des services HTTP (moteur BIRD-QL).

### *3.1. Intégration par les règles de configuration*

BIRD est capable d'intégrer des données biologiques (voir Table 1) selon des thèmes sémantiques et les besoins réels des utilisateurs. BIRD est un système déclaratif. L'opérateur (administrateur) déclare des règles que le système enregistre. Ces règles modélisent les caractéristiques des produits (banques) à charger. L'opérateur défini des règles dans des fichiers XML (voir Figure 5). Le contenu de ces fichiers de configuration est édité soit par une interface graphique soit manuellement.

```
<config_product  nameconfig="MyESTMouse">

    <product_identification type="expression pattern">
        <repository name="/Genbank/" recursive="no">
        <repository_update name="/Genbank-New/" recursive="no">
        <namefile pattern="*gbest"/>
        <repository_temp dir="/genbank_temporary" />
    </product_identification>
    <content_identification>
        <content node="organism" value="Mus musculus"/>
    </content_identification>
     <content_ignored> Reference, Comment   </content_ignored>
     <into-bird schema="MyESTMouse"  type =" GENBANK" />

</config_product>
```

**Figure 5.** *Exemple des règles de configurations.*

La description des produits se fait par configuration. Une configuration est un ensemble de règles permettant d'identifier et de charger des produits d'une même catégorie ou même type (séquence, protéine) et destinés à un même schéma donné

par l'opérateur. L'opérateur manipule cette notion de configuration qui masque en fait les tables relationnelles créées par BIRD. Cela lui évite d'avoir à faire une classification manuelle des produits. Le chargement de produits dans la base se fait pour une configuration donnée. Par contre, les configurations sont totalement invisibles pour les utilisateurs finaux.

Pour charger des produits, l'opérateur précise la configuration voulue ainsi qu'un répertoire de base. BIRD va alors explorer récursivement ce répertoire et tenter de charger tous les produits correspondant à la configuration.

La fonctionalité de chargement analyse le fichier de configuration et la structure des produits. A partir de là, si le schéma donné n'existe pas, elle produit un nouveau schéma relationel héritage à partir d'un schéma prédéfini enregistré dans les méta-données. Elle produit aussi une liste des produits associant les noms des produits identifiés à un schéma de données. Une fois la génération des nouveaux schémas ou la modification d'anciens schémas réussies, BIRD peut charger les fichiers (uniquement le contenu) de données dans la base conformément au schéma des produits.

Grâce à ces règles, nous pouvons modéliser le contenu d'un entrepôt de données ou une base de données. Ces règles nous permettent de scinder des banques de données en plusieurs bases de données relationelles et filtrer des données en fonction des besoins. L'exemple ci-dessus montre les règles de configuration des données pour créer facilement un schéma de données MyESTMouse en chargeant les fichiers orginaux de Genbank correspondant avec des règles de configuration simples.

### *3.2. Moteur de requêtes de haut niveau BIRD-QL*

Les données hétérogènes intégrées dans BIRD sont représentées sous plusieurs tables relationnelles. L'exploitation de ces données par des requêtes SQL n'est pas évidente sauf pour les développeurs ou les utilisateurs experts du domaine. Pour cette raison, BIRD s'est équipé d'un langage (Figure 6) et d'un moteur de requête de haut niveau : BIRD-QL, qui permet aux bioinformaticiens d'exprimer facilement des requêtes et d'extraire des connaissances. Il permet aussi de traiter des requêtes croisées entre des données biologiques et des données phénotypiques. Grâce à ce langage, nous avons pu facilement optimiser et contrôler les performances du système.

La structure de ce langage permet aux utilisateurs de manipuler facilement ses requêtes et de construire des conditions de recherche selon la logique booléenne. L'utilisateur peut ainsi filtrer les données sélectionnées en ajoutant des conditions au fur et à mesure. Ces conditions sont entièrement indépendantes les unes des autres et sont reliées par les opérateurs logiques *AND*. Le Table 2 donne le nombre possible d'utilisations de chaque clé et les contraintes dans une requête BIRD-QL. Les

méthodes de fouille de données comme Treille de Galois (Jay *et al.*, 2006) ou règles d'association (Agrawal *et al.*, 1993 ; Carmona-Saez *et al.*, 2006) sont potentiellement utilisées pour extraire des motifs d'information pertinents dans la base de données biomédicales. Nous allons donc intégrer les procédures de génération des règles d'association et de recherche de motifs (AssociationPattern) dans le moteur BIRD-QL.

**ID** <*list of id/ac/query_id* > **DB** <dbnames>
**WH** Field **Contains** kw1 |& kw2 |& kw_n
**WH** Field **Contains** kw1 |& kw2 |& kw_n
…
**WH** **PATTERN** <*function SequencePattern()* >
**WH** **PATTERN** <*function StructuralDistance()*>
**WH** **PATTERN** <*function AssociationPattern()*>
**WH** **PATTERN** <function InteractionProtein()
**WH** **BLAST** constraint
**FD** <*field-out*>
**FM** <*fasta/flat/xml/Object/OID*>
**LM** <*n*>

**Figure 6.** *Structure du motif BIRD-QL*

| Clé | Nom | Cardinalité | Exemple |
|---|---|---|---|
| ID | Séquence id, protéine id, query id | 1 | ID Q92PK5 |
| WH | Where | 0..N | WH OS contains « Homo sapiens » |
| WH BLAST | Contrainte blast | 0..N | WH Blast score >1 |
| WH PATTERN | Motifs/ Fonctions scientifiques prédéfinies dans BIRD | 0..N | **StructureDistance**(PDB_id,x,y,z) >10<br>**AssociationPattern***(Right(protA,ProtB,ProtC),Left(ProtK),sup=30,conf=90)<br>*Futur travail: règles d'association |
| FD | Paramètres de sortie ou une des valeurs d'une fonction scientifique | 1 | FD AC,ID,SQ<br>FD Get_Count(ID)<br>FD GetStatitic(Taxid,OS,)<br>FD **AssociationPattern(..)** |
| LM | Limite | 0..1 | LM 10 |
| FM | Format | 0..1 | FM fasta/embl/genbank/xml |

**Table 2.** Liste des clés de BIRD-QL et exemple de requête

Afin d'être en adéquation avec le modèle des données biologiques et les besoins réels des biologistes, nous avons été amenés à développer notre propre langage de requêtes BIRD-QL que l'on peut assimiler à un « dialecte » fait d'assemblage de SQL et de fonctions propres à BIRD visant à exploiter au maximum le moteur SQL du IBM DB2. BIRD-QL peut utiliser les fonctions de recherche en texte entier du DB2 pour l'exécution de requêtes portant sur les contraintes de texte entier.

La structure des tables relationnelles étant masquée, BIRD-QL facilite la création d'interfaces graphiques lisibles et aide les utilisateurs à dialoguer plus facilement avec les bases générées.

Un des avantages de BIRD-QL est lié au fait que l'on peut envoyer cette requête à n'importe quelle banque de données hébergée par BIRD ou à plusieurs instances BIRD à la fois par des Web Services ou par une API Java. Cette requête ne dépend pas des données intégrées (noms des banques de données, tables relationnelles) dans chaque base BIRD.

Les requêtes BIRD-QL sont traitées par plusieurs opérations complexes telles que la décomposition des contraintes ou la recombinaison des résultats des différentes sous-requêtes. Dans tous les cas, le résultat d'une requête est une liste d'OIDs (Object Identifier). Suivant une logique purement objet, on imaginerait le traitement des requêtes se faisant sur des collections d'objets persistants et retournant des listes d'instance venant de différentes classes (format hétérogène : Uniprot, Genbank, etc.). Cette approche serait pénalisante en termes de mémoire et de vitesse. Pour être plus performant, le moteur de requêtes contourne la couche objet. Il accède directement à la base de données par SQL et ne retourne que des OIDs. Cela permet de ne pas faire transiter sur le réseau ou en mémoire des données inutiles. On peut ainsi facilement faire des requêtes retournant un grand nombre de données sachant que ne seront copiés en mémoire que les objets réellement lus.

## 4. Conclusions et perspectives

Nous avons mis au point la première version de BIRD qui nous a permis d'implémenter facilement le centre de données Décrypthon. Ce centre de données est capable de gérer et d'extraire des données biologiques ou médicales complexes. Il est également considéré comme un nœud de stockage qui gère en temps réel les résultats de sortie des applications scientifiques. L'intégration directe des bases de données réutilisables dans la grille de calculs nous permet de mieux optimiser l'exploitation des données de grand volume et d'éviter également la présence de données redondantes sur la grille. Notre premier résultat a marqué une phase importante dans le processus d'extraction des connaissances dans une base de données relationnelles.

L'approche « intelligence artificielle » sera utilisée pour extraire automatiquement, dans ces bases de données complexes, des connaissances et des relations concernant des gènes, protéines, niveaux de production ou maladies cibles.

Ce futur travail impliquera le développement de nouveaux modèles de gestion et d'extraction des connaissances présentes dans les bases de données biologiques intégrées par BIRD. A l'issue de ce développement, un biologiste ou clinicien devrait pouvoir recueillir des informations synthétiques et décisionnelles sur des gènes ou des maladies. Les méthodes de fouille de données par des règles d'association seront implémentées pour analyser des motifs masqués d'informations pertinentes dans la base de données. Cette méthode de fouille pourra être utilisée pour optimiser la structure de la base de données et contruire une interface web intelligente s'appuyant sur les motifs de requêtes fréquentes provenant du Web Server ou du moteur BIRD-QL. Ces modèles de fouilles de données donneront lieu à une cartographie (cartographie sémantique des données de type Cartes auto-organisatrices), ou visualisation, des connaissances construites au travers d'ensemble de mot-clés liés (familles de protéines, maladies génétiques, etc.). Cette cartographie sera utilisée pour exploiter les données au moyen de systèmes de requêtes élaborées, véritables navigateurs qui mettront à profit les diverses régions (ou thématiques) de notre système.

## Remerciements

Ce projet est soutenu en majorité par le Programme Décrypthon (AFM, CNRS, IBM) et en partie par le projet européen EVI-GENORET Integrated Project LSHG-CT-2005-512036 . Nous remercions également les personnes ayant contribué à ce projet : M. Thierry Toursel (AFM), Mme. Anne-Marie Armbruster (IBM), M. Hervé Facci (IBM), M. Raphael Bolze (ENS-Lyon), Nicolas Bard (ENS-Lyon), Mme. Béatrice Charton (Crihan), M. Frédéric Plewniak (BIPS), Mme. Véronique Geoffroy(BIPS) et M. Serge Urge (IGBMC).

## Bibliogragphie

Agrawal R., Imielinski T., Swami A., « Mining association rules between sets of items in large databases », *Proc. ACM SIGMOD*, 1993, p. 207–216.

Beroud C., Collod-Beroud G., Boileau C., Soussi T., Junien C., « UMD (Universal mutation database): a generic software to build and analyze locus-specific databases », *Hum Mutat*, vol. 15, 2000, p. 86-94.

Bertone P., Gerstein M., « Integrative data mining: the new direction in bioinformatics », *IEEE Eng Med Biol Mag*, vol. 20, n°4, 2001, p. 33-40.

Carmona-Saez P., Chagoyen M., Rodriguez A., Trelles O., Carazo J., Pascual-Montano A., « Integrated analysis of gene expression by association rules discovery », *BMC Bioinformatics*, 7: 54, 2006.

Centre de données biomdélicales Décrypthon, http://d1.crihan.fr:8080/bird/index.html.

Crihan - Centre de Ressources Informatiques de HAute-Normandie

http://www.crihan.fr/calcul/decrypthon.

Haas L.M., Schwarz P.M., Kodali P., Kotlar E., Rice J.E., Swope W.C., « DiscoveryLink: A system for integrated access to life sciences data sources », *IBM System Journal,* vol. 40, n°2, 2001.

Hernandez T., Kambhampati S., « Integration of biological sources: current systems and challenges ahead », *ACM SIGMOD*, 2004.

Lee T.J., Pouliot Y., Wagner V., Gupta P., Stringer-Calvert D.W., Tenenbaum J.D., Karp P.D., « BioWarehouse: a bioinformatics database warehouse toolkit », *BMC Bioinformatics,* 7: 170, 2006.

JMOL, http://jmol.sourceforge.net

Thompson J.D., Muller A., Waterhouse A., Procter J., Barton G.J., Plewniak F., Poch O., « MACSIMS : multiple alignment of complete sequences information management system », *BMC Bioinformatics*, 7: 318, 2006.

Jay N., Kohler F., Napoli A., « Using Formal Concept Analysis for mining and interpreting patient flows within a healthcare network », *Fourth International Conference on Concept Lattices and Their Applications-CLA,* 2006.

Michel L., Nguyen H., Motch, C., « SAADA: Astronomical Databases Made Easier », *Astronomical Data Analysis Software and Systems XIV,* M. B. Edited by Shopbell P. and Ebert R., Pasadena, California, USA, ASP Conference Series, vol. 347, 2005.

Muilu J., Peltonen L., Litton, J.E., « The federated database – a basis for biobank-based post-genome studies, integrating phenome and genome data from 600 000 twin pairs in Europe », *European Journal of Human Genetics*, vol. 15, 2007, p. 718–723.

Nguyen H., Michel L., Motch C., « Building an Astronomical Database with Saada », *Astronomical Data Analysis Software and Systems XV*, Madrid, Spain, Astronomical Society of the Pacific, ASP Conference Series, vol. 351, 2006.

Olund G., Lindqvist P., Litton J.E., « BIMS: an information management system for biobanking in the 21st century », *IBM Systems Journal***,** vol. 46, n° 1, 2007, p. 171 – 182.

Programme Décrypthon, http://www.decrypthon.org.