

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

RAPPORT DE PROJET DE 15 CRÉDITS PRÉSENTÉ À
L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

COMME EXIGENCE PARTIELLE
À L'OBTENTION DE LA
MAITRISE EN GÉNIES DES TECHNOLOGIES DE L'INFORMATION

PAR
Boby LAFORET

RÉTRO-INGÉNIERIE D'UN LOGICIEL EXISTANT DU DOMAINE DE LA SANTÉ EN
PRÉPARATION POUR SA MODERNISATION

MONTRÉAL, LE 15 JUILLET 2015



Boby Laforet, 2015



Cette licence [Creative Commons](https://creativecommons.org/licenses/by-nc-nd/4.0/) signifie qu'il est permis de diffuser, d'imprimer ou de sauvegarder sur un autre support une partie ou la totalité de cette œuvre à condition de mentionner l'auteur, que ces utilisations soient faites à des fins non commerciales et que le contenu de l'œuvre n'ait pas été modifié.

PRÉSENTATION DU JURY

CE RAPPORT DE PROJET A ÉTÉ ÉVALUÉ

PAR UN JURY COMPOSÉ DE :

M. Alain April, directeur de projet
Génie logiciel et Technologies de l'information à l'École de technologie supérieure

M. Abdelaoued Gherbi, président du jury
Génie logiciel et Technologies de l'information à l'École de technologie supérieure

REMERCIEMENTS

Être en emploi à temps plein et être aux études à temps plein constituent pour moi le plus grand défi à relever pour ce projet de recherche. En dépit de tout, avoir une famille à supporter pendant qu'on étudie mérite d'énormes efforts et de concessions de part et d'autre.

J'aimerais d'abord remercier M. Alain April, mon directeur de projet, pour son encadrement impeccable et infatigable. Son aide et ses conseils m'aident à réaliser beaucoup de progrès au moment même que je travaille sur le projet. Il est l'encadreur qu'un étudiant doit avoir lors de son projet. Un merci spécial pour son dévouement constant à aider ses étudiants à progresser dans leurs études. Un remerciement spécial à ma conjointe Stéphanie Mésias pour m'avoir appuyée durant toutes les études. Je la remercie également de m'avoir encouragé à ne pas abandonner quand je suis affaibli par la grande charge de travail à accomplir. Je la remercie pour la confiance placée en moi en dépit des moments difficiles que nous avons connus ensemble. Je tiens à remercier mon cher enfant Bob Stéphane Porkendorff qui croit que son papa est le meilleur papa du monde, qu'il est le plus fort.

Je remercie également mes frères Carlo et Fritz, qui malgré la distance qui nous sépare, m'appellent souvent pour m'encourager dans l'effort que je déploie pour faire cette maîtrise. Ils me rappellent souvent que c'était mon rêve et que je dois le réaliser.

Finalement, je tiens à remercier mes collègues Macklyvens, Haendel, Jean Roosevelt qui m'encouragent toujours de ne pas abandonner malgré les difficultés.

RÉTRO-INGÉNIERIE D'UN LOGICIEL EXISTANT DU DOMAINE DE LA SANTÉ EN PRÉPARATION POUR SA MODERNISATION.

Boby LAFORET

RÉSUMÉ

Plusieurs entreprises, pour leurs besoins internes, font développer des logiciels capables de les aider à satisfaire ces besoins. Certains logiciels, développés en interne, sont parfois abandonnés en cours de développement ou même délaissés après avoir été développés. Dans certains cas, ces systèmes sont délaissés sans documentations importantes ou même lorsque les documentations sont présentes, ils n'étaient pas conçus pour faciliter la compréhension du système. La rétro-ingénierie est une technique très répandue permettant de recouvrir d'un côté la conception du système, mais également toutes connaissances utiles pouvant aider sa maintenance et son évolution.

De nos jours, la technologie informatique devient le pilier essentiel sur lequel s'appuient les chercheurs dans le milieu médical. Avoir un système logiciel capable de les aider dans l'expérimentation; dans l'analyse et la compréhension des résultats est un incontournable surtout dans le domaine de la recherche en génomique. Le défi majeur est de les aider à recouvrir la conception et toutes les connaissances utiles pouvant les permettre de moderniser un système de recherche génomique délaissé par l'équipe de développement. Ce projet vise d'abord à présenter une revue de la littérature sur le domaine de la rétro-ingénierie logicielle tout en abordant le sujet sur la réingénierie logicielle. Cette revue aborde certains aspects de l'historique; des systèmes patrimoniaux, de certaines définitions du domaine, des modèles du domaine, et les risques associés. En second temps, la revue est préparatoire à l'application de la rétro-ingénierie dans un projet afin d'extraire les connaissances du système. Cette approche confirmera que la maintenance et l'évolution d'un système nécessitent le support de la rétro-ingénierie.

RÉTRO-INGÉNIERIE D'UN LOGICIEL EXISTANT DU DOMAINE DE LA SANTÉ EN PRÉPARATION POUR SA MODERNISATION.

Boby LAFORET

ABSTRACT

Several companies for their internal needs are developing software that can help them meet those needs. Some software developed are both abandoned during development or neglected after being developed. In some cases these systems are neglected without significant documentation or even when the documents are present, they were not designed to facilitate understanding of the system. Reverse engineering is a common technique to cover, on the one hand the system design but also all useful knowledge to help its maintenance and evolution.

Nowadays, computer technology becomes the main pillar on which researchers rely in the medical community. Having a software system that can help in the experiment; in the analysis and understanding of the results is a must, especially in the genomics field. The major challenge is to help them to cover the design and all the useful knowledge that can afford to modernize genomic research system abandoned by the development team. This project aims first to present a literature review on the field of software reverse engineering while addressing the topic on software reengineering. This review discusses some aspect of the history; legacy systems, the scope of certain definitions, models and associated risks. Second time, the review is preparatory to the implementation of the reverse engineering to extract knowledge of the system. This is an approach that confirms that the maintenance and development of a system requires the support of reverse engineering.

2.4.4.5	Identification de la conception du système	35
2.4.4.5.1	Architecture du sous-système <i>Prognomix-DSS</i>	36
2.4.4.5.2	Traçabilité et reproduction des expériences	36
2.4.4.5.2.1	L'objectif de la conception d'un module de traçabilité.....	37
2.4.4.5.2.2	Architecture de données de la traçabilité.	37
2.4.4.5.2.3	Architecture logique du module de traçabilité	38
2.4.4.5.2.3.1	Portail Logbook « Logbook Portal ».....	39
2.4.4.5.2.3.2	Paramètres de sorties « <i>Output descriptor</i> »	41
2.4.4.5.2.3.3	Base de données des résultats « <i>DBresults</i> ».....	41
2.4.4.5.2.3.4	Évolution du module de traçabilité.	42
2.4.4.5.2.4	Architecture logicielle du module de traçabilité	43
2.4.4.5.2.4.1	Diagramme de classes du module « <i>controller</i> »	44
2.4.4.5.2.4.2	Diagramme des classes du module « lib »	45
2.4.4.5.2.4.3	Diagramme de classes pour le module test	46
2.5	Techniques et outils utilisés	47
2.5.1	Rétro-ingénierie selon une vision statique	47
2.5.2	Rétro-ingénierie selon une vision dynamique.....	48
2.5.3	Technique de rétro-ingénierie de base de données	48
2.5.4	Technique de rétro-ingénierie de code source	48
2.6	État d'avancement de l'activité de rétro-ingénierie	49
2.7	L'effort déployé pour le travail de rétro-ingénierie	49
2.8	Conclusion chapitre 2.	50
CHAPITRE 3 DESCRIPTION DE L'EXPÉRIMENTATION		53
3.1	INTRODUCTION	53
3.2	Sommaire du projet.....	53
3.3	Critique de la rétro-ingénierie	54
3.3.1	Faiblesse du modèle	54
3.3.2	Force du modèle.....	54
3.3.3	Opportunité	55
3.4	Détails pour enrichir le modèle théorique de la rétro-ingénierie	55
3.5	Analyse des outils disponibles de la rétro-ingénierie existants	57
3.6	Tableau comparatif des différents outils.....	58
3.7	Analyse des résultats.....	59
3.7.1	Résultats de la première étape du projet.	59
3.7.2	Résultats de la deuxième étape du projet.....	60
3.8	Améliorations futures.....	60
3.9	Conclusion	61
BIBLIOGRAPHIE.....		63
ANNEXE I INVENTAIRES SUR LES MATÉRIELS EXISTANTS		65
ANNEXE I		65
1	Computer 1: Linux Server	65

Environment.....	65
1.1 IP Addresses.....	65
1.2 Administrator connection.....	65
1.3 Hardware.....	65
1.4 Filesystems.....	65
1.5 Local users	65
Open ports.....	66
 ANNEXE II DOCUMENT MENTIONNANT PROGNO MIX.....	 69
	ANNEXE II
1. Document de présentation prognomix	69
 ANNEXE III DOCUMENTS IDENTIFIANTS LES CODES SOURCES.....	 73
	ANNEXE III
1. List des modules.....	73
2. Document de Configuration pour lancer le système.	74
3. Listes des répertoires contenant des codes sources sur le serveur	76
4. Listes des répertoires contenant des codes sources sur le serveur.	86
 ANNEXE IV LISTE DES BASES DE DONNÉES.....	 87
	ANNEXE IV
1. Description de la base de données prognomix_test	87
2. Description de la base de données « Experiment ».....	98
 ANNEXE V CONFIGURATION DU SYSTÈME	 101
	ANNEXE V
1- Les étapes de configurations du système.....	101
 ANNEXE VI FONCTIONNALITÉS DU SYSTÈME.....	 105
1- L'écran principal après le lancement du système.	105
3.10 1- LogBook	105
3.11 2- Administrations	107
3.12 3- Define Parameters.....	107
3.13 4- Search Experiments	107
3.14 5- Save Experiments	109
3.15 6- Associations.....	109
3.16 7- Machine Learning	110
3.17 8- PCA	110
3.18 9- Descriptive Stats	111
3.19 10- SNP Comparison	112
 Snp comparison analysis help.....	 113

Purpose:.....113
Usage: 113
 3.20 11- Descriptors..... 113

LISTE DES TABLEAUX

	Page
Tableau 1: Les risques potentiels de la réingénierie logicielle (Rosenberg et al., 1996).....	21
Tableau 2: Méthodologie --- Liste des tâches.....	29
Tableau 3: L'effort déployé pour le travail de rétro-ingénierie	49
Tableau 4: Tableau comparatif des outils par rapport au projet	58

LISTE DES FIGURES

	Page
Figure 1:Niveaux d'abstractions (Byrne, 1992)	12
Figure 2:Modèle de Réingénierie générale, traduit et interprété (Byrne, 1992).....	14
Figure 3: Vue haut-niveau du flux d'activité de la rétro-ingénierie (Telea, 2012)	16
Figure 4:Procédure de la rétro-ingénierie (Byrne, 1991).....	19
Figure 5: Vue organisationnelle du sous-système « <i>Prognomix-DSS</i> »	33
Figure 6: Vue organisationnelle du sous-système « <i>Experiment Manager</i> »	34
Figure 7: Vue haut-niveau de l'architecture du sous-système « <i>Prognomix-DSS</i> ».....	36
Figure 8: Architecture de données du module de traçabilité.	37
Figure 9: Architecture logique de la traçabilité	38
Figure 10: <i>Logbook</i> l'outil de traçabilité des expériences.....	39
Figure 11: Logbook, détails de l'expérience	40
Figure 12: Traçabilité par la comparaison de deux expérimentations	40
Figure 13: Paramètres de sorties de l'expérimentation	41
Figure 14: Liste des modules et les classes composantes.	43
Figure 15: Diagramme des classes du module <i>Controller</i>	44
Figure 16: Diagramme des classes du module « lib »	45
Figure 17: Diagramme de classes du module « tests ».....	46
Figure 18:Techniques de rétro-ingénierie se basant sur le modèle de la rétro-ingénierie.	47
Figure 19: Proposition d'enrichissement au modèle théorique de référence de la rétro-ingénierie.....	56

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

Biogenix-DSS: Biogenix Discovery System Support

Cobol, PL/1, LISP, RPG : Langages de programmation patrimoniaux.

AS/400 : Serveur d'application patrimonial, développé par IBM

Python : Langage de programmation orienté objet

Biogenix-DSS : Le nom du logiciel sous étude de la compagnie Biogenix Inc.

Biogenix Inc. : La compagnie qui a développé le logiciel Biogenix-DSS..

Prognomix Inc. : L'entreprise qui a subventionné la création de Biogénix-DSS.

Prognomix-DSS : Une version personnalisée de Biogénix-DSS pour Prognomix Inc.

Pylons : Cadriciel permettant de développer des applications web avec Python.

Experiment Manager : Dénomination du système permettant de monitorer, de tracer et de reproduire les expériences.

HTTP : Hyper-Text Transfer Protocol, le protocole utilisé pour présenter des documents sur le web.

HTML, CSS, JavaScript : Langage de présentation, de configuration et de manipulation des contenus web respectivement.

IP : Adresse unique utilise par chaque nœud d'un réseau informatique

TCP : Protocole utilisé dans la couche de transport du réseau.

LISTE DES SYMBOLES ET UNITÉS DE MESURE

P/M : symbole utilisé pour mesurer l'effort déployé pour faire un travail. C'est le nombre d'heures consacrées par mois, par une personne, sur une tâche.

INTRODUCTION

En voulant définir la réingénierie logicielle ou rétro-ingénierie logicielle, il s'avère évident de remonter le temps pour comprendre ce concept et son utilisation dans l'industrie informatique de développement logiciel. Les entreprises belges, durant la période de la révolution industrielle, ont procédé à la réingénierie de machineries fabriquées en Angleterre. L'objectif d'une telle activité c'était de fabriquer de machineries plus performantes que les prédécesseurs. Cet événement est considéré comme un des premiers cas d'utilisation pertinents de l'utilisation des techniques de la réingénierie du 19^e siècle (Capron et al., 2000). Cette pratique au cours du 19^e, fait de la Belgique la deuxième puissance industrielle d'Europe, derrière l'Angleterre.

En 1778, un autre cas, celui de Samuel Slater a été documenté. Il travaillait dans une usine de coton, en Angleterre (Refinefocus, 2015). Après avoir passé vingt et une années à travailler avec les machineries de l'époque, il devient expert et l'unique personne qui maîtrise les connaissances de la technologie utilisée par ces dernières. Il possède un modèle mental des plans de ces machines et, en 1789, s'envole vers New York où il a révolutionné, par la suite, l'industrie du textile américain.

En plus du domaine de la mécanique, la réingénierie devient un concept très répandu dans tous les autres domaines, par exemple les circuits électroniques et les circuits intégrés, etc. Au tout début de l'ère des ordinateurs personnels et des logiciels, la réingénierie a aussi été utilisée par plusieurs entreprises afin de comprendre l'équipement et le logiciel développé par d'autres entreprises. Une fois que la compréhension du système des autres est acquise, ces entreprises profitent pour en extraire les connaissances de la conception de ce système et de son fonctionnement. L'objectif d'un tel acte est de concurrencer les autres entreprises et de prendre une part de ces nouveaux marchés.

L'utilisation de la rétro-ingénierie dans le domaine du logiciel contribue à des avancées technologiques révolutionnaires pour les compétiteurs des grandes entreprises technologiques

comme IBM (Gilbert-MacMillan, 1993). D'un autre point de vue, la rétro-ingénierie laisse perplexes des grandes compagnies dans l'industrie de développement de logiciels propriétaires. Cette perplexité vient du fait que si leur compétiteur utilise cette technique, il pourra analyser les produits logiciels de ces grandes compagnies et d'en sortir un autre plus performant (Chikofsky et al., 1990), (Gilbert-MacMillan, 1993). Dans les années 70 à 80 et plus près de nous, en 1990, certains états des États-Unis ont même interdit formellement la pratique de la rétro-ingénierie effectuée par les ingénieurs et les entreprises afin d'empêcher le clonage des puces électroniques « *chips* » de l'industrie des semi-conducteurs. Ils ont aussi tenté d'empêcher la décompilation des logiciels (Samuelson et al., 2001).

Ce projet de recherche appliquée de 15 crédits consiste à recourir aux techniques de rétro-ingénierie du logiciel, afin de comprendre et de documenter un logiciel existant, du domaine de la recherche médicale (c.-à-d. le logiciel DSS de la firme Biogénix). Une autre équipe du laboratoire de recherche désire le moderniser. L'analyse et la compréhension du logiciel existant sont les premières étapes de ce projet. Par la suite, la documentation de l'architecture existante et des artefacts de ce système seront effectués.

Ce rapport est divisé en trois chapitres. Le premier chapitre présente la revue de littérature du domaine de la rétro-ingénierie du logiciel, à savoir : l'historique du domaine, la terminologie, les différentes techniques visant l'extraction des connaissances et la documentation de l'architecture et des artefacts existants, les difficultés et les risques de ce genre de projet. Le deuxième chapitre se consacre à la planification et la mise en application des différentes activités du plan de rétro-ingénierie visant l'analyse et la compréhension du logiciel « *Biogenix-DSS* ». Finalement, le troisième chapitre présente un sommaire de l'expérience d'utilisation de la rétro-ingénierie; une critique de la rétro-ingénierie en présentant sa faiblesse, sa force et l'opportunité qu'une entreprise peut en tirer d'une telle technique; une version améliorée du modèle rétro-ingénierie adopté pour le projet; l'analyse de certains outils de rétro-ingénierie tout en présentant un tableau comparatif de ces derniers; finalement les résultats de cette initiation à la rétro-ingénierie et une démonstration d'une amélioration

future de la part des intervenants qui auront à utiliser ces résultats de recherche pour la modernisation du logiciel.

CHAPITRE 1

REVUE DE LA LITTÉRATURE

1.1 INTRODUCTION

Ce premier chapitre présente la synthèse de l'état de l'art du domaine de la réingénierie du logiciel. Plus particulièrement l'activité de rétro-ingénierie. Cette synthèse est effectuée à partir d'une perspective de novice dans ce domaine. L'objectif de ce chapitre est d'aider le lecteur à s'initier au domaine et de comprendre comment mettre en application les concepts théoriques présents dans la littérature. D'un autre point de vue, il permettra aussi d'aider les professionnels du domaine du logiciel à s'initier aux stratégies de réingénierie du logiciel qui peuvent être adoptées pour rénover les logiciels de leur entreprise. Un court historique du domaine est présenté suivi d'une présentation des concepts de la réingénierie du logiciel telle que vécue dans l'industrie du logiciel. Les principales définitions sont présentées suivies d'un modèle conceptuel regroupant l'ensemble des activités de la réingénierie logicielle. On y aborde aussi les processus et procédures applicables à une activité de rétro-ingénierie. Finalement, une courte discussion abordera les aspects de risques d'une activité de rétro-ingénierie du logiciel. Cette synthèse permettra d'acquérir des connaissances utiles, structurées et orientées-résultats utiles pour le cas d'étude.

1.2 Historique

Il est difficile de préciser la date exacte de l'apparition et de la première publication du concept de réingénierie. Toutefois, il est publié (Capron et al., 2000) que le recours à la réingénierie des machineries de l'Angleterre a permis à la Belgique de devenir une grande puissance industrielle. Dans les années 1780, l'industrie textile américaine s'est révolutionnée par le biais du technicien qui travaillait avec les machines qui tissent les cotons, en exportant ses connaissances acquises dans l'industrie vers les industries américaines (White, 2013).

Plus récemment, au 20^e siècle, la réingénierie a été utilisée par « *General Motor Corporation* » pour concurrencer les produits de « *Ford Motor Company* » et vice-versa (Dhillon, 2002). Cette technique devient très répandue chez les militaires leur permettant ainsi de fabriquer des armes plus puissantes en étudiant ceux de leurs adversaires (Dhillon, 2002).

Selon certains articles et ouvrages consultés, le concept réingénierie a aussi été utilisé dans le domaine industriel, plus particulièrement celui de la mécanique (Friesinger et al., 2014). Dans ce domaine il a permis de décomposer et analyser des pièces mécaniques avec l'objectif de comprendre leur conception et de les surpasser. L'arrivée des premiers ordinateurs a suscité l'intérêt d'appliquer ce même concept à la réingénierie des matériels informatiques de l'époque. Le logiciel n'échappe pas à cette tendance. Dès 1990, l'influence de ce concept dans l'industrie de développement logiciel se remarque avec la publication de plus de 480 articles de recherche reliés directement à la réingénierie logicielle (Dhillon, 2002).

La réingénierie vise à moderniser les logiciels vieillissants. Ces systèmes dits patrimoniaux sont ceux qui sont visés par les activités de réingénierie avec l'objectif d'extraire des connaissances existantes concernant les règles d'affaires et les données et pour reconstruire des logiciels plus modernes.

L'histoire retient qu'en 1991 Linus Torvalds a développé le premier noyau de Linux. Ce noyau a été développé parce que M. Torvalds jugeait que tout le monde devrait avoir une version d'Unix sur son ordinateur. Bien qu'il étudia en Finlande à l'époque, il a pu solliciter l'aide d'internautes pour construire son logiciel libre en prenant comme cible le système *UNIX* existant qui était un produit propriétaire et coûtait très cher. Cette technique de réingénierie, utilisée par M. Torvalds, a révolutionné le domaine du génie logiciel et a donné, à un certain nombre de programmeurs, l'opportunité d'avoir un système qu'ils peuvent modifier à leur tour librement (Garrels, 2007).

La prochaine section introduit les notions de systèmes patrimoniaux, de la réingénierie et de la rétro-ingénierie logicielle en précisant les concepts et leurs applications dans le domaine du logiciel.

1.2.1 Les systèmes patrimoniaux

1.2.1.1 Définitions

On définit habituellement comme patrimonial (de l'anglais « *legacy system* ») « Tout système informatique hérité des années passées (en pratique 1970 à 1995), généralement développé sur mesure (« *in-house* ») pour supporter les fonctions importantes, sinon essentielles au fonctionnement des entreprises (Ulrich, 2002) ».

Ce sont généralement des systèmes (composés de matériel et/ou de logiciel) qui sont appelés à cohabiter avec d'autres systèmes, souvent plus modernes, mais leurs changements constituent un risque important et un coût élevé. Ces systèmes sont en général issus des technologies des ordinateurs centraux ou « mainframe » et ont été pour la plupart, développés entre les années 1970 et 1995.

Les entreprises manufacturières, les sociétés d'assurances, les banques et certaines grandes entreprises utilisent des logiciels écrits en langage *Cobol* ou en langage *Fortran*. Certaines autres entreprises des systèmes d'information datant de plusieurs décennies, par exemple le système *AS/400* dont le coût de la maintenance est de plus en plus élevé.

La présence de ces systèmes au sein des entreprises pose plusieurs défis aux gestionnaires, car les ressources humaines, en mesure de faire évoluer ces systèmes, sont de plus en plus rares.

1.2.1.2 Caractéristiques des systèmes patrimoniaux

Les systèmes patrimoniaux sont des systèmes qui, en général :

- utilisent une technologie de grande puissance, centralisée (« *Mainframe* »);

- Possèdent un système de gestions transactionnelles qui s'opèrent entre l'ordinateur central et les divers terminaux qui sont souvent d'une architecture distribuée.
- Utilisent souvent les langages de programmation de 2e et de 3e génération, tels *Cobols*, *PL/1*, *LISP*, *RPG* dans le cas d'*AS/400* ou Assembleur pour les moins répandus.
- Sont constitués d'un système de gestion de donnée (c.-à-d. gestion de base de données) non évolué, mais qui sont puissant et performant.
- Ont une taille élevée en ce qui concerne le nombre de « lignes de code » et du nombre modules/programmes.
- Sont souvent les résultats d'un produit développé en interne par l'entreprise. Donc personnalisés pour des taches spécifiques et « sur mesure ».
- Sont un ensemble technologique à savoir: plateforme, système d'exploitation, système de transactions, langages de programmation , système de gestion de données, matériel intégré, qui sont arrivés au point culminant du cycle de vie dont le support et la maintenance est maintenant un défi majeur.
- Sont souvent des systèmes dont la documentation est inexistante, incomplète ou absente. D'où la nécessité d'avoir recours à des personnes expertes dans le domaine pour assurer le support et la maintenance, en d'autres termes de garder en vie le système.

1.2.1.3 Les principales problématiques des systèmes patrimoniaux.

Les systèmes patrimoniaux restent et demeurent un mal nécessaire dans les entreprises. Les risques associés à leur présence soulèvent certaines problématiques dans les entreprises dont les principales sont:

- La désuétude technologique: comment assurer la survie du système si celui-ci n'est plus supporté par les fournisseurs?
- La rareté de compétences: comment assurer la maintenance et l'adaptation du système si le personnel compétent ou expert dans le domaine est de plus en plus rare (l'âge de retraite, promotions, départs)?

- Comment faire évoluer un système qui ne répond plus à notre besoin sachant qu'il existe uniquement les codes sources?

En tenant compte que, les modules développés dans ces genres de systèmes n'ont pas suivi un processus moderne d'ingénierie logicielle, leur qualité est faible et la traçabilité entre les composants est difficile.

1.3 Définitions du domaine

Maintenant que la problématique a été présentée, plusieurs définitions existent et sont tirées d'articles et livres tels: l'article "*Reverse Engineering and Design Recovery*" de (Chikofsky et al.,1990); le livre "*Engineering and Technology Management Tools and Applications*" de (Dhillon, 2002); un extrait du livre "*The Art of Reverse Engineering*" de (Friesinger et al.,2014); et finalement de l'article "*Reverse Engineering of Computer Software and U.S Antitrust Law*" de (Lande et al.,1996). Ces définitions reflètent les différentes perspectives de la réingénierie du logiciel :

- **Ingénierie ou ingénierie directe (*forward engineering*):** est le processus conventionnel de passer d'une abstraction de haut niveau et de conception détaillée vers le développement physique du système. (Dhilon, 2002)
- **Réingénierie (*reengineering*):** c'est le processus d'analyse et d'altération d'un système existant en vue de le reconstituer et l'implémenter sous une nouvelle forme. Généralement, le processus de réingénierie inclut des activités de rétro-ingénierie en vue d'obtenir une description (à un niveau supérieur d'abstraction) qui appuie le processus d'ingénierie (*forward engineering*) ou de restructuration (*restructuring*). (Chikofsky et al., 1990)
- **Re-structuration (*restructuring or refactoring*):** Consiste en une transformation d'un logiciel en un logiciel plus structuré sans altérer son comportement externe, c'est-à-dire sa fonctionnalité.

- **Re-documentation (*redocumentation*):** est la plus ancienne et la plus populaire des techniques de rétro-ingénierie. Elle consiste en la création de représentations sémantiquement équivalentes au niveau d'abstraction étudiée. Cela vise également de produire de la documentation d'un logiciel existant pour créer ou de mettre à jour la documentation actuelle. (Chikofsky et al., 1990)
- **Rétro-ingénierie (*reverse engineering*):** est un concept qui tire son origine du domaine de la mécanique. La rétro-ingénierie indique le processus d'analyse des objets ou systèmes existants en étudiant son plan de construction couche par couche afin de le reconstruire dans les moindres détails. Cette manière de reconstruction permet d'apporter des modifications et des ajustements aux nouvelles demandes et exigences, il signifie l'appropriation créative, la démocratisation de la connaissance, pour un développement plus approfondi. (Friesinger et al., 2014)

La rétro-ingénierie en elle-même, n'implique pas le changement du système ou de l'objet étudié ni de créer un nouveau système basé sur la réingénierie du système étudié. Il est un processus d'examen, mais non pas un processus de réplique. (Chikofsky et al., 1990)

- **Rétro-ingénierie logicielle (*software reverse engineering*):** Le concept de rétro-ingénierie appliquée dans le domaine logiciel se réfère à un ensemble de processus entrepris pour comprendre comment un programme logiciel est construit et comment il exécute ses fonctionnalités. (Lande et al., 1996)
- **Recouvrement de la conception (*design recovery*):** consiste à créer la conception de l'abstraction à partir de certains éléments (code, documentation existante de la conception si disponible, expériences personnelles, connaissances générales dans le domaine du problème). Le recouvrement de la conception est sensé de permettre à

reproduire toutes les informations pouvant aider à toute personne de comprendre ce que fait un système, la raison du système, etc. (Chikofsky et al., 1990)

- **Re-systémisation (*resystemization*):** est le processus de la rétro-ingénierie d'un élément impliquant l'éradication de la conception de l'environnement associé et les restrictions de développement en suivant l'ingénierie directe pour la mise en œuvre d'un nouvel environnement sans changer les fonctionnalités. (Dhilon, 2002)

1.4 Modèle de la réingénierie logicielle.

Il y a plusieurs décennies, les chercheurs du domaine du développement de logiciels ont effectué de la recherche principalement sur le développement du logiciel afin de résoudre les problèmes auxquels ils faisaient face quotidiennement. Ces chercheurs ont négligé les aspects de la maintenance et de l'évolution du système pendant plusieurs années (Müller et al., 1993). Ces pionniers du domaine du logiciel n'ont pas su anticiper que les logiciels développés dans les 1960 et 1970 survivraient 25 ans plus tard et auraient besoin d'être modifiés pour refléter les besoins changeants du marché. Ce sont ces systèmes qu'on reconnaît aujourd'hui sur le nom de « *legacy system* », qui ne peuvent pas être remplacés sans tirer profit de toute leur histoire. Ils contiennent des connaissances substantielles tapies dans les règles d'affaires de l'entreprise, à savoir : les exigences; les décisions de conceptions; les règles d'affaires qui ont évolué et ont été maintenues au fil des années et qui ne se retrouvent nulle part ailleurs (Müller et al., 1993). Dans la majorité des cas, ce sont des systèmes logiciels dont la qualité interne et externe n'est pas reluisante. Continuer à les exploiter dans leurs formes actuelles n'est pas satisfaisant pour l'entreprise. Il se peut même que la plateforme sur laquelle ils opèrent ne soit plus supportée par le fabricant. Ce n'est pas toujours facile d'acquérir un nouveau système ou d'en développer un nouveau pouvant le remplacer. Un tel travail nécessiterait des techniques d'extraction des connaissances existantes comme la réingénierie et la rétro-ingénierie.

Ce projet de maitrise se concentrera sur la rétro-ingénierie d'un logiciel semblable, le logiciel « *Biogénix-DSS* ».

La réingénierie consiste d'une part, la rétro-ingénierie « *reverse engineering* » du système existant pour extraire la compréhension et les connaissances existantes en arrière-plan de ce système. D'autre part, il utilise le domaine de l'ingénierie directe « *forward engineering* » pour altérer le mécanisme interne du système jusqu'à ajouter de nouvelles fonctionnalités ou pour en créer un tout nouveau système capable de répondre à des besoins futurs (Chikofsky et al., 1990). D'après Byrne (Byrne, 1992), la réingénierie peut être appliquée pour la refonte totale d'un système ou pour la refonte d'une portion de ce dernier. Toutefois, l'application de la réingénierie nécessite la compréhension des grandes phases qui ont amené au développement d'un système logiciel dont le niveau d'abstraction est aussi utilisé tant pour l'ingénierie directe tant pour la réingénierie logicielle.

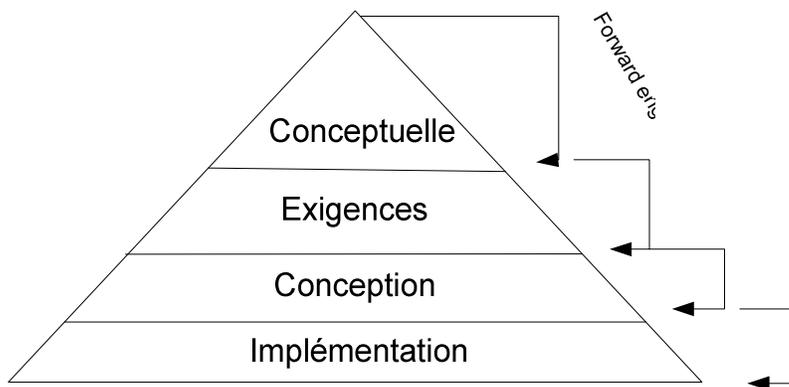


Figure 1: Niveaux d'abstractions (Byrne, 1992)

La figure ci-dessus, est inspirée du modèle présenté par Byrne (Byrne, 1992), illustre les grandes étapes que comprend typiquement un développement de logiciel. Le but de chaque niveau est de définir certaines caractéristiques qui déterminent le degré auquel un système peut faire valoir une propriété (Byrne, 1992). Cette figure est constituée de différents niveaux qui impliquent plusieurs caractéristiques du système. Le niveau « Conceptuel » est le plus

abstrait, dans lequel les caractéristiques fonctionnelles du système sont décrites d'un point de vue général ou global.

Le niveau « Exigences » décrit de manière plus détaillée les caractéristiques fonctionnelles du système. Le niveau « Conception » décrit les caractéristiques architecturales du système. Le niveau « Implémentation », le plus bas niveau de la pyramide, décrit les caractéristiques d'implémentation du système compréhension par un ordinateur. Ce modèle représente le modèle de l'ingénierie directe « *forward engineering* » duquel prendra naissance le modèle de la réingénierie.

Afin de mieux comprendre l'application ou l'utilisation concrète de la réingénierie à partir de la figure 1, un autre modèle sera présenté à la figure 2. Ce modèle est l'illustration d'une ascension et d'une descente qui peut être vue dans le contexte d'une montagne russe. Dans l'ingénierie directe « *forward engineering* », on commence toujours par le niveau le plus abstrait tandis que dans la réingénierie on aborde dans un premier temps, la rétro-ingénierie « *reverse engineering* » en commençant par le niveau le plus concret possible afin d'arriver au niveau le plus abstrait. Une fois arrivée au niveau le plus abstrait, à ce point, appliquer l'ingénierie directe « *forward engineering* » pour altérer le système ou pour en développer un autre, cela constitue la réingénierie « *re-engineering* ». Arrivé au stade de la réingénierie du nouveau système, de nouveaux plans et stratégies adoptés, doivent s'approprier. La stratégie de construire un nouveau système qui répond à des qualités attendues par le client, doit effectivement passer par l'élaboration de nouvelles techniques et de trouver les ressources appropriées afin d'atteindre l'objectif du projet.

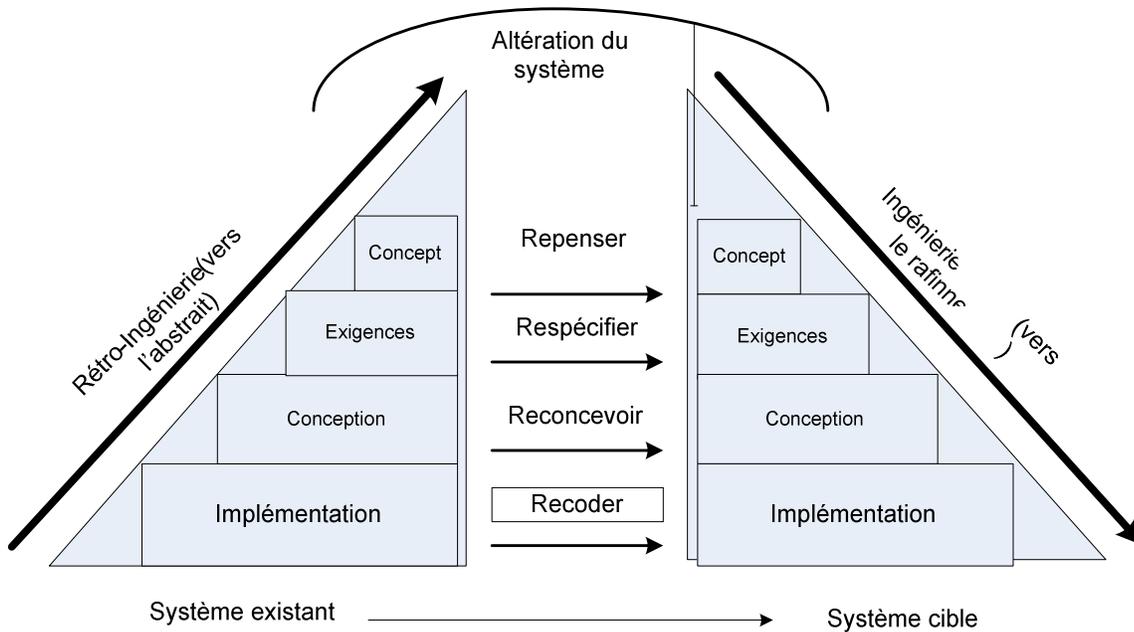


Figure 2: Modèle de Réingénierie générale, traduit et interprété (Byrne, 1992)

En regardant le sens des flèches, plusieurs cas de figure peuvent se présenter de la rétro-ingénierie vers la réingénierie. Si les concepts ne sont pas altérés ou si les exigences restent les mêmes, il sera possible à partir de la rétro-ingénierie de passer directement vers la conception et de continuer vers l'implémentation (Byrne, 1992).

1.4.1 La rétro-ingénierie

La rétro-ingénierie est un processus d'analyse d'un système afin :

- d'identifier les composants du système et leurs interrelations ;
- de créer des représentations du système dans d'autres formes ou à un plus haut niveau d'abstraction.

Généralement, la rétro-ingénierie implique l'extraction des artefacts de la conception « *design* » du système et d'en construire ou de synthétiser les abstractions, dont l'implémentation est dépendante (Chikofsky et al., 1990). Il faut noter que même lorsque la

rétro-ingénierie implique souvent un système existant et fonctionnel comme sujet, ce n'est pas une exigence que cela soit ainsi. Cela explique que la rétro-ingénierie est applicable à n'importe quel niveau d'abstraction et à n'importe quelle phase du cycle de vie du logiciel (Chikofsky et al., 1990).

La rétro-ingénierie est le processus d'extraction de savoir-faire ou de la connaissance de tout artéfact fait par l'humain (Samuelson et al., 2001). L'auteur n'approuve que la rétro-ingénierie et il précise qu'elle est la manière appropriée pour obtenir cette information, même lorsque l'intention est d'adapter le produit aux besoins du client. La rétro-ingénierie est applicable dans différents contextes, mais le contexte d'utilisation pour la compétitivité dans l'industrie de développement logiciel est celle la plus fréquente (Samuelson et al., 2001).

1.4.1.1 Objectifs de la rétro-ingénierie

En rapport à l'industrie de développement de logiciel et la nécessité de comprendre des systèmes existants, la rétro-ingénierie vise plusieurs objectifs. La rétro-ingénierie se donne pour objective:

- 1) Faciliter la réutilisabilité du système ou certaines de ses fonctionnalités.
- 2) Retrouver les informations perdues liées aux règles d'affaires, les besoins, la conception, etc.
- 3) Migrer d'une plateforme matérielle ou logicielle vers une autre
- 4) De fournir des documents manquants ou alternatifs du système existant.
- 5) D'assister à la maintenance et l'évolution du système existant.

L'un des buts principaux, du travail de rétro-ingénierie du système « *Biogenix-DSS* », est de pouvoir mettre en application les différents points de l'objectif du domaine, présenté ci-dessus.

1.4.1.2 Modèle de la rétro-ingénierie

Toute activité de rétro-ingénierie nécessite l'utilisation ou l'application d'un modèle comme patron de processus. Considérant des étapes de base de l'activité du domaine, le processus de la rétro-ingénierie peut-être décomposé en un sous-ensemble comportant trois tâches : 1) l'extraction; 2) l'analyse; et 3) la visualisation (voir la figure 3). À chaque étape du processus, il faut répéter chacune de ces tâches afin de trouver le résultat escompté.

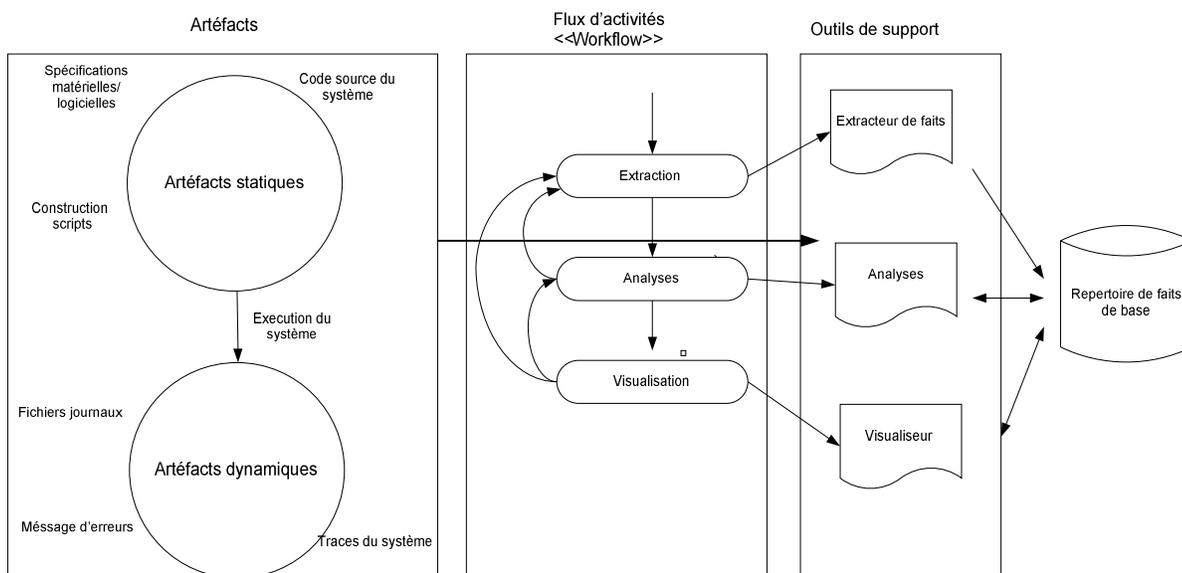


Figure 3: Vue haut-niveau du flux d'activité de la rétro-ingénierie (Telea, 2012)

Pour chaque tâche dans la figure 3, « flux activité », des outils sont associés afin d'aider à sa réalisation. La figure se trouvant complètement de droite de la figure 3, sert à stocker les informations après un examen minutieux. Ces informations sont structurées sous forme de modèle de données, de schéma de données ou métadonnées. La figure complètement de gauche démontre qu'en extrayant les informations des systèmes, il sera possible de distinguer les vues statiques de celles qui sont dynamiques. Les informations statiques (c.-à-d. code source, spécifications du système, scripts) peuvent être obtenues sans avoir à exécuter le système. Alors que les informations dynamiques (c.-à-d. fichiers, journaux, message d'erreurs, traces) sont collectées à partir de l'exécution du système.

Il est souvent souhaitable d'avoir à la fois des informations statiques et dynamiques disponibles, car elles donnent une image plus complète du système cible (Telea, 2012).

1.4.1.3 Procédure de la rétro-ingénierie

La mise en application de la rétro-ingénierie, dans un contexte réel, nécessite une marche à suivre. La première étape d'une telle démarche vise à extraire des informations détaillées de la conception à partir du plus haut niveau d'abstraction de la conception. Le plus bas niveau de détails, de la conception, peut être récupéré à partir du code source existant ou des documents de conception existants. Pour procéder à la mise en application de la rétro-ingénierie, ci-dessous les activités proposées :

- a) **Collecter les informations** : à cette étape, il faut collecter toutes les informations possibles concernant le produit logiciel. Les sources potentielles des informations sont : les codes sources; documents de conceptions; documentations de l'appel système et les méthodes externes; les expériences personnelles avec un logiciel similaire sont à considérer.
- b) **Analyser les informations collectées** : c'est incontournable de réviser les informations collectées. Cette activité permettra de se familiariser avec le système et ses composants. À ce stade, définir un plan pour comprendre les informations enregistrées est nécessaire.
- c) **Extraire la structure** : identifier la structure et l'utiliser pour créer un ensemble hiérarchique de tableaux de structure. Chaque nœud de la hiérarchie de structure correspondant à une méthode appelée dans le programme. Le tableau de structure enregistre la hiérarchie d'appels du programme. Les données transmises et retournées d'un nœud à un autre doivent être enregistrées.
- d) **Enregistrer les fonctionnalités** : pour chaque nœud du tableau de structure, enregistrer le traitement effectué dans la méthode du programme correspondant à ce

nœud. Un langage de description de page (par exemple le PDL) peut être utilisé pour décrire les fonctionnalités et méthodes du programme.

- e) **Enregistrer les flux de données** : pour construire un ensemble hiérarchique de diagrammes de flux de données qui modélise le logiciel, il faut analyser la structure et le langage description de page recouvert afin d'identifier les transformations des données dans le logiciel.
- f) **Enregistrer les flux de contrôles** : cette étape consiste en une identification de haut niveau de la structure du programme de contrôle puis de son enregistrement en utilisant des diagrammes de flux de contrôle. Elle se réfère à un contrôle de haut niveau qui affecte le fonctionnement d'ensemble du logiciel. Toute fois, elle ne se réfère pas au contrôle de bas niveau des traitements.
- g) **Revoir le recouvrement de conception** : cette étape consiste en une révision de la conception récupérée afin d'assurer la cohérence et l'exactitude avec des informations disponibles. Il est évident également d'identifier tous les éléments manquants de l'information afin de tenter de les localiser. Il faut revoir la conception pour vérifier qu'il représente correctement le programme.

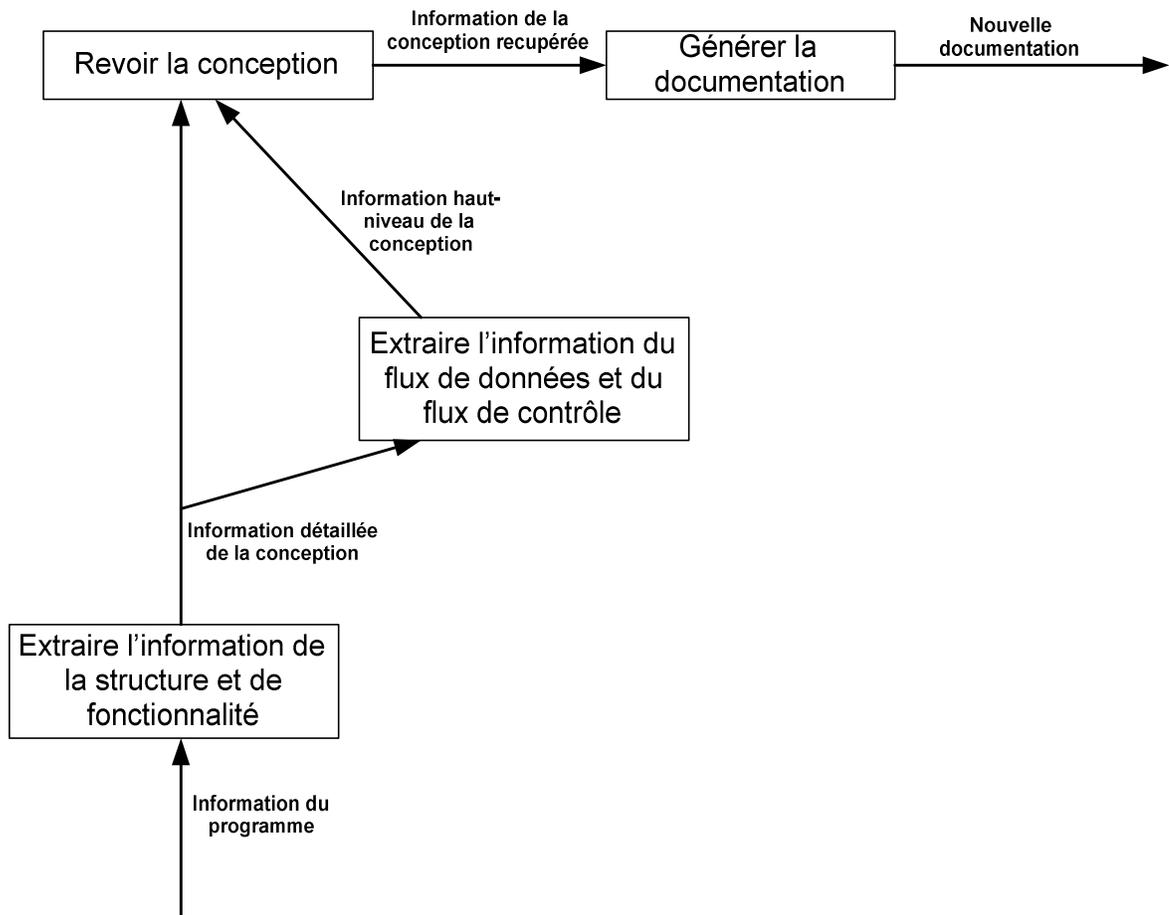


Figure 4: Procédure de la rétro-ingénierie (Byrne, 1991)

1.4.1.4 Documentation de la rétro-ingénierie

Au final, la rétro-ingénierie est une activité qui est menée pour comprendre un système existant. Cette activité implique plusieurs parties prenantes à qui l'état d'avancement ou les résultats de l'activité devraient être communiqués. L'adoption d'un modèle de communication est importante, mais également la documentation de la rétro-ingénierie est un impératif à ne pas négliger. La dernière étape de la procédure de la rétro-ingénierie exige qu'une nouvelle documentation de la conception du système soit générée. Cette documentation aura à expliquer le but du programme, un aperçu du programme, son histoire, etc. Probablement, les codes sources ne contiennent pas ses informations, il va falloir en consulter d'autres sources (Byrne, 1991).

1.4.1.5 Autres types de rétro-ingénierie

Dans le contexte spécifique de ce projet, en dehors de la rétro-ingénierie de manière générale, la rétro-ingénierie de code source « *code reverse engineering* » et la rétro-ingénierie des données « *data reverse engineering* » vont être utilisées pour extraire des connaissances sur l'architecture logicielle (*Biogenix-DSS*) et également pour extraire les schémas relationnels constituant le système informationnel de ce dernier (Müller et al., 2000).

1.4.1.6 Outils de la rétro-ingénierie

Dans le domaine de l'industrie et de celui de la recherche académique, un large éventail d'outils de rétro-ingénierie et de récupération de conception « *design recovery* » a été développé. Certains outils sont utiles pour la rétro-ingénierie dans une vision statique et d'autres pour la rétro-ingénierie dans une vision dynamique (Systä et al., 2000). Une brève description et un contexte d'utilisation de certains outils seront présentés dans la section suivante.

- **RIGI environnement de rétro-ingénierie** : utilise les graphes pour voir les artéfacts logiciels et leurs relations. Il soutient l'extraction et la récupération de l'information de la conception en dehors du système existant.
- **Éclipse Dali pour java** : est un atelier d'architecture pour l'extraction, la manipulation et les tests de conformité. Il intègre plusieurs outils d'analyse et enregistre les informations extraites dans un dépôt.
- **Imagix4D pour C, C++, java**: Permet de comprendre, de documenter et améliorer des sources codes patrimoniaux. Offre une vue 3D pour visualiser les facettes du logiciel
- **Instant reverse Python (Visual Paradigm)** : est un processus pour produire le diagramme des classes en prenant en entrée le code source. Cette fonctionnalité est intégrée dans l'application Visual Paradigm.
- **Epydoc** : c'est un logiciel qui permet la génération de la documentation de l'API pour les modules Python en fonction de leurs docstrings.
- **PyNSource** : Outil UML pour extraire le modèle à partir des codes sources pythons.

- **AnySQL Maestro** : Outil de rétro-ingénierie de base de données permettant de produire le Diagramme du schéma relationnel pour les bases de données PostgreSQL.
- **MS Visio 2007, version professionnelle**: est un logiciel qui est utilisé pour la rétro-ingénierie des bases de données, car il possède une fonctionnalité dédiée à cet effet.

1.5 Les risques liés à la réingénierie logicielle

Malgré les efforts de la réingénierie pour atténuer les risques, réduire les coûts de fonctionnement et maintenir le système existant, il reste un domaine à risque. Être capable d'identifier les risques très tôt dans le processus pourrait aider à être plus efficace. Ce tableau ci-dessous définit les différents risques susceptibles d'être gérés au cours de la réingénierie.

Tableau 1: Les risques potentiels de la réingénierie logicielle (Rosenberg et al., 1996)

Zone de risque	Risques
Processus	<ul style="list-style-type: none"> - Les coûts extrêmement élevés des manuels de réingénierie - Avantages de coûts qui ne se réalisent pas dans le cadre de temps nécessaire - Dérives de l'effort de réingénierie - Le manque d'engagement de la direction à la solution de réingénierie continue - Sous-systèmes pour l'approche choisie de manière incorrecte - Gestion de configuration inadéquate - Le manque d'assurance de la qualité - Le manque de programme de mesures - Réingénierie sans aucun expert en application locale disponible - Système de réingénierie ne fonctionnant pas correctement - Masse de documentation dont le coût de la production est trop élevé - La technologie de réingénierie logicielle est insuffisante pour atteindre les objectifs du domaine - La fonctionnalité patrimoniale devient obsolète avant que la réingénierie s'achève
Rétro-ingénierie	<ul style="list-style-type: none"> - Le langage utilisé n'est pas conçu pour exprimer les informations abstraites nécessaires pour les exigences et la conception. - Difficulté de capture plus de conception et d'exigences à partir du code source - L'Objet capturé est incomplet ou incorrect

	- Les connaissances des règles d'affaires existantes sont perdues dans le code source -recouvert des informations non utiles et non utilisées.
Ingénierie	- Nouvelles exigences et fonctionnalités sont ajoutées. - Capturé des objets qui ne seront pas intégrés dans le nouveau système - Difficulté dans la migration des données existante - Le niveau de préparation et de rétro-ingénierie est insuffisant
Personnel	- Les personnels n'ont pas le niveau de compétences requises et n'ont pas assez d'expériences dans la réingénierie.
Outils	-Dépendance à des outils qui ne fonctionnent pas comme prévu -des outils disponibles, mais nouveaux et immatures dont certaines fonctionnalités sont incomplètes. - Maturité - stabilité du fournisseur de l'outil et de la qualité de l'outil
Stratégie	-Engagement prématuré d'une réingénierie pour un système au complet. -À défaut d'avoir une vision à long terme avec des objectifs intermédiaires - Objectifs irréalistes - l'approche choisie ne répond pas aux objectifs de l'entreprise, le budget ou Calendrier - Pas de plan d'utilisation des outils de réingénierie

1.6 Les avantages de la réingénierie logicielle.

Les principaux avantages de la réingénierie logicielle sont les suivants (Abbas et al., 2012).

- **Augmenter la maintenabilité du logiciel** : cet avantage permet d'avoir un système facile à modifier et corriger les erreurs, améliorer la performance et certains autres attributs. Faciliter l'adaptation du système dans un environnement changeant.
- **Améliorer la performance du logiciel** : Cet avantage permet d'économiser d'argent, d'augmenter la productivité, de diminuer le coût de matériel et de développement, garder une bonne relation avec la clientèle.
- **Augmenter l'interopérabilité du logiciel** : facilité la communication entre les différents systèmes existants.
- **Diminuer la dépendance du système au personnel** : réduire la coordination entre plusieurs équipements de développement géographiquement éloignés ou distancés aiderait à être plus efficace.

- **Améliorer la testabilité** : réduire la complexité du système et de ses composants permet de gagner en terme de coûts de test et de temps de test.

1.7 Brève perspective

Dans les années 70, les systèmes ont été développés dans un langage moins évolué que ceux utilisés depuis plusieurs décennies. Les systèmes patrimoniaux développés dans le temps et existent encore aujourd'hui comme *AS/400* et les autres ont été développés en *COBOL*, *Fortran*, *RPG*, etc. De nos jours, la majorité des entreprises tente de se migrer vers la technologie web ou le « *cloud computing* ». Diverses technologies web utilisées existent actuellement et dans les prochaines années à venir elles seront considérées comme de vieilles technologies. Ceux qui auront à les utiliser encore devraient recourir à la technique de réingénierie ou la rétro-ingénierie en fonction du besoin. Être entre mesure d'appliquer la technique de rétro-ingénierie pour comprendre et permettre l'amélioration de ces applications nécessiterait beaucoup de compétences et de maîtrise de plusieurs technologies, outils et techniques adaptées.

Certaines entreprises utilisent encore des systèmes patrimoniaux. Parallèlement, elles font développer des applications web pour se migrer lentement vers les technologies en vogue actuellement. Dans deux décennies à venir, la charge de maintenance de leur système deviendrait lourde. Les groupes actuels de développements, s'ils n'anticipent pas et n'appliquent pas avec rigueur les bonnes pratiques de l'ingénierie logicielle dans les deux prochaines décennies, les mêmes crises de compétences pourraient se répéter.

1.8 Conclusion chapitre 1

Le premier chapitre a effectué une revue de la littérature du domaine de la réingénierie et de la rétro-ingénierie logicielle. Nous avons vu que la rétro-ingénierie comporte plusieurs techniques qui aident lors de la réingénierie d'un logiciel. Le choix des techniques doit être adapté aux situations particulières. Le chapitre 2 décrit la mise en application de certaines techniques de la rétro-ingénierie pour le système, sous étude, « *Biogenix-DSS* » afin d'en

extraire les connaissances utiles pour son amélioration et son évolution future. Certains livrables comme : l'architecture du système, l'architecture de la base de données, le document de configuration et de déploiement du système, la librairie des composants, les diagrammes de classes et d'autres représentations seront produits.

CHAPITRE 2

RÉTRO-INGÉNIERIE APPLIQUÉE À UNE ÉTUDE DE CAS

2.1 INTRODUCTION

La revue littéraire de la rétro-ingénierie effectuée dans le chapitre précédent a été utile pour identifier des concepts, approches et techniques nécessaires dans un contexte réel. Dans le deuxième chapitre, des techniques de rétro-ingénierie seront utilisées pour extraire et documenter certaines connaissances perdues ou non existantes d'un système existant. L'objectif est d'obtenir des connaissances concernant son architecture, sa configuration, la hiérarchie d'objets qu'il manipule de manière à préparer sa modernisation. D'une manière plus spécifique, la documentation du sous-système de traçabilité des expériences adoptée par ce système fera l'objet de ce projet de recherche. Cette information sera utile dans le cadre d'un projet de « *PhD* » pour aider le chercheur à comprendre la technique détaillée utilisée afin de reproduire une expérience dans le domaine de la santé.

2.2 Contexte

Le système « *Biogenix-DSS* », a été développé pour aider les chercheurs du domaine de la santé à effectuer et reproduire des expérimentations en laboratoire. Il a été développé par la société Biogenix Inc. Après plusieurs années de travail assidu, le financement manquant, les membres de l'équipe ont dû abandonner le projet. Ils abandonnent le projet et laissant derrière eux le logiciel et de la documentation sur plusieurs ordinateurs qui ont été mis en entrepôt. Plusieurs livrables du système se trouvent éparpillés sur ces ordinateurs sans laisser des détails pouvant permettre de savoir lesquels d'entre eux possèdent les bonnes versions ou la dernière version du système. Les documents trouvés ne sont pas clairs et ne sont pas précis de manière à facilement permettre à une autre équipe de prendre la relève du système. Une version opérationnelle partielle du logiciel est encore utilisée en production dans le laboratoire de recherche du Dr. Pavel Hamet du CHUM de Montréal.

Le gestionnaire du projet a l'intention de redonner vie au système. Son objectif est d'extraire toutes connaissances utiles du système afin de le moderniser. À noter que ce système est le résultat d'environ huit (8) années-personne de travail de développement. Les ordinateurs des membres de l'équipe développement sont rapatriés dans un laboratoire de l'École de Technologie Supérieure. Ce projet est le début de recherches minutieuses sur ces ordinateurs afin de trouver des indices du fonctionnement de ce système. Dans le cas où la dernière version du logiciel sur lequel travaillaient les équipes serait retrouvée, le premier défi à relever sera de le faire fonctionner. Le second défi sera d'appliquer des techniques de rétro-ingénierie logicielle afin d'extraire le plus grand nombre de connaissances utiles, à partir de son code source, jusqu'aux règles d'affaires si possible.

2.3 Les exigences à satisfaire

Le gestionnaire du projet aimerait que le système « *Biogenix-DSS* » soit opérationnel et fonctionnel pour en faire la démonstration aux chercheurs. Pour cela, différentes tâches sont proposées.

1. Identifier le système ainsi que sa vraie version (c.-à-d. la version la plus récente);
2. Identifier tous les documents utiles permettant de comprendre le fonctionnement du système;
3. Faire fonctionner le système;
4. Identifier le système informationnel manipulé;
5. Recouvrir l'architecture et les autres diagrammes du système;
6. Produire un rapport sur l'expérimentation.

2.4 Méthodologie appliquée

Basé sur la procédure de rétro-ingénierie présentée à la section 1.3.1.3 de ce document, le travail, en cinq étapes, à effectuer se résume ci-dessous :

2.4.1 Collecte des informations

Étape 1 : La première étape pour collecter les informations consiste à identifier les ordinateurs sur lesquels travaillaient les équipes de développement du logiciel. Voir à l'annexe 1, le rapport détaillé de cette étape;

Étape 2 : La deuxième étape consiste à identifier les documents qui mentionnent l'existence d'un éventuel système « *Prognomix-DSS* ou *Biogenix-DSS* » afin de découvrir son fonctionnement. Voir en annexe 2 tous les documents découverts lors de cette étape;

Étape 3 : Retrouver des documents qui identifient le code source à utiliser pour faire fonctionner le système. Voir à l'annexe 3, le rapport de cette étape;

Étape 4 : Trouver la base de données utilisée et les différentes tables nécessaires au fonctionnement du logiciel. Voir à l'annexe 4 pour le rapport associé à cette étape;

Étape 5 : Trouver tout indice mentionnant la fonction de traçabilité des expériences et produire la documentation nécessaire pouvant aider à en faire la réingénierie lors de la modernisation future du système.

2.4.2 Analyse des informations collectées.

Ayant colligé toutes les informations inventoriées précédemment, elles sont passées en revue afin d'identifier celles qui sont utiles et celles qui ne le sont pas. À la suite aux informations collectées sur les différents ordinateurs existants, certaines sont identifiées comme utiles et d'autres sont mises de côté. Par exemple, lors de l'étape 1, les trois ordinateurs qui sont rassemblés au laboratoire « *GÉLOG* » sur lesquels travaillaient les équipes de développement, deux d'entre eux contiennent des informations utiles pour la poursuite du projet. À l'étape 2, l'équipe travaillait sur trois systèmes différents qui utilisent presque le même système informationnel. Les trois systèmes sont libellés :

- a) *Biogenix-DSS*
 - *Prognomix-DSS*
 - *Experiment Manager*
- b) *FCI-DSS*
- c) *MDEE-DSS*
- d) *Trac*

Le système qui sera utilisé pour la poursuite du projet est « *Biogenix-DSS* » plus particulièrement le sous-système « *Experiment Manager* ». Le module « *Experiment Manager* » est le module responsable de la traçabilité des expériences. Tandis que pour le système *FCI_DSS*, il suffit de le faire fonctionner. *Trac* est le système de billetterie utilisée par l'équipe de développement, il est important de le trouver et de le faire fonctionner. Cette information a été donnée par l'ancien propriétaire de *Biogénix*. À l'étape 3, les documents sont catégorisés suivant leur système source. Certains modules sont répétés dans les trois systèmes ce qui sous-entend qu'ils possèdent tous des documents en commun. Il sera nécessaire d'identifier lesquels de ces documents seront utiles pour notre travail. À l'étape 4, plusieurs bases de données retrouvées sont communes aux autres ordinateurs.

Le sous-système « *prognomix-DSS* » utilise deux bases de données :

- a) « *Prognomix_test* »
- b) « *Experiment* »

Alors que le sous-système « *Experiment Manager* » utilise uniquement la base de données :

- a) « *Experiment* »

À l'étape 5, un nouveau système, sur lequel travaillaient les équipes de développement, a été trouvé. Le système est listé parmi les systèmes présentés plus haut dans cette section. C'est le système « *Experiment Manager* ». C'est le sous-système permettant d'assurer le suivi, la traçabilité et la reproductibilité des expériences effectuées par le groupe de chercheurs qui travaillaient dans le laboratoire de recherche de Prognomix Inc. Il sera documenté à la section 2.3.4.5.

2.4.3 Catégorisation des informations

Après l'analyse de tous les artefacts découverts à la section précédente, une classification est faite. Suivant leurs relations avec le système visé, ils sont classifiés en un ensemble de tâches à effectuer afin de faire fonctionner et extraire les connaissances du système. À la suite de la catégorisation, la section suivante présente un plan d'action adoptée afin de respecter différentes tâches qui devraient être réalisées pour satisfaire les défis du gestionnaire.

2.4.4 Plan d'action pour la poursuite de la rétro-ingénierie

Après avoir retrouvé des informations utiles de divers artefacts concernant le système à comprendre, un plan d'action est mis en place de façon à poursuivre l'activité de rétro-ingénierie. Une liste de tâches est développée en fonction des artefacts retrouvés.

Tableau 2: Méthodologie --- Liste des tâches

Liste des taches.	Détails
Configuration du système	À partir des données trouvées, il s'avère nécessaire d'essayer de configurer le système.
Test d'exécution du système	Une fois la configuration du système est en place, il faut tester le système pour voir son exécution ou les erreurs qu'il produit.
Comprendre les fonctionnalités	Tester chaque fonctionnalité du système pour voir son fonctionnement.
Identifier les différents modules du système	Identifier le code source de chacun des sous-systèmes à l'exception du <i>FCI_DSS</i> . Présenter un aperçu global du système aux intervenants. Présenter et décrire chaque module, à haut niveau, de chaque sous-système.
Identifier la conception du système	Présentation de haut niveau l'architecture du système et de ses diagrammes découverts et reconstruits.

2.4.4.1 Configuration du système

Sur un des serveurs, un document de configuration a été trouvé. Ce document de configuration est présenté à la section 2 de l'annexe 3. En essayant de suivre les étapes présentées dans ce document, pour configurer le système, des erreurs surviennent. Les erreurs surviennent à cause que la commande « *python setup.py develop* », tente de faire la mise à jour des modules « *Pylons* » déjà installés. « *Pylons* » est un « *cadriciel* » qui permet de développer des applications web avec le langage « *Python* ». Il contient un fichier de configuration « *setup.cfg* » qui liste la version courante de chaque module attendu pour faire fonctionner le système. Une fois cette commande est exécutée, toute la configuration du système est modifiée et le service de l'application n'est plus capable de démarrer. La commande exécutée, fais des mises à jour de nouvelles versions de « *Pylons* » et de ses différents modules. Dans le répertoire où sont installés les paquets sur le serveur de Linux, différentes versions de « *Pylons* » et de ses modules sont visibles. En se documentant sur internet par rapport à la configuration de « *Pylons* » et de ses composants, certaines autres commandes très utiles pour le projet ont été identifiées. Par exemple, la commande « *python setup.py install* » qui permet de faire l'installation de « *Pylons* » et de ses modules. L'un des aspects importants de cette investigation est le message d'erreur reçu à chaque tentative de lancement de l'application. Dans certains messages d'erreurs, le module que système attend pour fonctionner ainsi que le module qui est actuellement installé sont spécifiés. Cet indice suscite des curiosités jusqu'à vouloir supprimer les fichiers « *.egg* » se trouvant dans le « */usr/lib64/python2.5/site-packages/* » dont la version n'est pas celle attendue par le système. Par exemple, il fallait supprimer ce fichier « */usr/lib64/python2.5/site-packages/Pylons-1.0.1-py2.5.egg* » et laisser l'autre fichier « */usr/lib64/python2.5/site-packages/Pylons-0.9.6.2-py2.5.egg* », car ce dernier est nécessaire pour le lancement du système. Ce processus est également répété pour les modules de « *Pylons* » qui sont reliés à la version supprimée.

Après avoir effectué la réinstallation de « *Pylons* » et de ses modules. Après suppressions des fichiers « *.egg* » le système parvient à démarrer et une première présentation, aux

intervenants, a pu être effectuée. Après cette présentation, le système identifié étant une ancienne version, il fallait creuser un peu plus pour trouver laquelle est la plus récente. D'autres démarches ont été entreprises pour trouver la bonne version du système afin d'y appliquer une nouvelle configuration basée sur l'itération précédente.

La section suivante du rapport présente les démarches avancées, effectuées afin d'arriver à configurer le système et ensuite de le faire fonctionner.

2.4.4.2 Analyses et compréhensions des systèmes par le test de configuration

Une fois la bonne version du système identifiée, un certain nombre de tests et analyses sont effectués. Ces analyses et tests sont effectués avec l'objectif de mieux comprendre le système. Celui-ci a été configuré et testé avec succès. Un nouveau document de configuration a été produit de façon à faciliter une configuration et une exécution ultérieure plus rapide. Voir annexe 5 pour la nouvelle documentation pour la configuration de la version la plus récente du système.

2.4.4.3 Compréhension des fonctionnalités du système

Cette étape explique la stratégie appliquée pour comprendre les fonctionnalités du système. D'abord, une fois le système est lancé, le menu principal est affiché avec l'accès aux différentes fonctionnalités du système (voir annexe 6). Certains écrans du système ne sont pas accessibles ou sont accessibles, mais le cas d'utilisation n'est pas connu ou documenté. Pour comprendre certaines fonctionnalités, le recours à un bio-informaticien est incontournable, car nous n'avons suffisamment les connaissances du domaine. Un bio-informaticien a une meilleure compréhension du processus de l'expérimentation d'un laboratoire de recherche en santé. À l'annexe 6, avec copies d'écrans associés, certaines fonctionnalités sont expliquées de manière à permettre à d'autres équipes, dans le futur, d'avoir une idée claire de ce que fait chaque fonctionnalité du système.

2.4.4.4 Identification des modules du système

Le logiciel étant maintenant fonctionnel, les fonctionnalités peuvent être étudiées, mais certains paramètres du système nécessitent encore une compréhension plus profonde afin de déterminer leurs utilités. L'identification des modules du système semble être l'un des points clés pouvant permettre de se former un modèle mental du système étudié. En se référant à l'ouvrage de Paul Clément (Clément et al., 2010), qui décrit la manière de documenter l'architecture d'un système, la description du système y sera basée. Dans son ouvrage, il démontre trois (3) styles d'architecture de système et la façon appropriée pour les documenter suivant le style adopté. Dans ce travail, les modèles qui seront présentés, refléteront le plus fidèlement possible, les approches de modélisations architecturales proposées par l'auteur dans son ouvrage.

2.4.4.4.1 Vue organisationnelle des « *packages* » du système

Le système « *Biogenix-DSS* » est divisé en deux sous-systèmes. Le sous-système « *Prognomix-DSS* » destiné à l'administration et aux chercheurs. Le sous-système « *Experiment Manager* » pour gérer les expériences ainsi que les paramètres pouvant aider la reproduction de ces dernières. Chaque sous-système est structuré en unité organisationnelle appelée « *Package* ». Chaque « *package* » est constitué de fichier et de « *sous-package* ». Pour mieux comprendre la structure globale du système « *Biogenix-DSS* », quelques figures ont été créées et sont présentées dans la section suivante.

2.4.4.2 Vue globale des modules du sous-système « *Prognomix-DSS* »

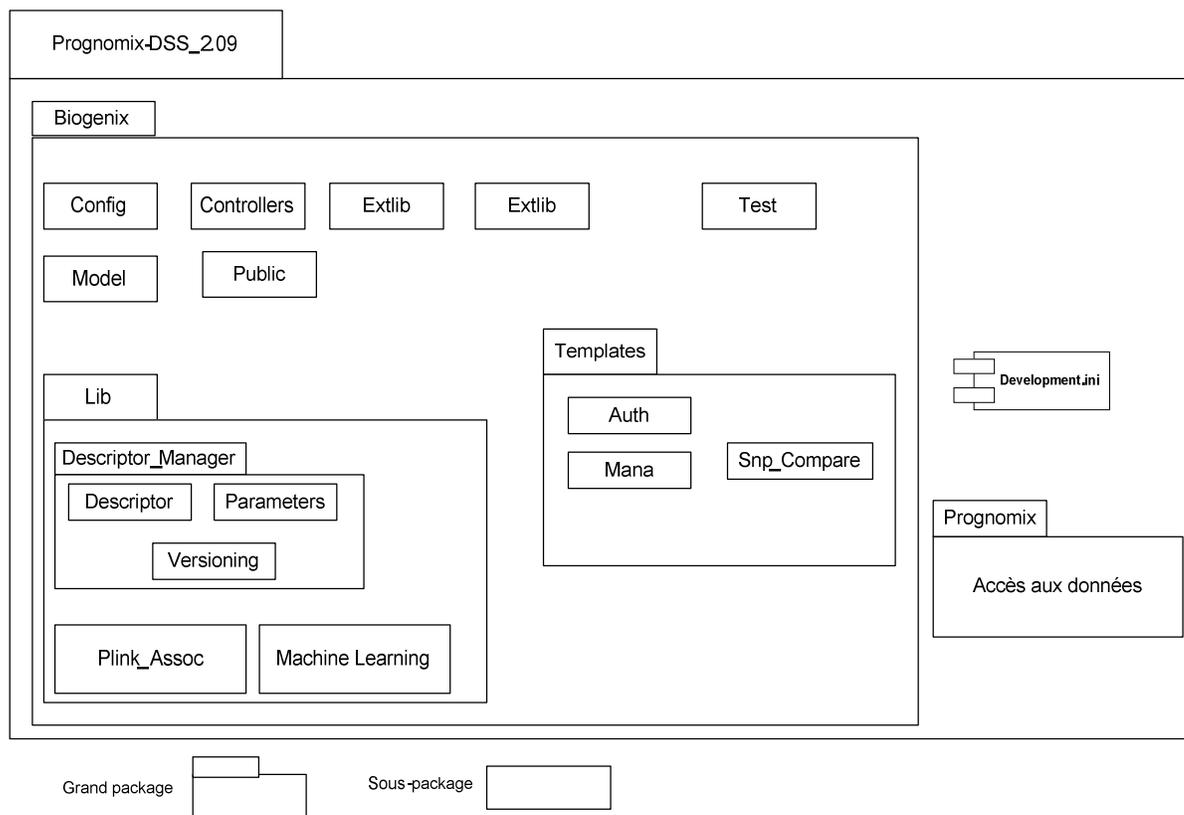


Figure 5: Vue organisationnelle du sous-système « *Prognomix-DSS* »

2.4.4.4.3 Vue globale des modules du sous-système « *Experiment Manager* »

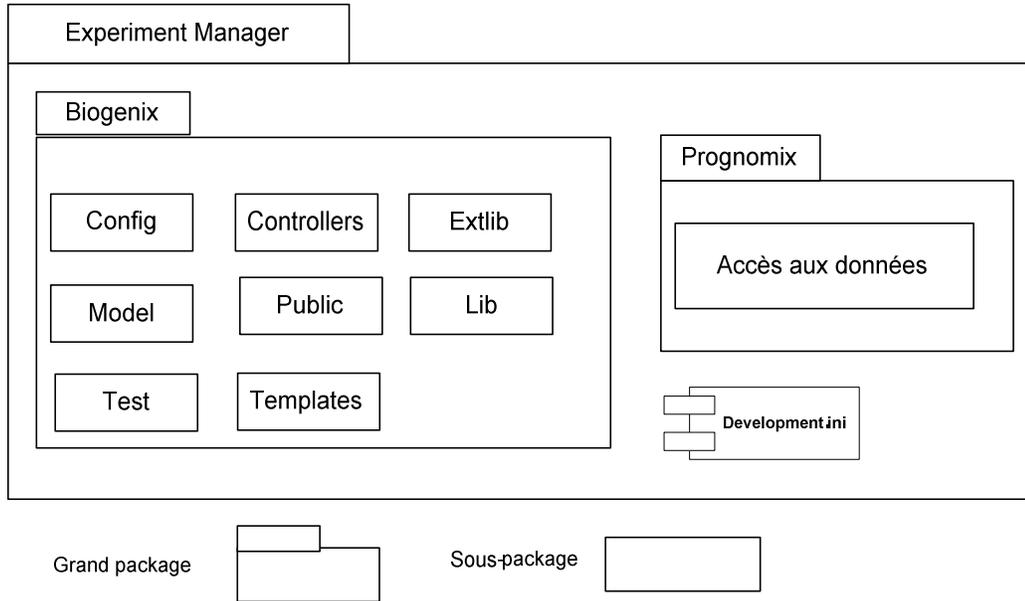


Figure 6: Vue organisationnelle du sous-système « *Experiment Manager* »

Les deux figures ci-dessus présentent l'unité organisationnelle du sous-système « *Prognomix-DSS* » et celle du sous-système « *Experiment Manager* » respectivement. Ci-dessous le rôle de chaque paquet, communs aux deux sous-systèmes, est présenté.

Controllers : Le répertoire « *controllers* » contient les contrôleurs de l'application. Les contrôleurs sont des composants qui prennent en charge les requêtes *HTTP* et renvoient des requêtes *HTTP*. Ils interagissent souvent avec le code des modèles et des « *templates* ».

Model : le répertoire « *model* » est l'endroit où est configuré le modèle de la base de données. C'est typiquement l'endroit où les tables et les relations sont définies.

Lib : Le répertoire « *lib* » inclut du code « *Python* » partagé par différents contrôleurs et bibliothèques de tierce partie.

Config : Le répertoire « *config* » inclut le code généré par le « cadriciel » et est exposé à l'application pour personnalisation.

Tests : Le répertoire « *tests* » est l'endroit où vous pouvez ajouter des tests « *Python* » automatisés pour l'application.

Extlib : Le répertoire dans lequel sont entreposées certaines bibliothèques externes qui seront utilisées par le système.

Templates : Le répertoire « *templates* » est l'endroit où les modèles de vues sont stockés.

Public : Le répertoire public inclut les fichiers statiques de l'application (c.-à-d., *HTML*, fichiers *CSS*, *JavaScript*, etc).

development.ini : C'est le fichier de configuration de l'application. Ce fichier inclut des paramètres comme l'adresse « *IP* » et le port « *TCP* » que le serveur doit écouter, la chaîne de connexion de la base de données, etc.

setup.cfg et setup.py : Ces fichiers contrôlent les différents aspects concernant la manière dont l'application est « packagée » lors de sa distribution.

Prognomix : est le répertoire qui contient la classe permettant de se connecter à la base de données. Ce répertoire contient, entre autres, les fichiers d'authentification du système.

2.4.4.5 Identification de la conception du système

Documenter le module de la traçabilité et de la reproduction des expériences reste l'objectif premier de ce travail de rétro-ingénierie. Toutefois, l'architecture du sous-système « *Prognomix-DSS* » sera présentée afin de mieux jauger le travail de rétro-ingénierie

nécessaire. La section qui suit présente, une vue de haut niveau du concept de traçabilité et de reproduction adopté par ce système.

2.4.4.5.1 Architecture du sous-système *Prognomix-DSS*

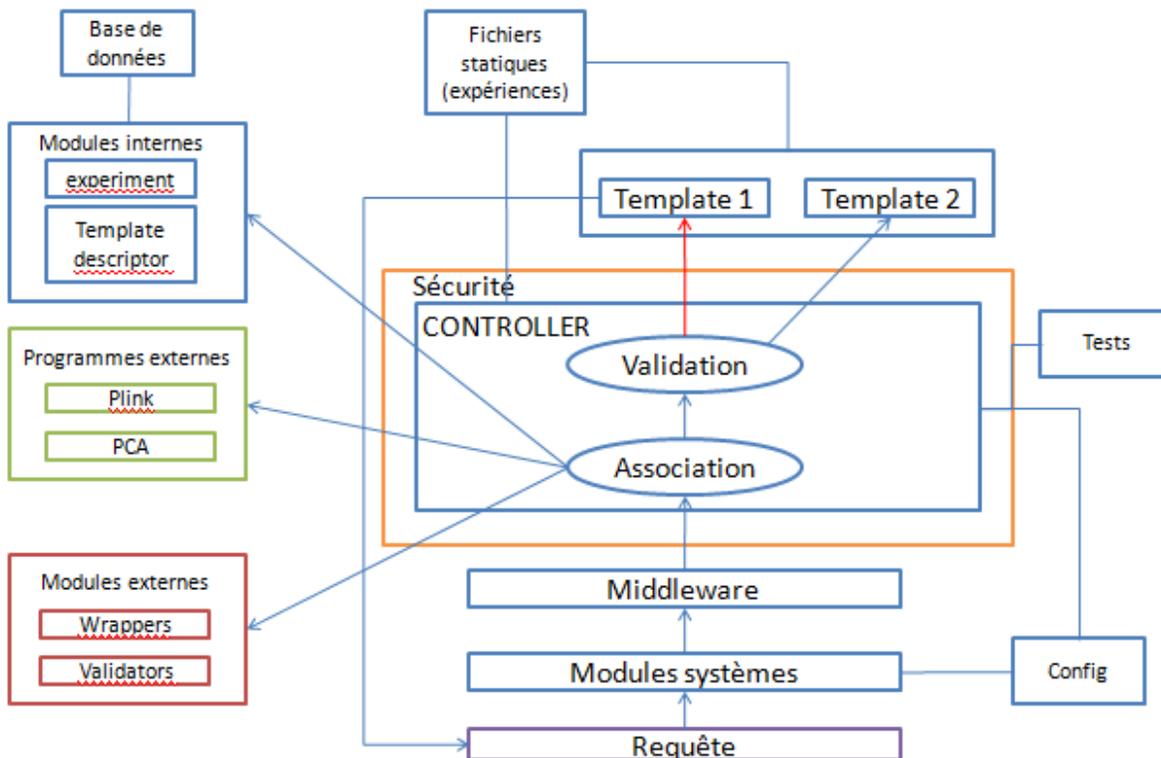


Figure 7: Vue haut-niveau de l'architecture du sous-système « *Prognomix-DSS* »

2.4.4.5.2 Traçabilité et reproduction des expériences

Le module de traçabilité développé dans ce système est similaire à un « aide-mémoire ». Ce module est développé pour enregistrer n'importe quelle analyse (expériences) indépendamment de son format et de son volume.

Le concept d'analyse est totalement générique dans l'environnement de « *Biogenix* ». Lors d'une expérience, l'utilisateur peut donc stocker tous les résultats qu'il désire : que ce soit des

données brutes, d'imagerie médicale, de télémétrie ou d'histologie et ainsi conserver une trace de tous les paramètres qu'il juge pertinent.

2.4.4.5.2.1 L'objectif de la conception d'un module de traçabilité.

Les analyses effectuées dans un laboratoire de recherche sont en constante évolution : les logiciels changent, l'équipement est remplacé, de nouveaux protocoles sont publiés, etc. Les chercheurs auront toujours besoin d'enregistrer et de reproduire des analyses. Les sous-systèmes « *Experiment Manager* » et « *Prognomix-DSS* » guident le chercheur dans cette démarche en lui permettant de construire à partir d'une interface graphique, un modèle de traçabilité pour son analyse. Un peu comme un cahier de laboratoire qui comporterait des champs prédéfinis. Le chercheur est donc assuré d'enregistrer le détail de l'information concernant ses analyses de manière cohérente à travers le temps.

Dans le cas où des expériences sont effectuées en dehors de l'environnement automatisé de traçabilité de « *Biogenix-DSS* », le chercheur doit seulement spécifier les fichiers à sauvegarder et fournir les paramètres à conserver ainsi que leurs valeurs. « *Biogenix-DSS* » se charge ensuite d'insérer la nouvelle expérience dans sa base de données après quoi l'expérience peut être consultée, recherchée et même reproduite.

2.4.4.5.2.2 Architecture de données de la traçabilité.

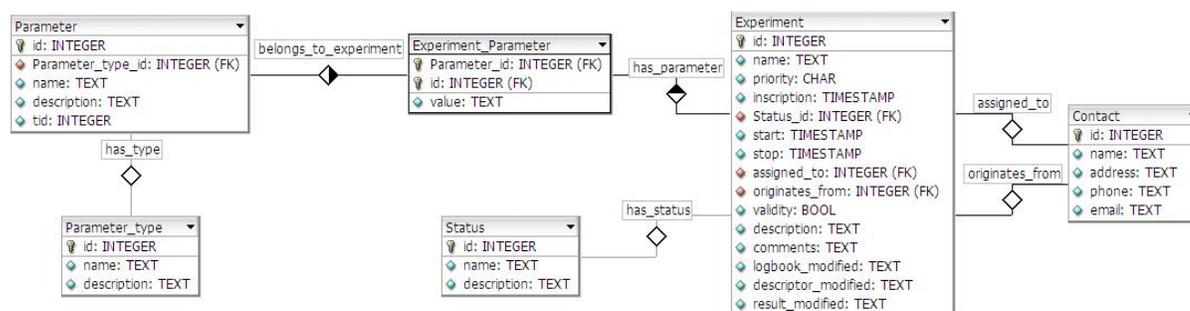


Figure 8: Architecture de données du module de traçabilité.

Ce schéma est reproduit à partir des informations la base de données « *Experiment* » dans laquelle s'enregistre tous les paramètres d'une expérimentation servant à sa traçabilité et sa reproduction.

Traçabilité : Les tables « *experiment* », « *contact* », et « *status* » contiennent les informations relatives au « logbook » d'une expérience, garantissant la traçabilité des expériences effectuées.

Reproductibilité : Les tables « *experiment_parameter* », « *parameter* » et « *parameter_type* » contiennent les informations relatives aux entrées/sorties d'une expérience, visant sa reproductibilité.

2.4.4.5.2.3 Architecture logique du module de traçabilité

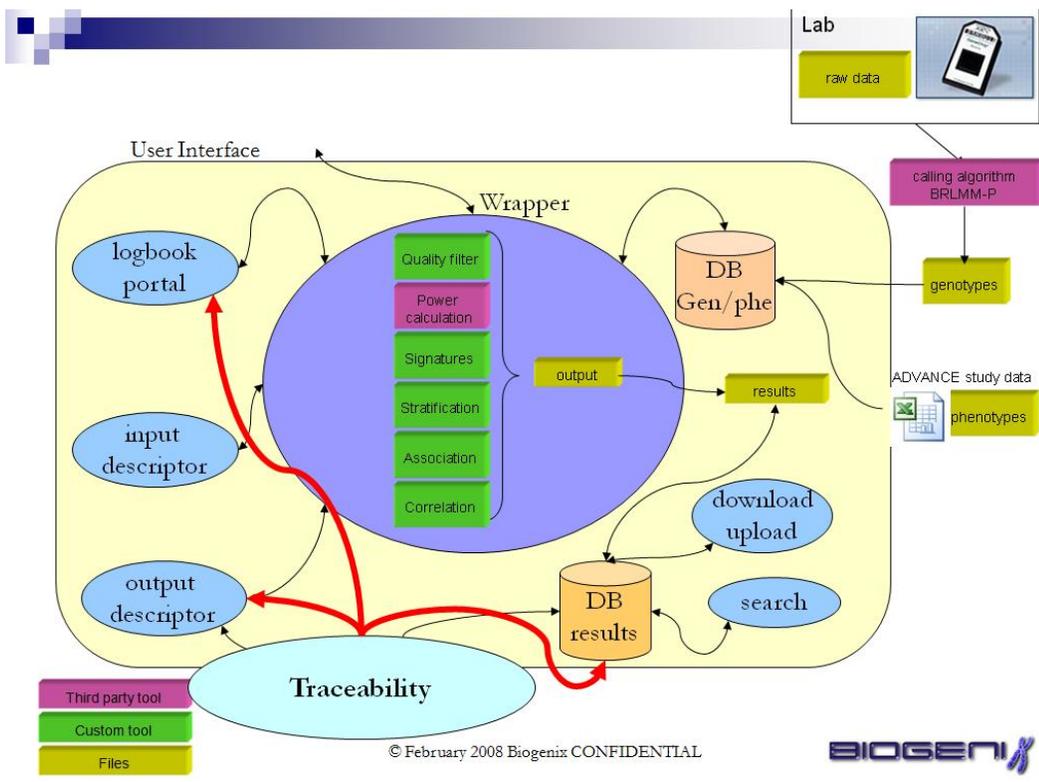


Figure 9: Architecture logique de la traçabilité

Les éléments clés à retenir dans ce modèle sont les suivants :

2.4.4.5.2.3.1 Portail Logbook « Logbook Portal »

Le portail *logbook* est un module de « *Experiment Manager* » permettant de retracer rapidement une expérience qui a été réalisée. Une fois connecté à l'application, l'utilisateur arrive à la page d'accueil (voir figures 10 et 11) de l'application où est affiché le *logbook* (priorité, date d'inscription, date de départ, date d'arrêt, statut, personne responsable, description et commentaires) des expériences. Afin de faciliter la visualisation, l'utilisateur peut trier les expériences selon le critère de son choix (Ex.: de la plus récente à la plus ancienne).

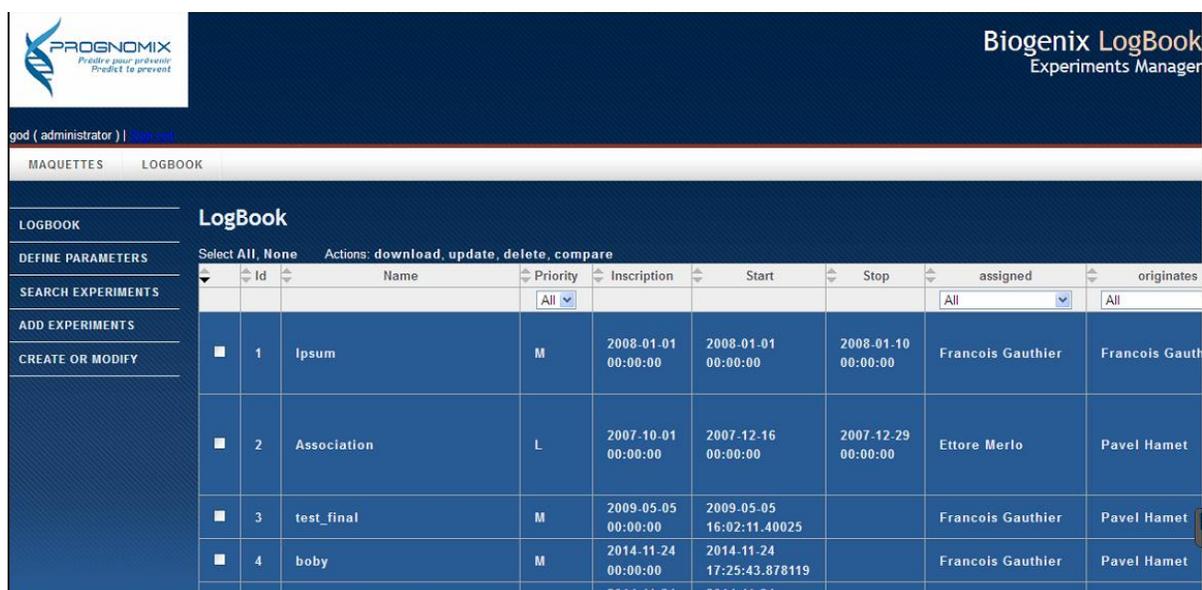
Le système offre de nouvelles fonctionnalités comme :

« *Download* » : pour télécharger une expérimentation sous forme d'un document compressé

« *Update* » : pour mettre à jour une expérimentation.

« *Delete* » : pour supprimer une expérimentation

« *Compare* » : pour comparer deux expérimentations.



Id	Name	Priority	Inscription	Start	Stop	assigned	originates
1	Ipsum	M	2008-01-01 00:00:00	2008-01-01 00:00:00	2008-01-10 00:00:00	Francois Gauthier	Francois Gauthier
2	Association	L	2007-10-01 00:00:00	2007-12-16 00:00:00	2007-12-29 00:00:00	Ettore Merlo	Pavel Hamet
3	test_final	M	2009-05-05 00:00:00	2009-05-05 16:02:11.40025		Francois Gauthier	Pavel Hamet
4	boby	M	2014-11-24 00:00:00	2014-11-24 17:25:43.878119		Francois Gauthier	Pavel Hamet

Figure 10: *Logbook* l'outil de traçabilité des expériences

LOGBOOK		7 - markov_test	
DEFINE PARAMETERS	Logbook		
SEARCH EXPERIMENTS	inscription:	2008-01-28 12:42:31	
ADD EXPERIMENTS	scheduled_stop:	2008-01-29 12:42:31	
CREATE OR MODIFY	name:	MARKOV_RECESSIVE_COMP_B	
	resdir:	markov_test	
	real_stop:	None	
	sid:	running	
	result_modified:	None	
	real_start:	None	
	validity:	None	
	scheduled_start:	2008-01-28 12:42:31	
	priority:	M	
	logbook_modified:	1210903380	
	comments:	None	
	assigned_to:	Pierre-Luc Brunelle	
	descriptor_modified:	1210256377	
	originates_from:	Pierre-Luc Brunelle	
	id:	7	
	description:	Complications, 1234 subjects, recessive model -- Correct genotype files	

Figure 11: Logbook, détails de l'expérience

En comparant deux expérimentations, dans la figure ci-dessous, le chercheur aura des détails comparatifs très intéressants comme :

- Les paramètres ajoutés qui sont de couleur jaune.
- Les éléments en rose qui désignent des paramètres qui sont différents les uns des autres.
- La couleur grise qui indique que les paramètres sont identiques.

LOGBOOK		Compare Experiments #3 and #7	
DEFINE PARAMETERS	<input type="checkbox"/> Identical <input type="checkbox"/> Added <input type="checkbox"/> Different		
SEARCH EXPERIMENTS	Logbook of #3		Logbook of #7
ADD EXPERIMENTS	inscription:	2009-05-05 00:00:00	2008-01-28 12:42:31
CREATE OR MODIFY	scheduled_stop:	None	2008-01-29 12:42:31
	name:	test_final	MARKOV_RECESSIVE_COMP_B
	resdir:	test_final	markov_test
	real_stop:	None	None
	sid:	completed	running
	result_modified:	None	None
	real_start:	None	None
	validity:	None	None
	scheduled_start:	2009-05-05 16:02:11.40025	2008-01-28 12:42:31
	priority:	M	M
	logbook_modified:	1241563732	1210903380
	comments:	comm	None
	assigned_to:	Francois Gauthier	Pierre-Luc Brunelle
	descriptor_modified:	None	1210256377
	originates_from:	Pavel Hamet	Pierre-Luc Brunelle
	id:	3	7
	description:	desc	Complications, 1234 subjects, recessive mo
	Descriptor of #3		Descriptor of #7
	descriptor_cfg:		/home/blaforet/Desktop/workflow/biogenix/bi
	descriptor_cfg_mtime:		1210256377
	experiment_dir:	test_final	markov_test
	id:	3	7
	logbook_cfg:	/home/blaforet/Desktop/workflow/biogenix/bi	/home/blaforet/Desktop/workflow/biogenix/bi

Figure 12: Traçabilité par la comparaison de deux expérimentations

2.4.4.5.2.3.2 Paramètres de sorties « *Output descriptor* »



Figure 13: Paramètres de sorties de l'expérimentation

Les descripteurs sont les différents paramètres utilisés lors de l'analyse ou expérimentation. Ces paramètres permettront la reproduction de cette même expérience dans le temps.

2.4.4.5.2.3.3 Base de données des résultats « *DBresults* »

Une fois l'expérience est effectuée, les résultats sont automatiquement ajoutés dans la base de données. L'expérimentation peut être effectuée dans le système en utilisant les fonctionnalités : « Association »; « Machine Learning »; « PCA »; « Descriptive Stats »; « SNP Comparison ». Toutefois, les expériences effectuées hors de l'environnement automatisé sont également acceptées par le système. Les résultats de telles expériences pourraient s'ajouter dans le module « *Public\ExperimentsDB* » sous forme de fichiers :

- config.dat
- logbook.dat

- options.dat
- affected.dat
- descriptor.dat
- nonAffected.dat
- r1-details.ps
- r1-pred.csv
- r1-roc.ps
- snpNameSet.dat
- r1-results.dat, etc.

Le système s'occupe d'en extraire les paramètres de ces résultats et les ajouter dans les tables correspondant de la base de données.



2.4.4.5.2.3.4 Évolution du module de traçabilité.

Le sous-système « *Experiment Manager* » a été développé en 2007, à cette époque l'ajout des expériences dans le système, était semi-automatique. Il était ainsi parce que les expériences devraient être d'abord ajoutées dans un répertoire spécifique « *Public\ExperimentsDB* » afin que l'application les détecte automatiquement. Le chercheur pouvait alors valider les nouvelles expériences après vérifications afin qu'elles soient insérées dans la base de données. En 2008, l'ajout des fonctionnalités d'analyses dans le sous-système « *Experiment Manager* » donnant naissance au sous-système « *Prognomix-DSS* ». Ces ajouts permettent au système d'être plus flexible, car il peut recevoir des expériences hors de son environnement

et également ceux-là effectués dans son environnement. Aucune des fonctionnalités du sous-système existant n'a été modifiée. Le sous-système « *Prognomix — DSS* » peut être utilisé en lieu et place du sous-système « *Experiment Manager* »

2.4.4.5.2.4 Architecture logicielle du module de traçabilité

Dans cette section, chaque module sera présenté avec les classes qu'il contient. Les classes qui seront étudiées en détail sont celles qui permettront de comprendre le module lié à la traçabilité. Certaines autres classes seront expliquées afin d'élargir le champ de compréhension pour sa modernisation.

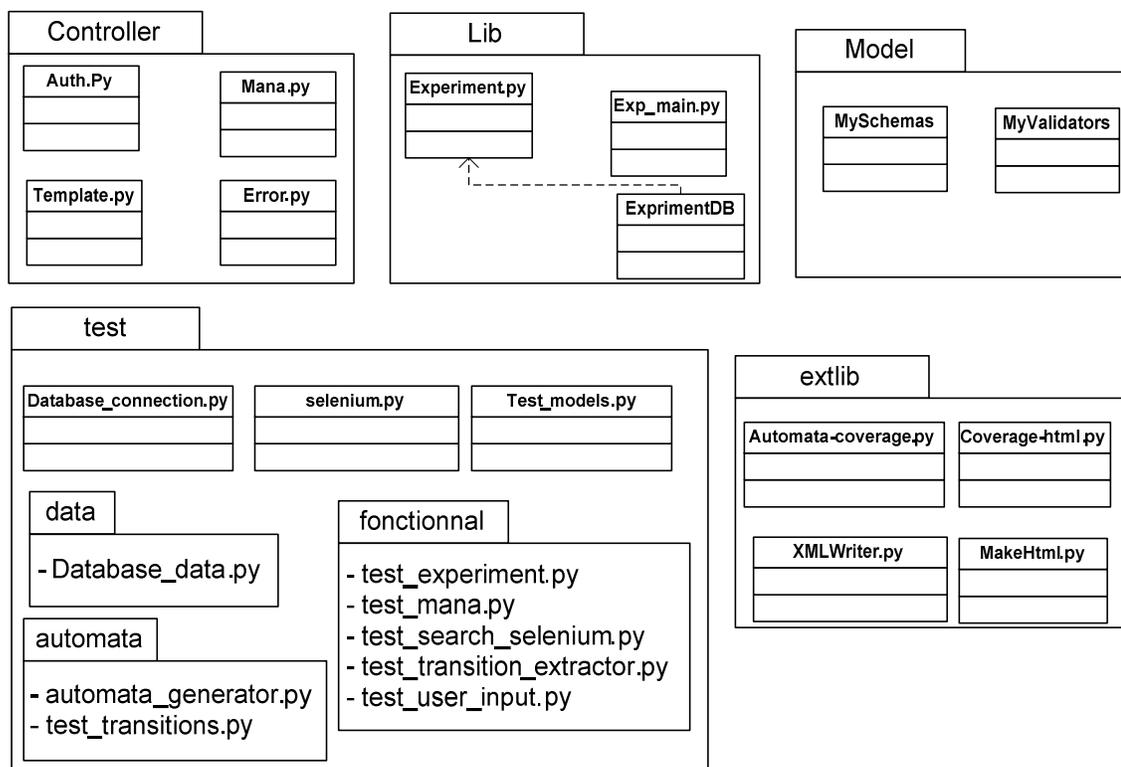


Figure 14: Liste des modules et les classes composantes.

2.4.4.5.2.4.1 Diagramme de classes du module « controller »

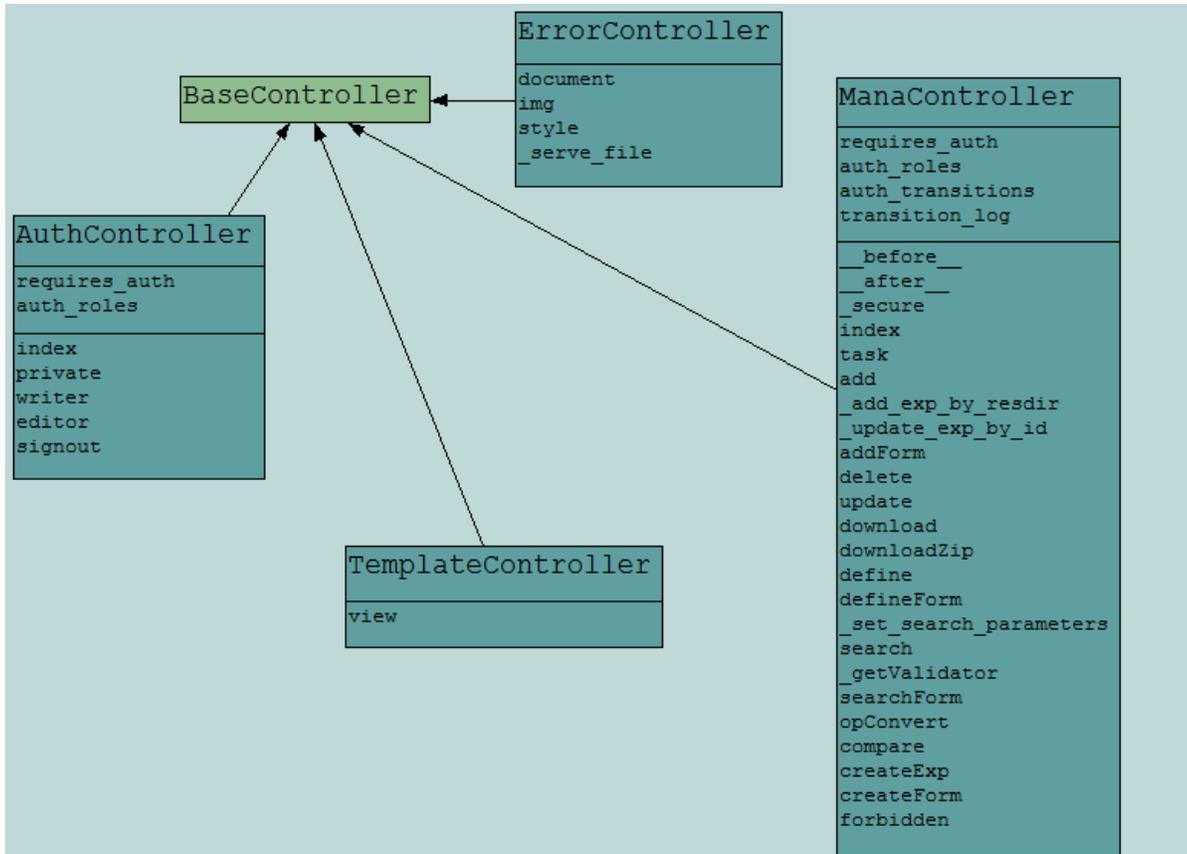


Figure 15: Diagramme des classes du module *Controller*

Classe BaseController : le contrôleur principal utilisé pour interpréter les requêtes des utilisateurs et décider du modèle et de la vue nécessaire pour satisfaire cette requête.

Classe ErrorController : interprète et génère les erreurs quand c'est requis.

Classe AuthController : interprète toutes les requêtes reliant à l'authentification du système et appelle la vue nécessaire. Ex. : Pour faire une recherche, les méthodes « *searchForm()* », « *_set_search_parameters()* » et « *search()* » sont appelés par la vue « rechercher ».

AuthController : interprète toutes les requêtes reliant à l'authentification dans le système et appelle la vue nécessaire.

Classe *TemplateController*: utilisé pour appeler la vue correspondante lors d'une demande par le « *controller* ».

Classe *ManaController* : cette classe interprète les requêtes émanant de chaque vue et s'assure d'appeler le modèle correspondant.

2.4.4.5.2.4.2 Diagramme des classes du module « lib »

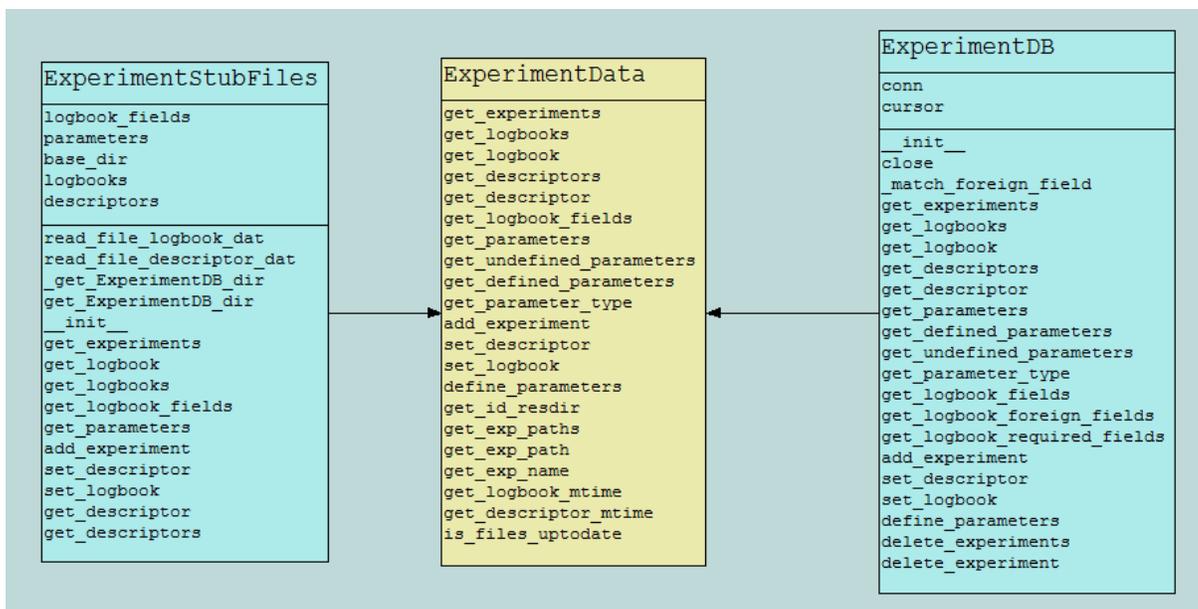


Figure 16: Diagramme des classes du module « lib »

Classe *ExperimentStubFiles* : Cette classe se trouve dans le fichier « *experiment.py* » qui est très important. Elle s'assure d'extraire automatiquement les paramètres d'une expérience dont les fichiers d'expérience sont déposés dans le répertoire « *public\ExperimentsDB* ». Une fois l'extraction est faite, elle appelle les méthodes de la classe « *ExprimentData* » afin de les enregistrer dans les bases de données.

Classe *ExperimentData* : rassemble tous les paramètres utilisés au cours de l'expérience afin de l'insérer dans la base de données.

Classe *ExperimentDB* : Permet d'établir la connexion à la base de données. Une fois la connexion est établie, cette classe appelle la classe « *ExperimentData* » pour extraire les données de l'expérience afin de les ajouter dans les tables respectives.

2.4.4.5.2.4.3 Diagramme de classes pour le module test

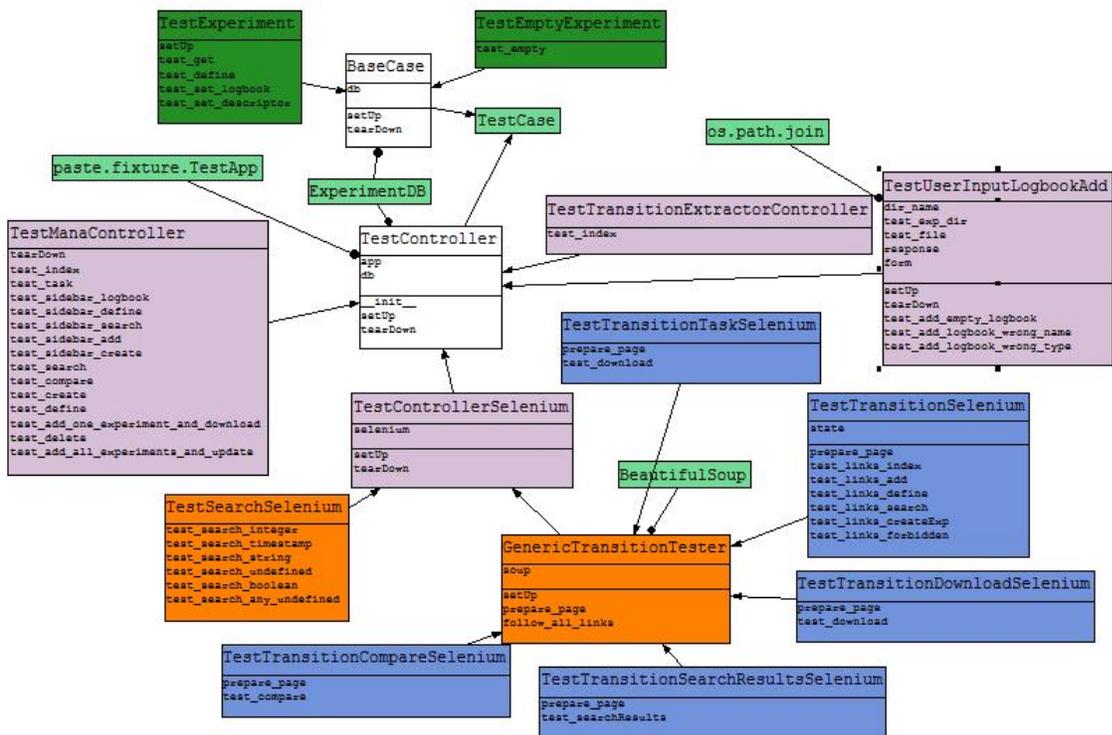


Figure 17: Diagramme de classes du module « tests »

Ce diagramme est utile pour expliquer les tests automatiques du système. Ce module est développé afin de s'assurer que le système satisfait aux contraintes de chaque cas d'utilisation et également aux résultats attendus par ce dernier.

2.5 Techniques et outils utilisés

Le modèle ci-dessous, qui a été présenté dans le chapitre 1 à la section 1.4.1.2 de ce rapport, décrit un ensemble de techniques utiles pour la rétro-ingénierie. Les artéfacts statiques permettent d'appliquer la technique rétro-ingénierie selon une vision statique. Ce qui est très utile dans la recherche des indices menant à retrouver le système sur lequel le travail doit être effectué. Les artéfacts dynamiques permettent d'appliquer la rétro-ingénierie selon une vision dynamique afin de mieux comprendre les fonctionnalités du système.

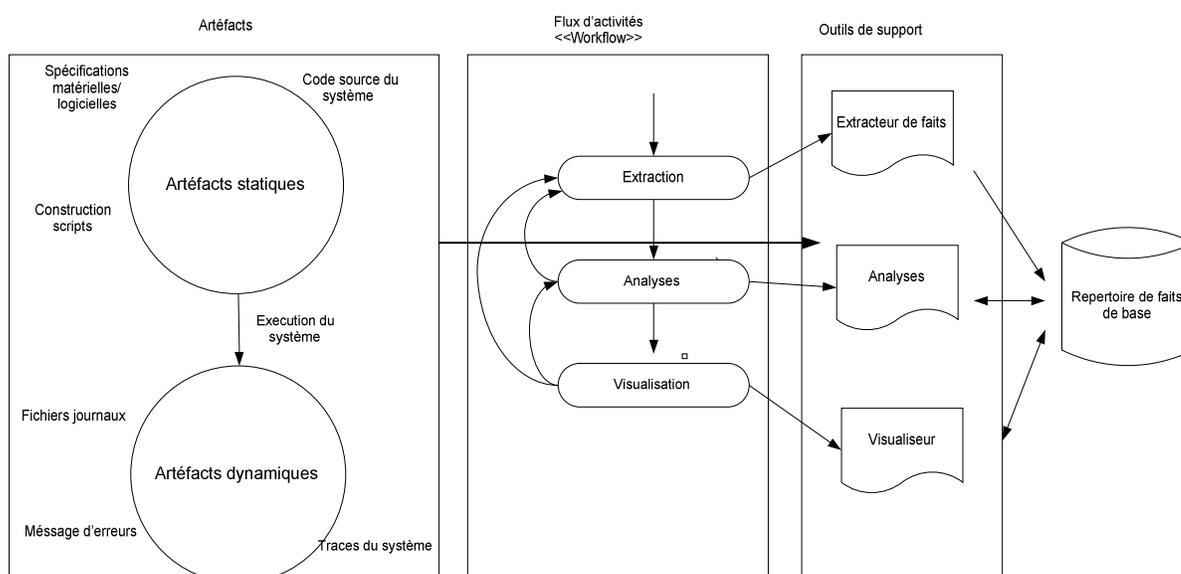


Figure 18: Techniques de rétro-ingénierie se basant sur le modèle de la rétro-ingénierie.

2.5.1 Rétro-ingénierie selon une vision statique

Chaque artéfact (c.-à-d. les documents existants, les codes sources, des diagrammes, un manuel, des copies d'écrans) du système est minutieusement étudié de manière à mieux comprendre l'envergure du système à redocumenter. Les premières copies d'écrans du système ont permis de mieux cibler les recherches dans les différents documents disponibles. Les premiers schémas du système retrouvés sur les ordinateurs ont contribué à faire des recherches plus poussées pour retrouver le code source du système (voir les figures dans l'annexe 2). Des rapports, retrouvés sur les ordinateurs, ont permis de localiser avec

précision l'endroit où se trouvent les répertoires du système et quelques autres systèmes liés (voir le tableau à l'annexe 2).

2.5.2 Rétro-ingénierie selon une vision dynamique

Cette technique permet, à partir de l'exécution du système, de comprendre certaines fonctionnalités. L'interaction avec le système permet d'acquérir de l'information utile concernant des subtilités importantes pouvant aider à comprendre les règles d'affaires derrière la conception. Les connaissances extraites à partir de cette technique facilitent la compréhension du système et la documentation de sa conception.

2.5.3 Technique de rétro-ingénierie de base de données

Une fois identifiée la base de données du système. La technique de rétro-ingénierie permettra:

- de comprendre la structure des données de la base de données
- de générer des modèles entités relations à partir des spécifications d'un modèle de base de données, en utilisant des outils appropriés.

L'outil « *Pgdump* » a été utilisé d'un côté pour faire la sauvegarde de la base de données, mais de l'autre côté pour comprendre sa structure. L'outil « *MS Visio 2007, version professionnelle* » est utilisé à cet effet pour générer le schéma entité relationnelle de cette base de données.

2.5.4 Technique de rétro-ingénierie de code source

Le code source de toute application est perçu comme un artéfact statique. Toutefois, certains outils de rétro-ingénierie facilitent l'extraction de certaines connaissances utiles du code source. L'analyse et la visualisation sont deux outils très importants qui facilitent l'extraction des premières connaissances (c.-à-d., l'organisation du code source, les structures de données utilisées). Le diagramme des classes est le premier artéfact tangible qui peut être extrait par un outil. Cette technique s'avère très utile durant ce travail, car il permet de tirer profit du

code source du sous-système « *Experiment Manager* ». L'outil « *PyNsource* » a été utilisé pour en extraire les diagrammes de classes.

2.6 État d'avancement de l'activité de rétro-ingénierie

La rétro-ingénierie du système « *Biogenix-DSS* » n'est pas complétée. Il reste beaucoup d'artéfacts à documenter, mais il fallait se concentrer plus particulièrement sur le module de la traçabilité (« *Experiment manager* »). Tous les artéfacts et ceux, nouvellement créés, seront très utiles pour compléter la rétro-ingénierie. Concernant le sous-système de traçabilité, les artéfacts les plus importants ont été trouvés et expliqués. Aux annexes, des artéfacts concernant la configuration et la documentation d'autres modules du système « *Biogenix-DSS* » et du système « *FCI_DSS* » sont décrits. C'est une source d'indices importants pour ceux qui vont utiliser ce travail pour moderniser le système.

2.7 L'effort déployé pour le travail de rétro-ingénierie

La rétro-ingénierie est une discipline qui nécessite beaucoup de compétences. Savoir identifier les indices qui conduisent vers d'autres indices est la compétence de base que requiert cette discipline. Les autres compétences sont secondaires. Toutefois, capable de mesurer les efforts pour une activité de rétro-ingénierie, exige une compétence organisationnelle de celui qui la pratique. Pour mesurer l'effort mis sur ce travail, le temps passé sur chaque tâche du projet est enregistré sous un format date et l'heure du début et de fin.

Ce tableau ci-dessous présente approximativement les efforts en heures/mois mis sur ce projet.

Tableau 3: L'effort déployé pour le travail de rétro-ingénierie

Tableau du nombre d'heures de travail (approximatif)			
		Tâches	P/M
	Septembre	Analyse et collecte d'information sur le système	41

2014	Octobre	Analyse et test de configuration du système	85
	Novembre	Configuration du système et présentation du premier Démo	91
	Décembre	configuration de FCI_DSS et Dépannage des erreurs du système	119
2015	Janvier	Mise en marche du système et Configuration du système Trac	35
	Février	Analyse des codes sources et identifications des modules du système	31
	Avril	Recherche documentaire	20
	Mai	Rédaction et révision sur la rétro-ingénierie — rapport	90
	Juin	Rédaction et révision sur la conception et la traçabilité du système — rapport	98
	Juillet	Rédaction- révision finale — rapport final	90
		Total	700

Dans ce tableau ci-dessus, 53% de l'effort déployé pour le projet est misé sur la collecte des informations et les tests de configuration du système. Tandis que 24% de cet effort est accordé à l'activité de rétro-ingénierie du code source et de la base de données du système. Le 23% d'efforts restants est misé pour la recherche documentaire, sur la rédaction et sur la révision du rapport. Le plus haut pourcentage démontre le temps nécessaire, dans une activité de rétro-ingénierie, pour chercher des indices, comprendre et faire fonctionner les systèmes.

2.8 Conclusion chapitre 2.

La mise en application, dans un contexte réel, des théories de la rétro-ingénierie évoquées dans le chapitre précédent, nous rappelle une pensée de Paul Humphreys (Humphreys P., 2004). Humphreys est l'auteur de l'ouvrage basé sur les méthodes computationnelles « *Extending Ourselves: Computational Science, Empiricism, and Scientific Method* ». Dans cet ouvrage, l'auteur livre ses pensées sur la théorie en démontrant qu'il existe une frontière ou une distance, entre une théorie et sa mise en application. Dans un autre paragraphe du même ouvrage, l'auteur argumente que la théorie est muette des fois, nous devons parler à sa place. « *The obvious solution is to be explicit about which parts of the theory are to be construed realistically and which merely instrumentally. The theory itself is mute, we must speak on its behalf.* » (Humphreys, 2004, p.85). Tout ceci c'est pour expliquer que la compréhension et l'interprétation des théories de la rétro-ingénierie constituent l'enjeu

auquel nous nous sommes confrontés lors de cette mise en application. Ces théories sont considérées comme des piliers de ce projet, car elles nous servent de modèle référentiel, mais elles ne fournissent pas assez de détails quant à leur mise en application dans un contexte concret comme dans le nôtre. Le modèle de la rétro-ingénierie présenté dans la section 1.4.1.2 a été utilisé pour le projet. Ce modèle présente sommairement des flux activités relatifs à l'extraction, l'analyse et la visualisation des informations du système. Il indique entre autres, des artefacts statiques, mais il n'explique pas le procédé pour en extraire les connaissances qui y résident. De même pour les artefacts dynamiques, qui constituent un point essentiel pour ce projet, son exploitation n'est pas expliquée non plus. Les outils techniques à utiliser ne décrivent pas le « comment » des choses, ils présentent uniquement leur but. Se référant à la procédure présentée à la section 1.4.1.3 du document, relative aux étapes à suivre pour entamer le projet, sa compréhension est floue. Cette procédure ne fournit aucun détail pertinent pouvant aider à une interprétation. Son interprétation dépend de nos jugements et de nos compétences. Bien que la rétro-ingénierie du système « *Biogenix* » a été un succès, au chapitre 3, une amélioration du modèle est envisagée. Quant à la procédure, elle ne peut pas être trop spécifique pour ne pas limiter les champs de recherches. Entre autres, pour aider un travail futur avec les outils, un tableau comparatif de ces outils sera proposé au chapitre 3. Il est évident que le travail de rétro-ingénierie effectué n'est pas complet relatif aux artefacts qui devraient être produits en lien aux différents systèmes existants. Toutefois, le travail effectué servira de référentiel pour l'équipe qui entamera la modernisation du système.

CHAPITRE 3

DESCRIPTION DE L'EXPÉRIMENTATION

3.1 INTRODUCTION

L'application des techniques de la rétro-ingénierie dans le chapitre est d'une importance significative pouvant aider une amélioration future du système. Ce chapitre se donne pour objectif de faire le point sur cette expérience et les résultats obtenus. Il présente des analyses détaillées sur les forces et les faiblesses observées lors de l'utilisation du modèle théorique de rétro-ingénierie. L'effort consacré pour réaliser un tel travail y est aussi présenté. Une telle analyse nous permettra de formuler des recommandations concernant les travaux futurs qui seront effectués. Nonobstant l'objectif d'améliorer l'effort consacré à un tel projet, un retour sur l'expérience d'utilisation du processus et des outils adoptés est également effectué. Après tout, l'expérience qui a été effectuée, après de longs mois, mérite d'être synthétisée. Ce chapitre a pour objectif de présenter des informations qui pourront faciliter la réutilisabilité de ce travail dans une expérience future.

3.2 Sommaire du projet

À la suite de la revue de la littérature du domaine de la rétro-ingénierie logicielle, un modèle et un processus, jugés importants, ont été adoptés pour ce projet. Le processus décrit de manière séquentielle, les étapes à suivre pour recueillir le maximum d'informations sur le système à étudier. L'application de ce processus, dans les démarches pour comprendre le système, a permis de collecter le maximum d'informations nécessaires avant de passer à l'étape d'étude plus en profondeur du système.

Ce modèle est basé sur la rétro-ingénierie par le recouvrement de la conception. Le modèle théorique décrit un ensemble d'artéfacts à identifier qui sont censés aider à mieux comprendre la conception du système et ses fonctionnalités. Ce modèle est le référentiel théorique pour ce projet, toutefois, sa mise en application nécessite de solides compétences

dans le domaine de la recherche et la rétro-ingénierie. Cela n'empêche pas qu'en suivant ce modèle, plusieurs techniques de rétro-ingénierie associées ont pu être adoptées pour le projet. Les différentes techniques de rétro-ingénierie appliquées pour extraire les connaissances utiles du système ont permis d'atteindre l'objectif du projet. L'analyse du tableau 3, présenté au chapitre précédent, décrit que l'effort d'analyse de conception a représenté 24% de l'effort tandis que 53% du reste de l'effort a été dépensé sur le processus de collecte d'informations de configuration du système existant. Donc, la plus grande partie de l'effort a été nécessaire afin de permettre une meilleure classification des informations collectées et ainsi réduire l'effort de documentation de l'architecture existante du système.

3.3 Critique de la rétro-ingénierie

Cette section vise une analyse critique constructive par rapport au modèle théorique publié et aux techniques utilisées.

3.3.1 Faiblesse du modèle

Le modèle sur lequel est basé tout le travail de rétro-ingénierie n'est pas parfait. Sa problématique principale est qu'il est trop général donc il est imprécis. La rétro-ingénierie logicielle est très utilisée, pratiquement, lors de l'intégration ou de la refonte d'un système complexe existant pour lequel les auteurs sont souvent absents. Le modèle théorique de référence ne fournit pas assez de détails pour aider son utilisateur et spécialement dans le cas où aucun autre artéfact que le code source n'est disponible. Il ne fournit également aucun détail pouvant aider son utilisateur dans le cas où le système existant n'est pas fonctionnel ou opérationnel.

3.3.2 Force du modèle

Bien que le modèle théorique de référence possède cette faiblesse, sa force réside dans sa simplicité de la représentation des « méta-activités » qui doivent être suivies lors de l'activité

de rétro-ingénierie. Cette faiblesse mène à une opportunité de création de nouvelles connaissances.

3.3.3 Opportunité

Le modèle théorique de référence permet de souligner l'opportunité qui s'offre aux entreprises qui possèdent des systèmes « *Legacy* » ou patrimoniaux. Avant que le système soit complètement arrêté, c'est l'occasion d'utiliser le modèle pour préparer certains artefacts qui serviront de bases de connaissances pour la prochaine étape lors de sa réingénierie.

3.4 Détails pour enrichir le modèle théorique de la rétro-ingénierie

Afin de permettre une meilleure compréhension et interprétation du modèle théorique, nous avons jugé d'en créer une autre version enrichie pour mieux guider ses utilisateurs futurs. L'objectif de cette proposition est de permettre une utilisation plus pratique du modèle dans un contexte où il y a peu de documentation et d'artefacts existants.

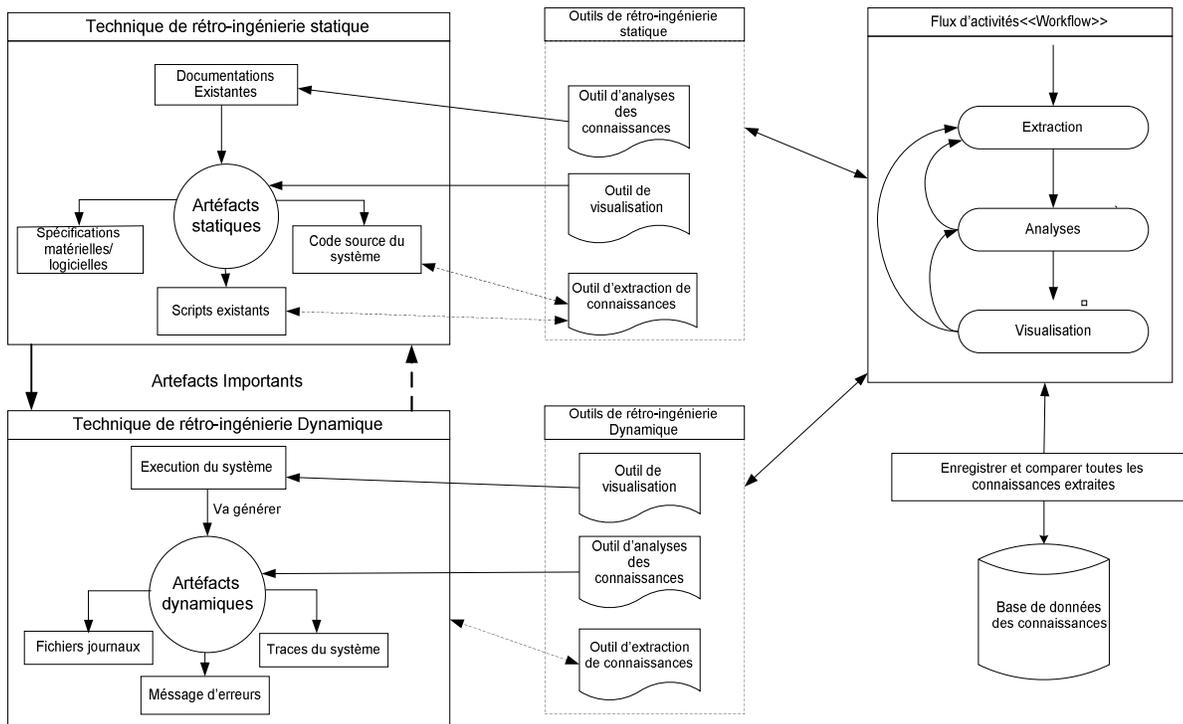


Figure 19: Proposition d'enrichissement au modèle théorique de référence de la rétro-ingénierie

La figure 19 présente deux techniques de rétro-ingénierie à appliquer dans une situation concrète en fonction des artéfacts disponibles.

- **Technique de rétro-ingénierie statique :** Dans le cas des artéfacts statiques mentionnés dans le modèle ci-dessus, il faut appliquer la rétro-ingénierie selon une vision statique, qui permettra de visualiser, d'analyser et d'en extraire toutes les connaissances utiles. Toutes les connaissances doivent être enregistrées sous forme de base de connaissances. Cela permettra de comparer les nouvelles connaissances aux autres connaissances existantes pour savoir si de telles connaissances s'avèrent nécessaires d'être considérées pour la poursuite des prochaines étapes.
- **Technique de la rétro-ingénierie dynamique :** Cette technique est utilisée dans le cas où le système est opérationnel ou fonctionnel. Son application permettra d'extraire des connaissances directement des règles d'affaires et de la conception du

système. Des outils de visualisation des fichiers générés par l'exécution des systèmes doivent être utilisés pour une meilleure analyse et extraction de connaissances utiles pour les prochaines étapes.

- **Outils de rétro-ingénierie :** À chaque type de technique de rétro-ingénierie, il est nécessaire d'appliquer les outils appropriés pouvant permettre de mieux visualiser, analyser et extraire les connaissances des artefacts.
- **Flux-Activité :** Identifie différents types d'opérations qui peuvent être menés sur les artefacts disponibles.
- **Base de connaissances :** Les connaissances extraites constituent des indices du système à étudier. Il est crucial de sauvegarder chaque indice du système afin de pouvoir les comparer avec les nouveaux indices qui seront trouvés, au fur et à mesure que les autres étapes sont franchies.

3.5 Analyse des outils disponibles de la rétro-ingénierie existants

Cette section présente une courte analyse des différents outils utilisés pour effectuer la rétro-ingénierie de notre système à l'étude :

Recherche et Analyse : les deux premiers outils utilisés pour faire ce projet. Ces outils sont d'une importance capitale surtout lorsqu'il fallait trouver des indices menant vers l'identification du vrai système sur lequel le projet doit être fait.

Les compétences techniques : La technique constitue un outil primordial dans la rétro-ingénierie. Elle est très utile surtout quand il faut tester et configurer un système dont aucune documentation de configuration n'existe.

PyNsource : c'est un outil très important surtout, quand il existe des codes sources des systèmes. Cet outil permet automatiquement de charger les codes sources et d'en extraire les

diagrammes de classes qui y sont associés. Comme son nom l'indique, l'outil s'applique uniquement aux codes sources de python.

MS Visio, version professionnelle : Cet outil est très pratique pour faire la rétro-ingénierie d'une base de données existante. Cette fonctionnalité de rétro-ingénierie de base de données existe dans la version professionnelle de Visio. Il a permis d'extraire le schéma relationnel des bases de données existantes pour ce projet.

3.6 Tableau comparatif des différents outils

Le tableau 4 présente différents outils de rétro-ingénierie réels qui sont référés théoriquement au chapitre 1. Dans ce tableau, les outils sont catégorisés suivant leur flux d'activité et leur rapport d'utilisation dans le projet.

Tableau 4:Tableau comparatif des outils par rapport au projet

Outils	Flux d'activité			Projet
	Visualisation	Analyse	Extraction	Utilité pour le projet
Éclipse Dali pour java	Oui	Non	Oui	Aucune
Imagix4D pour C, C++, java	Oui	Non	Oui	Aucune
Instant reverse Python (Visual Paradigm)	Oui	Non	Oui	Aucune
Epydoc	Oui	Oui	Oui	Oui
PyNSource	Oui	Non	Oui	Oui
AnySQL Maestro	Oui	Non	Oui	Non
MS Visio 2007 Professionnel	Oui	Non	Oui	Oui
Recherche et analyse	Oui	Oui	Non	Oui
Compétences techniques	Oui	Oui	Oui	Oui

3.7 Analyse des résultats

Après une longue activité de rétro-ingénierie du système « *Biogenix-DSS* », il s'avère important de présenter une analyse synthèse des résultats obtenus.

La première étape du projet consistait à trouver le système sur lequel la rétro-ingénierie devait être effectuée. À cette étape, des résultats positifs ont été obtenus. Tandis que la deuxième étape du projet consistait à appliquer la rétro-ingénierie sur le système identifié afin de redocumenter la conception et éventuellement les règles d'affaires d'un de ses sous-systèmes.

3.7.1 Résultats de la première étape du projet.

À la première étape du projet, les résultats ci-dessous ont été obtenus :

1. Identification du système cible et de plusieurs autres systèmes connexes.
 - *Biogenix-DSS (Experiment Manager, Prognomix-DSS)* : système incluant un sous-système d'expérimentation et un sous-système de traçabilité.
 - *FCI-DSS* : système incluant un sous-système d'expérimentation sur des animaux et un sous-système de traçabilité.
 - *MEIDEI-DSS*
 - *Trac* : système de billetterie utilisé par l'équipe de développement pour garder une trace des changements et erreurs logiciels corrigés dans les deux systèmes « *Biogenix-DSS* » et « *FCI-DSS* ».
2. Identification de certaines documentations sur la configuration et les fonctionnalités du système.
 - Documentation de spécifications logicielles et matérielles qui héberge le système.
 - Identification des codes sources du système
 - Guide utilisateur pour l'utilisation du logiciel « *Biogenix-DSS* »
 - Documentation de la liste des utilisateurs et chemin d'accès complet de chaque système.

- Document de configuration du système.
- Identification des systèmes informationnels utilisés par le système Biogenix-DSS :
 - o *Prognomix_test* : qui sauvegarde les informations analytiques des expérimentations
 - o *Experiment* : qui sauvegarde les paramètres de traçabilité d'une expérimentation
- Configuration et lancement du système « *Biogenix-DSS* »

3.7.2 Résultats de la deuxième étape du projet

À la deuxième étape du projet, les résultats ci-dessous ont été obtenus :

1. Test des fonctionnalités des systèmes
2. Extraction des connaissances sur le fonctionnement des systèmes
3. Extraction de la conception du système à partir du code source existant :
 - Extraction du module du sous-système Prognomix-DSS
 - Extraction du module du sous-système *Experiment Manager*
 - Extraction de l'architecture logicielle du système *Biogenix-DSS*
 - Extraction du modèle de traçabilité du système *Experiment Manager*
 - Extraction des copies-écrans du sous-système de traçabilité.
 - Extraction du diagramme des classes de certains modules importants.
4. Amélioration du modèle de rétro-ingénierie
5. Rapport partiel et final du projet

3.8 Améliorations futures

Le projet jusqu'ici a été réalisé avec succès. Des efforts ont été déployés afin d'extraire plus de connaissances possibles sur le système existant. Un modèle théorique de rétro-ingénierie, à haut niveau, a servi de référence à ce projet afin d'aider à identifier certaines activités sur lesquelles il a fallu consacrer des efforts. Y a-t-il une opportunité dans le futur pour une autre équipe ou une autre personne de redémarrer un tel projet en réutilisant ce qui a été fait? Cette question est pertinente et elle suscite l'intérêt de répondre par la démonstration ci-dessous.

Il est évident de comprendre qu'aucune autre équipe ou personne ne pourra redémarrer un projet facilement sans consulter ce qui a été fait précédemment. Il est évident que l'activité de rétro-ingénierie nécessite un minimum des connaissances techniques. Toutefois, avec le travail qui a été effectué dans ce projet, des informations sont disponibles, à une autre personne, qui se donnera le défi de continuer ce projet et de consulter la documentation récupérée ici. Dans ce projet, 53% des efforts ont été consacrés à la recherche et à la compréhension du système. Pour une personne qui doit redémarrer le projet dans le futur, cela ne devrait lui prendre qu'environ 15% d'effort étant donné que les informations sont maintenant disponibles. Un tel pourcentage est estimé car la documentation existante et le modèle de rétro-ingénierie amélioré vont pouvoir aider à se faire rapidement un modèle mental et épargner beaucoup d'effort de recherche. Dans ce projet, 24% d'efforts ont été consacrés à la rétro-ingénierie des codes sources et des bases de données. L'amélioration future de ce système nécessitera la prise en compte de certains autres paramètres, donc beaucoup plus d'effort devra être prévu pour la conception, qui est estimé à 35% d'effort. En résumé, pour redémarrer un projet de rétro-ingénierie sur ce même logiciel, une nette amélioration d'effort serait prévisible, soit 63% de 700 heures, qui donnent un total de 455 heures.

3.9 Conclusion

La rétro-ingénierie est une technique qui est appelée à récupérer les connaissances des systèmes patrimoniaux. Cette technique a permis la redocumentation d'un système existant, l'extraction de sa conception et l'extraction de son architecture matérielle et logicielle. Pour y parvenir, un modèle de référence s'avère utile mais ce projet a permis de le raffiner. Le modèle résultant étant beaucoup plus détaillé aidera à mieux aiguiller le travail d'une personne qui voudrait continuer le travail de rétro-ingénierie. Dans ce chapitre, un tableau comparatif des outils utilisés dans ce projet pourrait servir d'indicateurs de redémarrage du travail. L'appréciation de tout travail est mesurée par son résultat, tel en est l'objectif de la section 3.1.8 de ce document. Il est important d'admettre qu'aucune initiative de ce genre

n'est parfaite dès le départ. En conclusion, la section 3.8 présente des indicateurs qui estiment les efforts qui seront diminués grâce à la récupération d'information lors de cette recherche.

BIBLIOGRAPHIE

- Capron, H., Meeusen, W. The National innovation system of Belgium: *Contributions to economics*. Berlin: Springer-Verlag, 2000, 245 p.
- Refinefocus. What is reverse engineering? [En ligne], (Page consulté le 9 Mars 2015), available at: <http://refineandfocus.com/2012/05/>
- Gilbert-Macmillan, K. Intellectual property law for reverse engineering computer programs in the European Community, *Santa Clara high technology law journal*, 1993, 9 (1): 247-264, available at DigitalCommons: <http://digitalcommons.law.scu.edu/chtlj/vol9/iss1/6/>
- Chikofsky, E.J., Cross, J.H. II. Reverse engineering and design recovery: *A taxonomy, Software, IEEE*, 1990, 7 (1): 13-17, available at IEEEExplore: <http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=43044>
- Samuelson, P., Scotchmer, S. The law and economics of reverse engineering, *Yale law journal*, 2001, 111(1575): 1577-1663, available at: <http://www.yalelawjournal.org/article/the-law-and-economics-of-reverse-engineering>
- White, G. S. Memoir of Samuel Slater: *The father of American manufactures*. Philadelphia, 1836, 491 p. Available at: <https://archive.org/details/memoirsamuelsla02whitgoog>
- Dhillon, B. S. Engineering and technology management tools and applications. Artech House, 2002, 402p.
- Friesinger, G., Herwig, J. The art of reverse engineering. *Transcript-Verlag*, 2014, 300p.
- Garrels, M. Introduction to linux. 2007, 223 p, available at TLDP: <http://tldp.org/LDP/intro-linux/intro-linux.pdf>
- Ulrich, W. M. Legacy systems: transformation strategies. NJ: Prentice Hall ptr upper saddle river, 2002, 300 p. Available at ACM: <http://dl.acm.org/citation.cfm?id=515375>
- Lande, R.H., Sturgis, S. Reverse engineering of computer software and U.S. antitrust policy, *Harvard journal of law and technology*, 1996, 9(2):238-281, available at SSRN: <http://ssrn.com/abstract=1356017>
- Müller, H. A., Tilley, S. R. et al. Understanding software systems using reverse engineering technology perspectives from the Rigi project, *Conference of the centre for advanced studies on collaborative research*, 1993, 1(1): 217-226, available at: ACM: <http://dl.acm.org/citation.cfm?id=962309>

- Byrne, E. J. A Conceptual foundation for software re-engineering, *Software maintenance, proceedings conference*, 1992, 1(1):226 – 235, available at IEEE explore: <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=242539>
- Telea, A. C. Reverse engineering: *Recent advances and applications*. InTech, 2012, 290 p. Available at <http://www.cs.rug.nl/~alex/PAPERS/InTech12/book.pdf>
- Byrne, E. J. Software reverse engineering: *A case study*, *Software — practice and experience*, 1991, 21(12), 1349–1364, available at CiteseerX: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.14.5976&rep=rep1&type=pdf>
- Müller, H. A et Janhke, J. H. et al. Reverse engineering: *A roadmap*, *Conference on the future of software engineering*, 2000, 1(1):47-60, available at ACM: <http://dl.acm.org/citation.cfm?id=336526>
- Systä, T. et al. Static and dynamic reverse engineering techniques for Java software systems, 2000, 232p. Available at CiteseerX: <http://citeseerx.ist.psu.edu/viewdoc/versions?doi=10.1.1.16.9520>
- Dr. Rosenberg, L. H., Hyatt. L. E. et al. Software re-engineering, 1996, 1-31, available at CiteSeerX: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.99.447&rep=rep1&type=pdf>
- Abbas, A. S., Jeberson, W. et al. The need of re-engineering in software engineering international, *Journal of engineering and technology*, 2012, 2(2):292-295, available at Iet-Journals: http://iet-journals.org/archive/2012/feb_vol_2_no_2/9319521326798117.pdf
- Clément, P., Bachmann, F. et al. Documenting software architectures: *Views and beyond*. Addison-Wesley Professional, 2e Edition, 2010, 592 p.
- Humphreys, P. Extending Ourselves: Computational science, empiricism, and scientific method, *Oxford University Press*, Oxford, 2004, 172 p.

ANNEXE I

INVENTAIRES SUR LES MATÉRIELS EXISTANTS

ANNEXE I

[Computer 1: Linux Server](#)
[Computer 2: Francois Desktop](#)
[Computer 3: Prog1 Desktop](#)□

1 Computer 1: Linux Server

Environment

OS: Linux Gentoo R4 (2007)
Kernel: 2.6.21 x86_64

1.1 IP Addresses

eth0: 10.180.121.60
hostname: serverbiogenix

Note: Modified file /etc/conf.d/net to use ÉTS DHCP. Eth0 was previously a static IP: 10.0.10.39 / 255.255.255.0

1.2 Administrator connection

ssh: 10.180.121.60
user: root
pass: sassi.2008.caia

1.3 Hardware

CPU: Core2 6420 @ 2.13 GHz

1.4 Filesystems

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/sda3	290G	160G	116G	58%	/
udev	3.9G	668K	3.9G	1%	/dev
shm	3.9G	0	3.9G	0%	/dev/shm
/dev/sdb1	294G	55G	224G	20%	/mnt/backup
/dev/sda1	54M	9.8M	41M	20%	/boot

1.5 Local users

apache:x:81:81:added by portage for apache:/var/www:/sbin/nologin
postgres:x:1008:1009:./home/postgres:/bin/bash

intégration:x:1007:1008:./home/integration:/bin/bash
share:x:1004:1005:Compte partage:/home/share:/bin/bash

```

brunelle:x:1002:1003:Pierre-Luc Brunelle:/home/brunelle:/bin/bash
gurau:x:1003:1004:Alexandru Gurau:/home/gurau:/bin/bash
letarte:x:1005:1006::/home/letarte:/bin/bash
Gauthier:x:1006:1007::/home/gauthier:/bin/bash
lavoie:x:1009:1010::/home/lavoie:/bin/bash
tlavoie:x:1010:1011::/home/tlavoie:/bin/bash
flebel:x:1011:1012::/home/flebel:/bin/bash

```

Open ports

Protocole	IP	Port	Service	Notes
TCP	0.0.0.0	111	sunrpc	ONC RPC (Sun RPC) ??
TCP	127.0.0.1	631	cups	Printer
TCP	127.0.0.1	5432	postgresql	Database
TCP	0.0.0.0	669		
TCP		22	ssh	Remote connection
UDP	0.0.0.0	32768		
UDP	0.0.0.0	666		
UDP	0.0.0.0	5353	mdns	Multicast DNS (mDNS) ??
UDP	0.0.0.0	111	sunrpc	ONC RPC (Sun RPC) ??
UDP	0.0.0.0	631	cups	Printer
UDP		32769		
UDP		177	xdmcp	X Display Manager Control Protocol (XDMCP)
UDP		5353	mdns	Multicast DNS (mDNS) ??

```

log: integration
password: apres.watson
db: prognomix_test

```

List of databases:

```

List of databases
Name      | Owner  | Encoding
-----+-----+-----
experiment      | postgres | SQL_ASCII
experiment_demo  | Gauthier | SQL_ASCII
experiment_test  | postgres | SQL_ASCII
experiment_test_fci | intégration | SQL_ASCII
experiment_test_fci2 | intégration | SQL_ASCII
fci              | Gauthier | SQL_ASCII
fci_production   | intégration | SQL_ASCII
fci_production_ori | intégration | SQL_ASCII
fci_test         | intégration | SQL_ASCII
fci_test2        | intégration | SQL_ASCII
fci_test3        | intégration | SQL_ASCII

```

gad	Gauthier	SQL_ASCII
mdeie	Gauthier	SQL_ASCII
postgres	postgres	SQL_ASCII
prognomix_experiment	postgres	SQL_ASCII
prognomix_test	intégration	SQL_ASCII
snp_data	Gauthier	SQL_ASCII
template0	postgres	SQL_ASCII
template1	postgres	SQL_ASCII

Computer2: Francois Desktop

My Computer Information - General

Refresh screen

Specifications		Processor	
System manufacturer		Intel(R) Core(TM)2 CPU 6420 @ 2.13GHz	
System Model:	System Product Name	Version:	x86 Family 6 Model 15 Stepping 6
BIOS Version:	American Megatrends Inc. 0405	Speed:	2123 MHz
Operating System		General Computer Info	
Microsoft Windows XP Professional		System Name: BIOGENIX-6C0A18	
Version:	5.1.2600	Domain:	WORKGROUP
Service Pack:	3.0	Time Zone:	Eastern Daylight Time
Location:	C:\WINDOWS	Connection:	Workstation (standalone)
PID:	76487-OEM-0027147-79023	Proxy Server:	None
Hot Fix:	XpsEPSC	IP Address:	10.180.121.69 10.180.121.69 10.180.121.69
		IPX Address:	Not Enabled
Memory (RAM)		Local Disk	
Capacity:	2 GB	Total Capacity: 596.16 GB	
		Sum of Hard Disks: (C: D:)	
			<input type="checkbox"/> Used: 78.47 GB
			<input type="checkbox"/> Free: 517.68 GB

Le contenu du répertoire « My documents »

Repertoire My documents	Informations sur le contenu
Backup_Biogenix	
Biogenix	
Biogenix.7z	
Cygwin	
DGA1031_Report_v.1.3.docx	
Downloads	
Logiciel	
MAGIX Downloads	
MicroOLAP Database Designer for PostgreSQL	
Prognomix-DSS_2.09	
Python-2.5.2	
Visual Studio 2008	
gauthier-serveur64.ppk	
manuel-bgx.pdf	
mydoc.txt	
package	
pgadmin.log	
sess2010.pdf	

C:\Cygwin:

Le système « cygwin » est un environnement Linux utilisées sous « Wwindows » afin de travailler dans un environnement similaire que le serveur.

Contenu du répertoire Cygwin	Détails sur le contenu
Cygwin.bat	
Cygwin.ico	
FCI_DSS	
MarkupSafe-0.23	
MarkupSafe-0.23.tar.gz	
MarkupSafe-0.9.3	
MarkupSafe-0.9.3.tar.gz	
PasteScript-1.7.3	
PasteScript-1.7.3.tar.gz	
Pylons-0.9.7	
Routes-1.10.3	
Routes-1.10.3.tar.gz	
Thumbs.db	
bin	
css	
cygdrive	
dev	
etc	
find.exe.stackdump	
home	
htpd2.exe.stackdump	
lib	
nose-0.11.3	
nose-0.11.3.tar.gz	
proc	
pylons-master	
sbin	
srv	
tmp	
usr	
var	

ANNEXE II

DOCUMENT MENTIONNANT PROGNOXIX

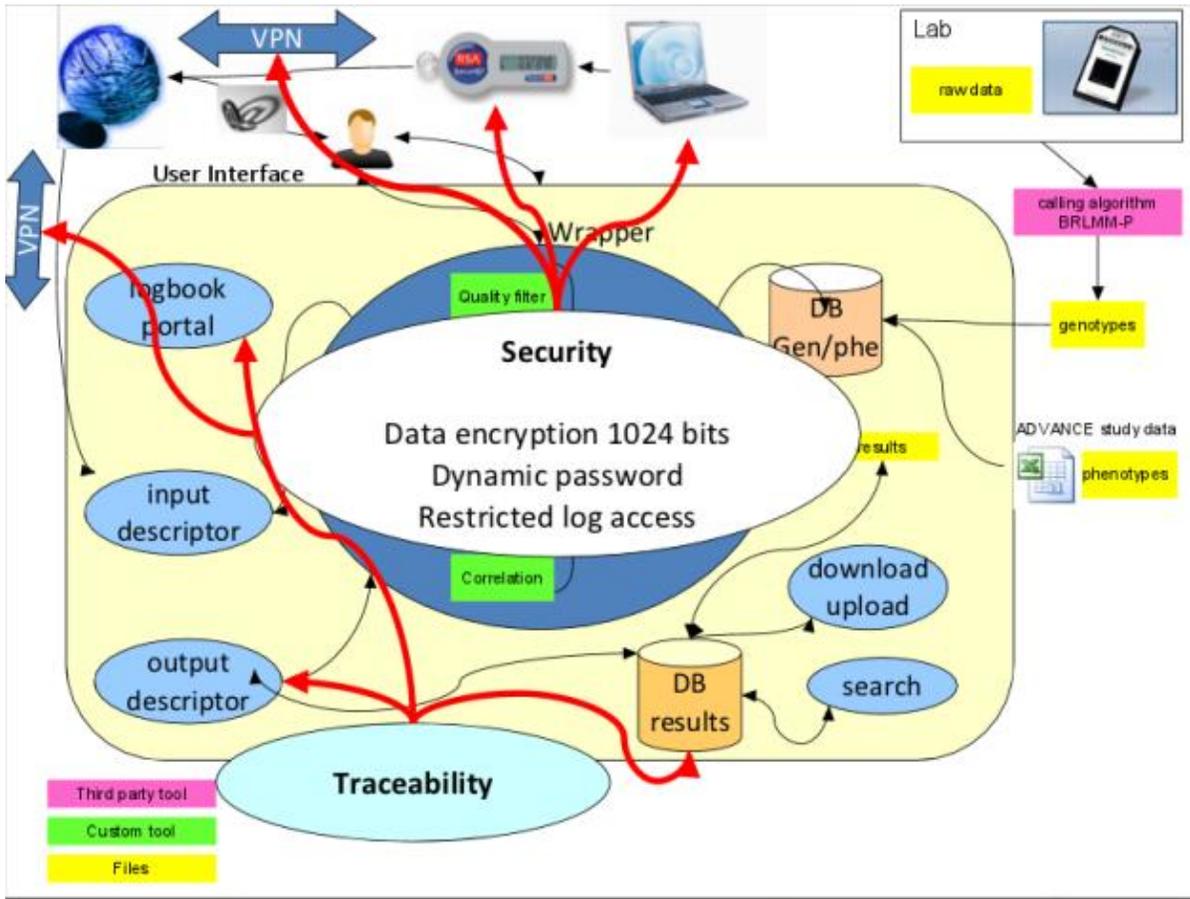
ANNEXE II

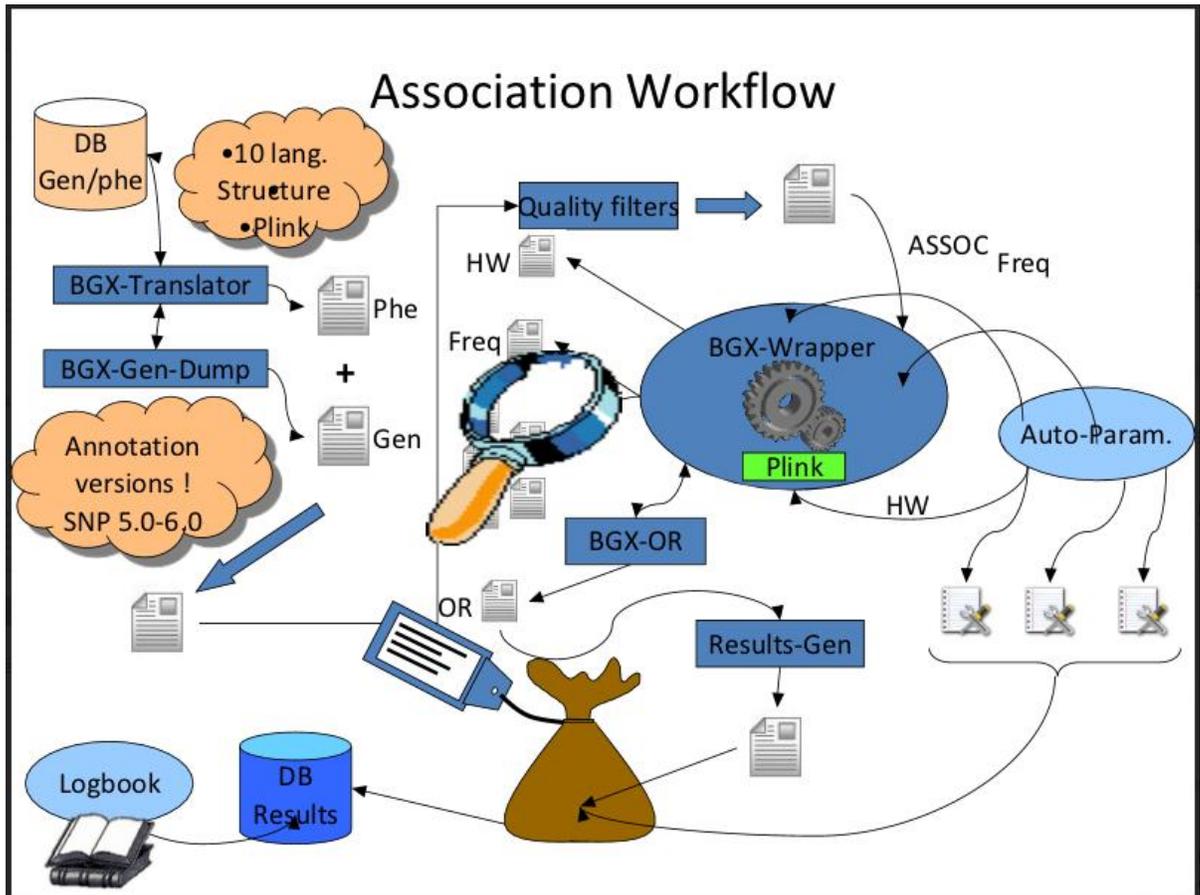
1. Document de présentation prognomix

Prognomix hardware architecture after February 2008

- IBM blade center architecture
- 44 cores (Intel Xeon processors)
- 12 Go memory by server for a total of 64 Go of RAM
- 5 Terabytes of hard disc
- Fully automated backup on tape via IBM Tivoli







ANNEXE III

DOCUMENTS IDENTIFIANTS LES CODES SOURCES

ANNEXE III

1. List des modules

Module 1— Template Descriptor : used to create template of descriptor files

Use cases::

- 1- Creating a template
- 2- Creating a descriptor file based on a template
- 3- Adding a parameter to the database

Actors : User

Module 2- Automatic phenotype Update :

Automatic phenotype update module will allow Prognomix-DSS environment to automatically reflect addition/deletion/rename of a phenotype view in Prognomix database. This module will be added to Prognomix-DSS environment.

Prognomix users therefore won't need to ask Biogenix to add/remove/modify phenotypes in the environment anymore.

External dependencies

This module depends on the existence of a database compliant with PROGNOMIX database version 6 specifications document.

Product function :

This module will allow automatic update of phenotype lists in Prognomix-DSS based on a phenotype table in Prognomix database.

Use Cases:

- 1- Create a new phenotype view in Prognomix database.
- 2- Update phenotype_views table in Prognomix database with the name of the new view.
- 3- Log in Prognomix-DSS
- 4- Click on the association menu on the left column of Prognomix-DSS
- 5- Assert the newly created view name is in the phenotype name list.

Module 3: Automation of PCA calculation

Use cases:

1 General wrapper validity checks

All wrappers shall check that the domain of their inputs correspond to the expected domain.

Wrappers input parameters shall be restricted to the minimal subset of input parameters of the underlying program required to execute the wrapper's task. Wrappers shall detect any conflicting combination of input parameters and raise an exception with an appropriate message when it happens.

2 Extraction of genetic information with dump_genetic.py

This extraction script shall perform the same validity checks as dump_genetic.py, implemented by Pierre-Luc Brunelle.

This extraction script shall handle errors according to errors handling mechanisms implemented by Pierre-Luc Brunelle in Prognomix's genotype python library.

This extraction script shall throw an exception with a significant message when any unhandled error occurs

3 Concatenation of genetic information with plink_cat_geno

This script shall check that both .map and both .plink-info files contain the same information before concatenation of .ped files.

This extraction script shall throw an exception with a significant message when any unhandled error occurs.

4 Extraction of unassociated snps with plink_unassociated_snps.py

This extraction script shall check that the header of its input file is equal to "CHR, SNP, A1, A2, TEST, AFF, UNAFF, CHISQ, DF, P"

This extraction script shall check that the association test name is in "GENO, TREND, ALLELIC, DOM, REC"

This extraction script shall ignore p-values equal to "NA"

This extraction script shall throw an exception with a significant message when any error occurs

5 Extraction of spaced snps with dump_plink_spaced_snps.py

This extraction script shall perform the same validity checks as dump_spaced_snps.py, implemented by Pierre-Luc Brunelle.

The input of this extraction script shall be the output of plink_unassociated_snps.py

If both Affymetrix 5.0 and 6.0 dumps are required, output files must be identical between both dumps.

This extraction script shall handle errors according to errors handling mechanisms implemented by Pierre-Luc Brunelle in Prognomix's genotype python library.

This extraction script shall throw an exception with a significant message when any unhandled error occurs

```

6 Concatenation of genetic information with csv2_cat_geno
This script shall check that the headers of both input files contain the same information, possibly in different order.
This script shall read two csv2 files in input and output a single csv2 file containing header and all the information of both input files.
This extraction script shall throw an exception with a significant message when any unhandled error occurs.

7 Extraction of country with dump_country.py
This extraction script shall check that the header of its input file is of the form (id, affymetrix snp id1, affymetrix snp id2, affymetrix snp id3, etc.)
This extraction script shall handle errors according to errors handling mechanisms implemented by Pierre-Luc Brunelle in Prognomix's genotype python
library.
This extraction script shall throw an exception with a significant message when any unhandled error occurs

8 PCA calculation with pca.py
This extraction script shall perform the same validity checks as pca.py, implemented by Pierre-Luc Brunelle.
The input to this script shall be the output of dump_plink_spaced_snps.py
This extraction script shall handle errors according to errors handling mechanisms implemented by Pierre-Luc Brunelle in Prognomix's genotype python
library.
This extraction script shall throw an exception with a significant message when any unhandled error occurs

9 Scatter plot of PCA with pca_plot.py
This extraction script shall perform the same validity checks as pca_plot.py, implemented by Pierre-Luc Brunelle.
PCA values given as input to this script shall be the output of pca.py
Country names given as input to this script shall be the output of dump_country.py
This extraction script shall handle errors according to errors handling mechanisms implemented by Pierre-Luc Brunelle in Prognomix's genotype python
library.
This extraction script shall throw an exception with a significant message when any unhandled error occurs

Module 4- Descriptive Statistics
Descriptive statistics module will allow end-user to compute descriptive statistics of a phenotype against a set of phenotypes. This module will be added to
BGX-DSS environment.

The main purpose of this module is to provide an overview of a phenotype and its interactions with other phenotypes.
External dependencies
This module requires pids files (See appendix 1) to be stored in a pre-determined folder on Prognomix's cluster.
Scenarios
Executing a descriptive statistics analysis
Click descriptive statistics menu in the left-hand column of BGX-DSS.
Fill out experiment description section.
Select a patient ids file.
elect a base phenotype and a set of comparison phenotypes.
Select an output file name for your analysis.
Click next, verify the parameters in the confirmation page and click done.

```

2. Document de Configuration pour lancer le système.

```

#####
##
## Author :   Francois Gauthier
## Date :    March, 25, 2008
## Application :   Biogenix experiment manager
## Subject :    Installation, general usage and testing of the application
##
#####

```

```

/* Installation of pylons */
See documentation at : http://wiki.pylonshq.com/display/pylonsdocs/Installing+Pylons

```

```
$ easy_install Pylons==0.9.6.1
```

```

If easy_install is not present do yum install python-setuptools in
"Fedora Core 7" or "yum install python-setuptools-devel" in Fedora 8.
/* Getting a local copy of the experiment manager framework */

```

Copy the entire workflow folder on your computer. The folder is available on SVN in the folder :
 serveurur64/opt/SVN/workflow

/* Ensuring the application is up-to-date */

Go to workflow/biogenix and type :

```
$ python setup.py develop
```

/* Patch the PooledDB.py file */

In DBUtils 0.9.4 there is minor bug when a script is shutting down when PooledDB object is getting cleaned up (this happens while testing)

You have to replace the `__del__` method of PooledDB class in the file PooledDB.py.

```
def __del__(self):
    """Delete the pool."""
    if hasattr(self, '_connections'):
        self.close()
```

Becomes

```
def __del__(self):
    """Delete the pool."""
    if hasattr(self, '_connections') and self._connections > 0:
        self.close()
```

See <http://www.mail-archive.com/webware-devel@lists.sourceforge.net/msg02636.html> for further details

/* Create and populate the database */

We use the postgreSQL database system available at : <http://www.postgresql.org/>

Database creation and population scripts can be found in workflow/biogenix/sql

/* Start the web server and use the application */

```
$ /workflow/biogenix/start.sh
```

You should be able to use the experiment manager by pointing your browser at <http://localhost:5000/mana>

/* Testing requirements */

Requirements :

- A test database named `experiment_test` with the same structure as the real database (see `workflow/biogenix/sql/experiment_tables.sql`)
- Selenium remote control server 0.9.2, available at : <http://selenium.openqa.org>.
- The file 'selenium-server.jar' in a folder included in the `$CLASSPATH` environment variable.
- The file 'selenium.py' in a folder included in the `$PYTHONPATH` environment variable.

/* Testing the application */

```
$ /workflow/biogenix/test.sh
```

A file named `coverage-summary.xml` will be produced in the folder `workflow/biogenix`. This file can be viewed in a web browser and provides links to annotated/coloured python source files.

IMPORTANT NOTE : I have deactivated the coverage report (I only need the `.coverage` file) by commenting out line 102 in

/usr/lib/python2.5/site-packages/nose-0.10.1-py2.5.egg/nose/plugins/cover.py

3. Listes des répertoires contenant des codes sources sur le serveur

Voici le listing de mes répertoires de travail.

Tous les dossiers data/ ne seront pas décrits. Ils reviennent dans tous les environnements pylons/apache et servent de données temporaires (un peu comme la cache).

LEGENDE	
Important	Toujours utiliser. Ne pas supprimer.
Archive	J'en conserve une copie sur mon ordinateur pour références.
Script_Amélie	Script ou outil développé par Amélie pour sauver du temps...
Gen	Fichier généré automatiquement
Docu	Contiens de la documentation

Nom du fichier	Role	Description
./pgpass	Important	Ce fichier contient les informations de sécurités pour se connecter à la bd PostgreSQL. Voir documentation au besoin
./psql_history	Important	Voir doc PostgreSQL
./python-eggs/	Important	Voir doc PostgreSQL
./sqlite_history	Important	Voir doc sqlite
./ssh/	Important	Contiens des fichiers de sécurité pour établir des connexions vers des serveurs (bgx ou Prognomix)
./subversion/config	Important	Fichiers pour gérer SVN
./Xauthority	??	À toujours été sur mon ordinateur. Dois avoir un lien avec Xterm
./bin/biogenix/	Important	Contiens des outils développés chez Biogenix
./bin/external/	Important	Contiens des outils externes utilisés dans nos programmes
./bin/prognomix/	Important	Contiens des outils développés pour Prognomix qu'on a mis dans leur environnement.
./create_table_rat_mouse_database.sql	Archive	Une des premières versions de la base de données du FCI
./eclipse-jee-ganymede-SR1-linux-gtk-x86_64.tar.gz.tar	Archive	Fichier d'installation d'Éclipse
./fci_env/	Important	L'environnement utilisé pour développer les outils du FCI
./fci_env/bin/activate	Important	Voir doc virtualenv.py
./fci_env/bin/activate_this.py	Important	Voir doc virtualenv.py
./fci_env/bin/coverage	Important	Voir doc virtualenv.py
./fci_env/bin/easy_install	Important	Voir doc virtualenv.py
./fci_env/bin/easy_install-2.5	Important	Voir doc virtualenv.py
./fci_env/bin/mako-render	Important	Voir doc virtualenv.py
./fci_env/bin/nosetest	Important	Voir doc virtualenv.py
./fci_env/bin/nosetest-2.5	Important	Voir doc virtualenv.py
./fci_env/bin/paster	Important	Voir doc virtualenv.py

./fci_env/bin/pygmentize	Important	Voir doc virtualenv.py
./fci_env/bin/python.exe	Important	Voir doc virtualenv.py
./fci_env/bin/testall.sh	Important	Voir doc virtualenv.py
./fci_env/bin/to3.sh	Important	Voir doc virtualenv.py
./fci_env/FCI_releases/.project	Important	Fichier nécessaire à Eclipse
./fci_env/FCI_releases/.pydevproject	Important	Fichier nécessaire à Éclipse
./fci_env/FCI_releases/FCI/coverage-summary.xml	Gen	Ce fichier est généré automatiquement lorsqu'on roule les tests.
./fci_env/FCI_releases/FCI/development.ini	Important	Fichier de configuration utilisé par pylons/apache. Voir doc pylons.
./fci_env/FCI_releases/FCI/docs/index.txt	Gen	Fichier généré par pylons. Voir doc pylons.
./fci_env/FCI_releases/FCI/docs/selenium	Docu	Documentation brute utilisée pour monter le manuel d'instruction pour le FCI
		Tous les *.docu = morceau de documentation qui ont été assemblés pour créer le manuel d'instruction
		Tous les scripts "cat_docu" s'occupaient d'assembler les "sous" morceaux de documentation en "morceaux".
		Les fichiers "purpose" contiennent un paragraphe descriptif de la section de documentation dans laquelle ils se trouvent.
./fci_env/FCI_releases/FCI/docs/selenium/total_docu	Script_Amélie	Assemble les morceaux de documentation provenant de chaque dossier.
./fci_env/FCI_releases/FCI/docs/sql/bd_creator	Script_Amélie	À exécuter pour régénérer la base de données du FCI. Donne une liste des erreurs à la fin.
./fci_env/FCI_releases/FCI/docs/sql/constraints.sql	SQL	Contiens les contraintes de la bd FCI tel qu'extrait du schéma Visio.
./fci_env/FCI_releases/FCI/docs/sql/create_deng_m_tables.sql	SQL	Script SQL permettant de créer les tables de bd d'un chercheur précis. Le _m indique que ce sont les tables pour les souris (mouse)
./fci_env/FCI_releases/FCI/docs/sql/create_deng_r_tables.sql	SQL	Script SQL permettant de créer les tables de bd d'un chercheur précis. Le _m indique que ce sont les tables pour les rat
./fci_env/FCI_releases/FCI/docs/sql/create_fci_database.sh	Archive	À exécuter pour régénérer la base de données du FCI.
./fci_env/FCI_releases/FCI/docs/sql/create_fci_database.sql	Archive	À exécuter pour régénérer la base de données du FCI.
./fci_env/FCI_releases/FCI/docs/sql/create_fci_database3.sh	Script_Amélie	À exécuter pour régénérer la base de données du FCI.
./fci_env/FCI_releases/FCI/docs/sql/create_jlavoie_m_tables.sql	SQL	Script SQL permettant de créer les tables de bd d'un chercheur précis. Le _m indique que ce sont les tables pour les souris (mouse)
./fci_env/FCI_releases/FCI/docs/sql/create_jlavoie_r_tables.sql	SQL	Script SQL permettant de créer les tables de bd d'un chercheur précis. Le _m indique que ce sont les tables pour les rat
./fci_env/FCI_releases/FCI/docs/sql/create_jtremblay_m_tables.sql	SQL	Script SQL permettant de créer les tables de bd d'un chercheur précis. Le _m indique que ce sont les tables pour les souris (mouse)
./fci_env/FCI_releases/FCI/docs/sql/create_jtremblay_r_tables.sql	SQL	Script SQL permettant de créer les tables de bd d'un chercheur précis. Le _m indique que ce sont les tables pour les rat
./fci_env/FCI_releases/FCI/docs/sql/create_phamet_m_tables.sql	SQL	Script SQL permettant de créer les tables de bd d'un chercheur précis. Le _m indique que ce sont les tables pour les souris (mouse)

./fci_env/FCI_releases/FCI/docs/sql/creat e_phamet_r_tables.sql	SQL	Script SQL permettant de créer les tables de bd d'un chercheur précis. Le _m indique que ce sont les tables pour les rat
./fci_env/FCI_releases/FCI/docs/sql/creat e_public_tables.sql	SQL	Script SQL permettant de créer les tables publiques du FCI
./fci_env/FCI_releases/FCI/docs/sql/creat e_table_rat_mouse_database.sql	Archive	Ancien script de création de bd pour le fci
./fci_env/FCI_releases/FCI/docs/sql/creat e_views.sql	SQL	Script SQL permettant de créer les vues pour la section "Search" pour le FCI
./fci_env/FCI_releases/FCI/docs/sql/experi ment_data.sql	SQL	Script SQL pour insérer les données de base pour la bd d'expérience (coté Biogenix)
./fci_env/FCI_releases/FCI/docs/sql/experi ment_tables.sql	SQL	Script SQL pour créer la bd d'expérience (coté Biogenix)
./fci_env/FCI_releases/FCI/docs/sql/experi ment_test_data.sql	SQL	Script SQL pour insérer les données de base pour la bd d'expérience (coté Biogenix)
./fci_env/FCI_releases/FCI/docs/sql/gener ate_creation_script.py	Script	Permet de générer le script SQL de création de tables pour chaque chercheur
./fci_env/FCI_releases/FCI/docs/sql/init_fc i.sql	SQL	Script SQL pour initialiser la bd
./fci_env/FCI_releases/FCI/docs/sql/Make file	-	Fichier appelé lors de la création de la bd.
./fci_env/FCI_releases/FCI/docs/sql/popul ate_dengm_tables.sql	SQL	Script SQL permettant d'insérer des données (training) dans la bd d'un chercheur précis. Le _m indique que ce sont les tables pour les souris (mouse)
./fci_env/FCI_releases/FCI/docs/sql/popul ate_deng_m_tables.sql	SQL	Script SQL permettant d'insérer des données (training) dans la bd d'un chercheur précis. Le _m indique que ce sont les tables pour les souris (mouse)
./fci_env/FCI_releases/FCI/docs/sql/popul ate_deng_r_tables.sql	SQL	Script SQL permettant d'insérer des données (training) dans la bd d'un chercheur précis. Le _r indique que ce sont les tables pour les rats
./fci_env/FCI_releases/FCI/docs/sql/popul ate_fci.py	Script	Permet de générer le script SQL d'insertion de données dans les tables pour chaque chercheur
./fci_env/FCI_releases/FCI/docs/sql/popul ate_lavoie_m_tables.sql	SQL	Script SQL permettant d'insérer des données (training) dans la bd d'un chercheur précis. Le _m indique que ce sont les tables pour les souris (mouse)
./fci_env/FCI_releases/FCI/docs/sql/popul ate_lavoie_r_tables.sql	SQL	Script SQL permettant d'insérer des données (training) dans la bd d'un chercheur précis. Le _r indique que ce sont les tables pour les rats
./fci_env/FCI_releases/FCI/docs/sql/popul ate_public_m_tables.sql	SQL	Script SQL permettant d'insérer des données (training) dans le schéma public. Le _m indique que ce sont les tables pour les souris (mouse)
./fci_env/FCI_releases/FCI/docs/sql/popul ate_public_r_tables.sql	SQL	Script SQL permettant d'insérer des données (training) dans le schéma public. Le _r indique que ce sont les tables pour les rat
./fci_env/FCI_releases/FCI/docs/sql/popul ate_tremblay_m_tables.sql	SQL	Script SQL permettant d'insérer des données (training) dans le schéma public. Le _m indique que ce sont les tables pour les souris (mouse)
./fci_env/FCI_releases/FCI/docs/sql/popul ate_tremblay_r_tables.sql	SQL	Script SQL permettant d'insérer des données (training) dans le schéma public. Le _r indique que ce sont les tables pour les rat
./fci_env/FCI_releases/FCI/docs/sql/script schema_formation.sql	SQL	Script permettant de créer la bd finale pour le FCI
./fci_env/FCI_releases/FCI/docs/sql/tables /Lavoie/	SQL	Contiens des scripts sql d'insertion de données pour un chercheur précis.
./fci_env/FCI_releases/FCI/docs/sql/tables	SQL	Contiens des scripts sql d'insertion de données pour un

/MiceGenotypes.sql		chercheur précis.
./fci_env/FCI_releases/FCI/docs/sql/tables/public/	SQL	Contiens des scripts sql d'insertion de données pour la table public.
./fci_env/FCI_releases/FCI/docs/sql/tables/Tremblay/	SQL	Contiens des scripts sql d'insertion de données pour un chercheur précis.
./fci_env/FCI_releases/FCI/fci/config/environment.py	Important	Voir doc pylons
./fci_env/FCI_releases/FCI/fci/config/environment.pyc	Important	Voir doc pylons
./fci_env/FCI_releases/FCI/fci/config/middleware.py	Important	Voir doc pylons
./fci_env/FCI_releases/FCI/fci/config/middleware.pyc	Important	Voir doc pylons
./fci_env/FCI_releases/FCI/fci/config/routing.py	Important	Voir doc pylons
./fci_env/FCI_releases/FCI/fci/config/routing.pyc	Important	Voir doc pylons
./fci_env/FCI_releases/FCI/fci/config/__init__.py	Important	Voir doc pylons
./fci_env/FCI_releases/FCI/fci/config/__init__.pyc	Important	Voir doc pylons
./fci_env/FCI_releases/FCI/fci/controllers/admin.py	Important	Contrôle la section administration du FCI
./fci_env/FCI_releases/FCI/fci/controllers/admin.pyc	Important	Contrôle la section administration du FCI
./fci_env/FCI_releases/FCI/fci/controllers/analysis.py	Important	Contrôle la section Experiments du FCI
./fci_env/FCI_releases/FCI/fci/controllers/analysis.pyc	Important	Contrôle la section Experiments du FCI
./fci_env/FCI_releases/FCI/fci/controllers/auth.py	Important	Contrôle la section authentification du FCI (login logout)
./fci_env/FCI_releases/FCI/fci/controllers/auth.pyc	Important	Contrôle la section authentification du FCI (login logout)
./fci_env/FCI_releases/FCI/fci/controllers/compare.py	Important	Contrôle la partie "compare" du FCI (voir logbook)
./fci_env/FCI_releases/FCI/fci/controllers/compare.pyc	Important	Contrôle la partie "compare" du FCI (voir logbook)
./fci_env/FCI_releases/FCI/fci/controllers/data_extraction.py	Important	Contrôle la section Data Extraction du FCI
./fci_env/FCI_releases/FCI/fci/controllers/data_extraction.pyc	Important	Contrôle la section Data Extraction du FCI
./fci_env/FCI_releases/FCI/fci/controllers/error.py	Important	Voir doc pylons
./fci_env/FCI_releases/FCI/fci/controllers/error.pyc	Important	Voir doc pylons
./fci_env/FCI_releases/FCI/fci/controllers/main.py	Important	Contrôle les sections de base du FCI (logbook, template descriptor, ...)
./fci_env/FCI_releases/FCI/fci/controllers/main.pyc	Important	Contrôle les sections de base du FCI (logbook, template descriptor, ...)
./fci_env/FCI_releases/FCI/fci/controllers/manage_mice.py	Important	Contrôle la section Animal files du FCI
./fci_env/FCI_releases/FCI/fci/controllers/manage_mice.pyc	Important	Contrôle la section Animal files du FCI
./fci_env/FCI_releases/FCI/fci/controllers/	Important	Contrôle la section Reproduction du FCI

manage_reproductions.py		
./fci_env/FCI_releases/FCI/fci/controllers/ manage_reproductions.pyc	Important	Contrôle la section Reproduction du FCI
./fci_env/FCI_releases/FCI/fci/controllers/ procedures.py	Important	Contrôle la section Procedures du FCI
./fci_env/FCI_releases/FCI/fci/controllers/ procedures.pyc	Important	Contrôle la section Procedures du FCI
./fci_env/FCI_releases/FCI/fci/controllers/ samples.py	Important	Contrôle la section Samples du FCI
./fci_env/FCI_releases/FCI/fci/controllers/ samples.pyc	Important	Contrôle la section Samples du FCI
./fci_env/FCI_releases/FCI/fci/controllers/ search.py	Important	Contrôle la section Search du FCI
./fci_env/FCI_releases/FCI/fci/controllers/ search.pyc	Important	Contrôle la section Search du FCI
./fci_env/FCI_releases/FCI/fci/controllers/ telemetry.py	Important	Contrôle la section Telemetry du FCI
./fci_env/FCI_releases/FCI/fci/controllers/ telemetry.pyc	Important	Contrôle la section Telemetry du FCI
./fci_env/FCI_releases/FCI/fci/controllers/ template.py	Important	Voir doc pylons
./fci_env/FCI_releases/FCI/fci/controllers/ template.pyc	Important	Voir doc pylons
./fci_env/FCI_releases/FCI/fci/controllers/ __init__.py	Important	Voir doc pylons
./fci_env/FCI_releases/FCI/fci/controllers/ __init__.pyc	Important	Voir doc pylons
./fci_env/FCI_releases/FCI/fci/extlib/	Important	Librairies externes utilisés par l'applications
./fci_env/FCI_releases/FCI/fci/lib/admin_q ueries.py	Important	Contient les requêtes SQL section Administration
./fci_env/FCI_releases/FCI/fci/lib/analysis _queries.py	Important	Contient les requêtes SQL section Experiments
./fci_env/FCI_releases/FCI/fci/lib/app_glo bals.py	Important	Voir doc pylons (contient entre autres les droits des utilisateurs)
./fci_env/FCI_releases/FCI/fci/lib/app_glo bals.pyc	Important	Voir doc pylons (contient entre autres les droits des utilisateurs)
./fci_env/FCI_releases/FCI/fci/lib/app_glo bals2.py	Important	Voir doc pylons (contiens entre autres les droits des utilisateurs)
./fci_env/FCI_releases/FCI/fci/lib/base.py	Important	Voir doc pylons
./fci_env/FCI_releases/FCI/fci/lib/base.pyc	Important	Voir doc pylons
./fci_env/FCI_releases/FCI/fci/lib/const.py	Important	Permet de définir des variables constantes dans Pylons
./fci_env/FCI_releases/FCI/fci/lib/const.py c	Important	Permet de définir des variables constantes dans Pylons
./fci_env/FCI_releases/FCI/fci/lib/db.py	Important	Permet d'établir une connexion vers la bd
./fci_env/FCI_releases/FCI/fci/lib/db.pyc	Important	Permet d'établir une connexion vers la bd
./fci_env/FCI_releases/FCI/fci/lib/descript or_manager/	Important	Module des descriptors (crée des descriptors selon le type d'analyse)
./fci_env/FCI_releases/FCI/fci/lib/dictDeco de.py	Important	Voir doc Python
./fci_env/FCI_releases/FCI/fci/lib/dictDeco de.pyc	Important	Voir doc Python
./fci_env/FCI_releases/FCI/fci/lib/diets_qu	Important	Contient les requêtes SQL section Diets

eries.py		
./fci_env/FCI_releases/FCI/fci/lib/diets_queries.pyc	Important	Contient les requêtes SQL section Diets
./fci_env/FCI_releases/FCI/fci/lib/experiment.py	Important	Contient les requêtes SQL utilisées aux débuts de
./fci_env/FCI_releases/FCI/fci/lib/experiment.pyc	Important	Contient les requêtes SQL section Diets
./fci_env/FCI_releases/FCI/fci/lib/fci_constants.py	Important	Constantes pour le FCI
./fci_env/FCI_releases/FCI/fci/lib/fci_constants.pyc	Important	Constantes pour le FCI
./fci_env/FCI_releases/FCI/fci/lib/fci_queries.py	Important	Contiens des requêtes SQL pour le FCI (général)
./fci_env/FCI_releases/FCI/fci/lib/fci_queries.pyc	Important	Contiens des requêtes SQL pour le FCI (général)
./fci_env/FCI_releases/FCI/fci/lib/helpers.py	Important	Voir doc python
./fci_env/FCI_releases/FCI/fci/lib/helpers.pyc	Important	Voir doc python
./fci_env/FCI_releases/FCI/fci/lib/mydiff.py	Important	Utilisé pour comparer les fichiers
./fci_env/FCI_releases/FCI/fci/lib/mydiff.pyc	Important	Utilisé pour comparer les fichiers
./fci_env/FCI_releases/FCI/fci/lib/pageParser.py	Important	Pour la sécurité (accès des utilisateurs)
./fci_env/FCI_releases/FCI/fci/lib/pageParser.pyc	Important	Pour la sécurité (accès des utilisateurs)
./fci_env/FCI_releases/FCI/fci/lib/parents_queries.py	Archive	N'est plus utilisé
./fci_env/FCI_releases/FCI/fci/lib/parents_queries.pyc	Archive	N'est plus utilisé
./fci_env/FCI_releases/FCI/fci/lib/procedures_queries.py	Important	Contient les requêtes SQL section Procedures
./fci_env/FCI_releases/FCI/fci/lib/procedures_queries.pyc	Important	Contient les requêtes SQL section Procedures
./fci_env/FCI_releases/FCI/fci/lib/protocols_queries.py	Important	Contient les requêtes SQL section Protocoles
./fci_env/FCI_releases/FCI/fci/lib/protocols_queries.pyc	Important	Contient les requêtes SQL section Protocoles
./fci_env/FCI_releases/FCI/fci/lib/sample_queries.py	Important	Contient les requêtes SQL section Samples
./fci_env/FCI_releases/FCI/fci/lib/sample_queries.pyc	Important	Contient les requêtes SQL section Samples
./fci_env/FCI_releases/FCI/fci/lib/telemetry_queries.py	Important	Contient les requêtes SQL section Telemetry
./fci_env/FCI_releases/FCI/fci/lib/telemetry_queries.pyc	Important	Contient les requêtes SQL section Telemetry
./fci_env/FCI_releases/FCI/fci/lib/utils.py	Important	Contiens des outils généraux pour le projet du FCI
./fci_env/FCI_releases/FCI/fci/lib/utils.pyc	Important	Contiens des outils généraux pour le projet du FCI
./fci_env/FCI_releases/FCI/fci/lib/__init__.py	Important	Voir doc python
./fci_env/FCI_releases/FCI/fci/lib/__init__.pyc	Important	Voir doc python

./fci_env/FCI_releases/FCI/fci/model/adminSchemas.py	Important	Validation
./fci_env/FCI_releases/FCI/fci/model/adminSchemas.pyc	Important	Validation
./fci_env/FCI_releases/FCI/fci/model/mySchemas.py	Important	Validation
./fci_env/FCI_releases/FCI/fci/model/mySchemas.pyc	Important	Validation
./fci_env/FCI_releases/FCI/fci/model/myValidators.py	Important	Validation
./fci_env/FCI_releases/FCI/fci/model/myValidators.pyc	Important	Validation
./fci_env/FCI_releases/FCI/fci/model/sample_information.py	Important	Validation - contiens des données critiques sur les échantillons (familles d'organes ou de fluides)
./fci_env/FCI_releases/FCI/fci/model/sample_information.pyc	Important	Validation - contiens des données critiques sur les échantillons (familles d'organes ou de fluides)
./fci_env/FCI_releases/FCI/fci/model/validation_queries.py	Important	Validation - contiens des requêtes pour la validation
./fci_env/FCI_releases/FCI/fci/model/validation_queries.pyc	Important	Validation - contiens des requêtes pour la validation
./fci_env/FCI_releases/FCI/fci/model/__init__.py	Important	Voir doc python
./fci_env/FCI_releases/FCI/fci/model/__init__.pyc	Important	Voir doc python
./fci_env/FCI_releases/FCI/fci/modules/bin/dataQuest.py	Important	module pour importer les données de télémétrie
./fci_env/FCI_releases/FCI/fci/modules/bin/dataQuest.pyc	Important	module pour importer les données de télémétrie
./fci_env/FCI_releases/FCI/fci/modules/bin/dataQuest_wrapper.py	Important	module pour importer les données de télémétrie
./fci_env/FCI_releases/FCI/fci/modules/bin/dataQuest_wrapper.pyc	Important	module pour importer les données de télémétrie
./fci_env/FCI_releases/FCI/fci/modules/bin/__init__.py	Important	Voir doc python
./fci_env/FCI_releases/FCI/fci/modules/bin/__init__.pyc	Important	Voir doc python
./fci_env/FCI_releases/FCI/fci/modules/lib/domain_checker.py	Important	Voir François
./fci_env/FCI_releases/FCI/fci/modules/lib/domain_checker.pyc	Important	Voir François
./fci_env/FCI_releases/FCI/fci/modules/lib/launchers.py	Important	Voir François
./fci_env/FCI_releases/FCI/fci/modules/lib/wrappers.py	Important	Voir François
./fci_env/FCI_releases/FCI/fci/modules/lib/__init__.py	Important	Voir doc python
./fci_env/FCI_releases/FCI/fci/modules/lib/__init__.pyc	Important	Voir doc python
./fci_env/FCI_releases/FCI/fci/modules/__init__.py	Important	Voir doc python
./fci_env/FCI_releases/FCI/fci/modules/__init__.pyc	Important	Voir doc python
./fci_env/FCI_releases/FCI/fci/public/Administration/	Important	C'est dans ce dossier qu'on retrouve les fichiers (ou dossier) qui apparaîtront dans la section Documents du FCI

./fci_env/FCI_releases/FCI/fci/public/	Important	*.jpg = Images utilisées pour le contenu web de l'environnement.
		*.gif = Images utilisées pour le contenu web de l'environnement.
		*.css = Stylistation du contenu web.
./fci_env/FCI_releases/FCI/fci/public/download/	Important ?	Ce dossier contient des fichiers générés pour le téléchargement. Je ne suis pas sûre si on l'utilise encore par contre.
./fci_env/FCI_releases/FCI/fci/public/ExperimentsDB/	Important	Ce dossier contient des expériences qui peuvent être téléversées vers l'environnement.
./fci_env/FCI_releases/FCI/fci/public/html/	Important	Contiens des fichiers html nécessaires pour l'affichage de l'environnement.
./fci_env/FCI_releases/FCI/fci/public/icons/	Important	Contiens quelques icônes nécessaires pour l'affichage de l'environnement.
./fci_env/FCI_releases/FCI/fci/public/images/	Important	Contiens des images nécessaires pour l'affichage de l'environnement.
./fci_env/FCI_releases/FCI/fci/public/javacript/	Important	Contiens tous les scripts Java nécessaires à l'environnement
./fci_env/FCI_releases/FCI/fci/public/tree/	Important	Contiens des images nécessaires pour l'affichage de l'environnement.
./fci_env/FCI_releases/FCI/fci/public/uploaded_images/	Important	Contiens les images téléversées (section immuno et histo)
./fci_env/FCI_releases/FCI/fci/templates/admin/	Important	Contiens les gabarits pour la section Administration
./fci_env/FCI_releases/FCI/fci/templates/analyses/	Important	Contiens les gabarits pour la section Experiments
./fci_env/FCI_releases/FCI/fci/templates/compare/	Important	Contiens les gabarits pour la section Compare (voir logbook)
./fci_env/FCI_releases/FCI/fci/templates/data_extraction/	important	Contiens les gabarits pour la section Data Extraction
./fci_env/FCI_releases/FCI/fci/templates/main/	important	Contiens plusieurs gabarits entre autres ceux du Logbook, des gabarits, de l'administration ...
./fci_env/FCI_releases/FCI/fci/templates/main/business.mako	Important	Ce gabarit définit la base de l'environnement. Entre autres le menu à gauche
./fci_env/FCI_releases/FCI/fci/templates/manage_mice/	Important	Contient les gabarits pour la section Animals files (génotypes, souris, protocole, diètes...)
./fci_env/FCI_releases/FCI/fci/templates/manage_reproductions/	Important	Contiens les gabarits pour la section Reproduction
./fci_env/FCI_releases/FCI/fci/templates/procedures/	Important	Contiens les gabarits pour la section Events et Procedures
./fci_env/FCI_releases/FCI/fci/templates/samples/	Important	Contiens les gabarits pour la section Samples
./fci_env/FCI_releases/FCI/fci/templates/search/	Important	Contiens les gabarits pour la section Search
./fci_env/FCI_releases/FCI/fci/templates/telemetry/	Important	Contiens les gabarits de la section Telemetry
./fci_env/FCI_releases/FCI/fci/tests/	Important	Contiens tous les tests pour l'environnement.
./fci_env/FCI_releases/FCI/fci/tests/cover/	Important	Section pour les tests de couvertures. Générés automatiquement.
./fci_env/FCI_releases/FCI/fci/tests/data/	Important	Contiens quelques fichiers utilisés dans les tests.
./fci_env/FCI_releases/FCI/fci/tests/database_connection.py	Important	Les tests par scripts ont besoin d'une connexion à la bd pour insérer les données de tests.
./fci_env/FCI_releases/FCI/fci/tests/database_connection.pyc	Important	Les tests par scripts ont besoin d'une connexion à la bd pour insérer les données de tests.

./fci_env/FCI_releases/FCI/fci/tests/fonctional/	Important	Contiens tous les tests scriptés (à l'aide de Selenium).
./fci_env/FCI_releases/FCI/fci/tests/selenium/	Important	Contiens tous les tests Selenium (firefox) et plusieurs captures d'écran (*.png)
./fci_env/FCI_releases/FCI/fci/tests/selenium/xml/test_suite	Important	Permet d'ouvrir tous les tests dans Selenium (open test suite)
./fci_env/FCI_releases/FCI/fci/tests/selenium.py	Important	Voir François
./fci_env/FCI_releases/FCI/fci/tests/selenium.pyc	Important	Voir François
./fci_env/FCI_releases/FCI/fci/tests/test_models.py	Important	Voir doc python
./fci_env/FCI_releases/FCI/fci/tests/test_models.pyc	Important	Voir doc python
./fci_env/FCI_releases/FCI/fci/tests/transition.log	Important	Voir François
./fci_env/FCI_releases/FCI/fci/tests/__init__.py	Important	Voir doc python
./fci_env/FCI_releases/FCI/fci/tests/__init__.pyc	Important	Voir doc python
./fci_env/FCI_releases/FCI/fci/websetup.py	Important	Voir doc python
./fci_env/FCI_releases/FCI/fci/websetup.pyc	Important	Voir doc python
./fci_env/FCI_releases/FCI/fci/__init__.py	Important	Voir doc python
./fci_env/FCI_releases/FCI/fci/__init__.pyc	Important	Voir doc python
./fci_env/FCI_releases/FCI/FCI.egg-info/	Important	Voir doc python
./fci_env/FCI_releases/FCI/MANIFEST.in	Important	Voir doc python
./fci_env/FCI_releases/FCI/README.txt	Doc	Voir doc python
./fci_env/FCI_releases/FCI/setup.cfg	Important	Voir doc python
./fci_env/FCI_releases/FCI/setup.py	Important	Voir doc python
./fci_env/FCI_releases/FCI/setup.pyc	Important	Voir doc python
./fci_env/FCI_releases/FCI/start.sh	Script_Amélie	Permet de lancer l'environnement avec Paster.
./fci_env/FCI_releases/FCI/template-coverage.xsl	??	??
./fci_env/FCI_releases/FCI/test.ini	Important	Voir doc python
./fci_env/FCI_releases/FCI/test.sh	Script_Amélie	Permet de lancer les tests (nosetest)
./fci_env/FCI_releases/steps	Script_Amélie	Script utilisé pour les livraisons au CHUM. Exporte vers un dossier spécifié, enlève les .py. FAIRE TRÈS ATTENTION AVANT DE L'EXÉCUTER...
./fci_env/lib/python2.5/	Important	Voir doc python
./jre-6u11-linux-x64.bin	binaire	Java
./lib/biogenix/domain_checker.py	Important	Outils biogenix (voir François)
./lib/biogenix/domain_checker.pyc	Important	Outils biogenix (voir François)
./lib/biogenix/launchers.py	Important	Outils biogenix (voir François)
./lib/biogenix/launchers.pyc	Important	Outils biogenix (voir François)
./lib/biogenix/wrappers.py	Important	Outils biogenix (voir François)
./lib/biogenix/wrappers.pyc	Important	Outils biogenix (voir François)

./lib/pgdb.py	Important	Voir doc python (accès à la bd)
./lib/pgdb.pyc	Important	Voir doc python (accès à la bd)
./lib/prognomix/	Important	Outils pour l'environnement de Prognomix.
./Mail/inbox/.xmhcache	??	À toujours été sur mon ordinateur.
./org.python.pydev.feature_1.4.2.jar	Archive	Pydev (version téléchargée)
./postgresql_autodoc/ChangeLog	??	À toujours été sur mon ordinateur.
./postgresql_autodoc/dia.tpl	??	À toujours été sur mon ordinateur.
./postgresql_autodoc/dot.tpl	??	À toujours été sur mon ordinateur.
./postgresql_autodoc/html.tpl	??	À toujours été sur mon ordinateur.
./postgresql_autodoc/Makefile	??	À toujours été sur mon ordinateur.
./postgresql_autodoc/neato.tpl	??	À toujours été sur mon ordinateur.
./postgresql_autodoc/postgresql_autodoc.pl	??	À toujours été sur mon ordinateur.
./postgresql_autodoc/xml.tpl	??	À toujours été sur mon ordinateur.
./postgresql_autodoc/zigzag.dia.tpl	??	À toujours été sur mon ordinateur.
./PyGreSQL-4.0	Important	Dossier de PygreSQL
./PylonsBookCode/	Archive	Dossier contenant le code des exercices donnés sur le site PylonsBook
./pythonUML/pyut-1.3a-20041228.linux-i686.tar.gz.tar	Archive	Install de PythonUML
./README	??	??
./scripts/CreateBase	Script_Amélie	Permet de créer la coquille pour un nouveau projet (crée les dossiers, contrôleurs...)
./scripts/step.txt	Script_Amélie	Script utilisé pour les livraisons au CHUM. Exporte vers un dossier spécifié, enlève-les .py. FAIRE TRÈS ATTENTION AVANT DE L'EXÉCUTER...
./start_bd	Script_Amélie	Permet de se connecter à la bd
./SVN/FCI/	Archive	Archive d'une ancienne version pour le FCI.
./SVN/francois_software/	Important	Outils développés par François
./SVN/pgx_dss_env/	Important	Environnement virtuel pour Prognomix
./SVN/Prognomix-DSS/	Archive	Archive de l'environnement pour Prognomix.
./SVN/Prognomix-DSS_2.08/	Archive	Archive de l'environnement pour Prognomix.
./SVN/Prognomix-DSS_2.09/	Archive	Archive de l'environnement pour Prognomix.
./SVN/Prognomix-DSS_OLD/	Archive	Archive de l'environnement pour Prognomix.
./SVN/UserCreator/	Script_Amélie	Script utilisé pour créer les utilisateurs PostgreSQL pour le FCI
./SVN/UserCreator/put_this_in_db.py.txt	Gen/Archive	Fichier généré. Allait dans db.py jusqu'à ce qu'on change son format. Désuet.
./SVN/UserCreator/put_this_in_development.ini.txt	Gen	Fichier généré. Son contenu définit les utilisateurs pour le authkit.
./SVN/UserCreator/script.sql	Gen	Fichier généré. Son contenu définit les utilisateurs et leurs droits aux schémas en SQL
./SVN/UserCreator/usercreator.py	Script_Amélie	Génère les fichiers script.sql et les 2 fichiers textes.
./SVN/UserCreator/usercreator.pyc	Script_Amélie	Génère les fichiers script.sql et les 2 fichiers textes.
./SVN/virtualenv.py	Important	Fichier permettant de créer les environnements virtuels

		(voir python doc)
./SVN/virtualenv.pyc	Important	Fichier permettant de créer les environnements virtuels (voir python doc)
./SVN/workflow/	Archive	Archive de l'environnement pour Prognomix.
./test_project/	Archive	Projets tests pour varier
./virtualenv.py	Important	Fichier permettant de créer les environnements virtuels (voir python doc)
./yui/	Archive	Projets tests pour me familiariser avec le yui (yahoo user interface)
./_tmp/BackUpFile_DominicLetarte/	Archive	Documents de Dominic Letarte

4. Listes des répertoires contenant des codes sources sur le serveur.

ANNEXE IV

LISTE DES BASES DE DONNÉES

ANNEXE IV

1. Description de la base de données prognomix_test

Schéma	Name	Type	Owner	Row Data	Field Number	Data yes/no
public	batch	table	Gauthier	0	3	No
public	batch_id_seq	sequence	gauthier	1	9	Yes
public	batch_member	table	Gauthier	0	2	No
public	batch_set	table	Gauthier	0	2	No
public	batches	table	Gauthier	0	2	No
public	category	table	Gauthier	0	4	No
public	cnv	table	Gauthier	0	4	No
public	cnv_annotation	table	Gauthier	0	11	No
public	cnv_run	table	Gauthier	0	3	No
public	cnv_run_id_seq	sequence	Gauthier	1	9	Yes
public	contact	table	Gauthier	0	5	No
public	contact_id_seq	sequence	Gauthier	1	9	Yes
public	covariate_set	table	Gauthier	0	1	No
public	covariate_set_id_seq	sequence	Gauthier	1	9	Yes
public	covariates	table	Gauthier	0	2	No
public	diagnosis	table	Gauthier	0	3	No
public	diagnosis_id_seq	sequence	Gauthier	1	9	No
public	difference_run	table	gauthier	0	2	No

public	difference_run_id_seq	sequence	Gauthier	1	9	Yes
public	differences	table	Gauthier	0	3	No
public	dna	table	Gauthier	0	3	No
public	dna_id_seq	sequence	Gauthier	1	9	Yes
public	dna_log	table	Gauthier	0	6	No
public	dna_log_id_seq	sequence	gauthier	1	9	Yes
public	dph_albuminuria_baseline	table	gauthier	200	3	Yes
public	dph_bmi_baseline	table	Gauthier	200	3	Yes
public	dph_have_complications_baseline	table	Gauthier	200	3	Yes
public	drug	table	Gauthier	0	3	No
public	drug_category	table	Gauthier	0	2	No
public	drug_id_seq	sequence	gauthier	1	9	Yes
public	dummy	table	intégration	1	3	Yes
public	experiment	table	Gauthier	0	10	No
public	experiment_id_seq	sequence	Gauthier	1	9	Yes
public	haplo_block	table	gauthier	0	4	No
public	haplo_block_id_seq	sequence	gauthier	1	9	Yes
public	haplotype	table	Gauthier	0	4	No
public	issues	table	Gauthier	0	2	No
public	lab_batch	table	Gauthier	0	3	No
public	lab_batches	table	Gauthier	0	2	No
public	measure	table	Gauthier	0	4	No
public	measure_id_seq	sequence	Gauthier	1	9	Yes
public	measure_old	table	gauthier	0	4	No
public	medical	table	Gauthier	0	5	No
public	medical_id_seq	sequence	gauthier	1	9	Yes
public	medical_old	table	gauthier	0	4	No

public	medication	table	Gauthier	0	4	No
public	medication_id_seq	sequence	gauthier	1	9	Yes
public	medication_old	table	gauthier	0	4	No
public	normality	table	Gauthier	0	4	No
public	pca	table	Gauthier	0	4	No
public	pca_run	table	Gauthier	0	3	No
public	person	table	Gauthier	200	10	Yes
public	pheno_view	table	gauthier	0	2	No
public	pheno_view_id_seq	sequence	gauthier	1	9	Yes
public	phenotype	table	Gauthier	0	4	No
public	phenotype_id_seq	sequence	gauthier	1	9	Yes
public	phenotype_views	table	integration	7	2	Yes
public	pseudo_autosomal	table	Gauthier	0	2	No
public	publication	table	Gauthier	0	10	No
public	publication_id_seq	sequence	Gauthier	1	9	Yes
public	region	table	Gauthier	0	3	No
public	region_run	table	gauthier	0	2	No
public	results	table	Gauthier	0	10	No
public	results_id_seq	sequence	gauthier	1	9	Yes
public	samples	table	Gauthier	0	3	No
public	samples_id_seq	sequence	Gauthier	1	9	Yes
public	schemachanges	table	Gauthier	19	6	Yes
public	snp_annotation	table	Gauthier	1873278	13	Yes
public	snp_chi_square	table	Gauthier	0	25	No
public	snp_cluster_chart	table	gauthier	0	3	No
public	snp_continuous	table	gauthier	0	18	No
public	snp_eigenstrat	table	gauthier	0	6	No

public	snp_genotype	table	gauthier	200	3	Yes
public	snp_quality	table	gauthier	0	5	No
public	snp_regression_linear	table	gauthier	0	19	No
public	snp_regression_logistic	table	gauthier	0	13	No
public	structure	table	Gauthier	0	5	No
public	structure_run	table	Gauthier	0	2	No
public	structure_run_id_seq	sequence	gauthier	1	9	Yes
public	version	table	Gauthier	6	2	Yes
public	version_id_seq	sequence	Gauthier	1	9	Yes
public	view_2000snps_pc1_1904_eigenstrat	view	Gauthier	0	2	No
public	view_2000snps_pc2_1904_eigenstrat	view	Gauthier	0	2	No
public	view_dph_albuminuria_baseline	view	gauthier			
public	view_dph_albuminuria_extreme_baseline	view	integration			
public	view_dph_amputation_baseline	view	gauthier			
public	view_dph_atrial_fibrillation_vs_no_heart_failure_baseline	view	Gauthier			
public	view_dph_atrial_fibrillation_vs_no_mi_baseline	view	Gauthier			
public	view_dph_atrial_fibrillation_vs_no_mi_nor_hf_baseline	view	Gauthier			
public	view_dph_background_retinopathy_baseline	view	gauthier			
public	view_dph_bmi_baseline	view	gauthier			
public	view_dph_cholesterol_affected_baseline	view	gauthier			
public	view_dph_clean_control_subset_baseline	view	Gauthier			
public	view_dph_glomerular_fr_total_1	view	Gauthier			
public	view_dph_glomerular_fr_vs_clean_controls_baseline	view	Gauthier			
public	view_dph_glomerular_fr_vs_super_controls_total_1	view	Gauthier			
public	view_dph_have_albuminuria_baseline	view	gauthier			
public	view_dph_have_albuminuria_event_1	view	Gauthier			
public	view_dph_have_albuminuria_total_1	view	Gauthier			
public	view_dph_have_complications_baseline	view	gauthier			

public	view_dph_hdl_affected_baseline	view	gauthier			
public	view_dph_hypertension_65_baseline	view	Gauthier			
public	view_dph_hypertension_70_baseline	view	Gauthier			
public	view_dph_hypertension_75_baseline	view	Gauthier			
public	view_dph_hypertension_hbp_baseline	view	gauthier			
public	view_dph_hypertension_or_mi_or_angina_baseline	view	Gauthier			
public	view_dph_ldl_affected_baseline	view	gauthier			
public	view_dph_low_creatinine_clearance_baseline	view	Gauthier			
public	view_dph_macro_albuminuria_affected_10yrs_diabetes_baseline	view	Gauthier			
public	view_dph_macro_albuminuria_affected_5yrs_diabetes_baseline	view	Gauthier			
public	view_dph_macro_albuminuria_no_albuminuria_baseline	view	Gauthier			
public	view_dph_macro_albuminuria_total_1	view	Gauthier			
public	view_dph_macro_vascular_75_vs_clean_controls_baseline	view	Gauthier			
public	view_dph_macro_vascular_75_vs_super_controls_baseline	view	Gauthier			
public	view_dph_macro_vascular_75_vs_super_controls_total_1	view	Gauthier			
public	view_dph_macro_vascular_vs_clean_controls_baseline	view	Gauthier			
public	view_dph_macro_vascular_vs_super_controls_baseline	view	Gauthier			
public	view_dph_macro_vascular_vs_super_controls_total_1	view	Gauthier			
public	view_dph_macular_oedema_baseline	view	gauthier			
public	view_dph_major_cerebrovascular_total_1	view	Gauthier			
public	view_dph_mean_arterial_pressure_baseline	view	Gauthier			
public	view_dph_micro_albuminuria_no_albuminuria_baseline	view	Gauthier			
public	view_dph_micro_albuminuria_total_1	view	Gauthier			
public	view_dph_micro_macro_cardiovascular_disease_event_1_date	view	Gauthier			
public	view_dph_micro_macro_vascular_75_vs_clean_controls_baseline	view	Gauthier			
public	view_dph_micro_macro_vascular_75_vs_super_controls_baseline	view	Gauthier			
public	view_dph_micro_macro_vascular_75_vs_super_controls_total_1	view	Gauthier			
public	view_dph_micro_macro_vascular_vs_clean_controls_baseline	view	Gauthier			
public	view_dph_micro_macro_vascular_vs_super_controls_baseline	view	Gauthier			

public	view_dph_micro_macro_vascular_vs_super_controls_total_1	view	Gauthier			
public	view_dph_micro_or_macroalbuminuria_vs_clean_controls_baseline	view	Gauthier			
public	view_dph_micro_or_macroalbuminuria_vs_super_controls_total_1	view	Gauthier			
public	view_dph_micro_vascular_75_vs_clean_controls_baseline	view	Gauthier			
public	view_dph_micro_vascular_75_vs_super_controls_baseline	view	Gauthier			
public	view_dph_micro_vascular_75_vs_super_controls_total_1	view	Gauthier			
public	view_dph_micro_vascular_vs_clean_controls_baseline	view	Gauthier			
public	view_dph_micro_vascular_vs_super_controls_baseline	view	Gauthier			
public	view_dph_micro_vascular_vs_super_controls_total_1	view	Gauthier			
public	view_dph_myocardial_infarction_or_angina_baseline	view	Gauthier			
public	view_dph_myocardial_infarction_total_1	view	Gauthier			
public	view_dph_myocardial_infarction_vs_clean_controls_baseline	view	Gauthier			
public	view_dph_myocardial_infarction_vs_super_controls_total_1	view	Gauthier			
public	view_dph_proliferative_retinopathy_baseline	view	gauthier			
public	view_dph_stroke_or_tia_baseline	view	Gauthier			
public	view_dph_stroke_or_tia_vs_clean_controls_baseline	view	Gauthier			
public	view_dph_stroke_or_tia_vs_super_controls_total_1	view	Gauthier			
public	view_dph_stroke_total_1	view	Gauthier			
public	view_dph_super_control_subset_total_1	view	Gauthier			
public	view_dph_systolic_bp_categories_baseline	view	Gauthier			
public	view_dph_total_coronary_events_baseline	view	Gauthier			
public	view_dph_total_coronary_events_total_1	view	Gauthier			
public	view_dph_triglycerides_affected_baseline	view	gauthier			
public	view_dph_waist_hip_baseline	view	gauthier			
public	view_med_ace_other_baseline	view	gauthier			
public	view_med_all_cause_death_total_1	view	Gauthier			
public	view_med_all_cause_death_total_1_date	view	Gauthier			
public	view_med_all_events_1_date	view	Gauthier			

public	view_med_alpha_glucosidase_baseline	view	gauthier			
public	view_med_amputation_baseline	view	gauthier			
public	view_med_amputation_left_baseline	view	gauthier			
public	view_med_amputation_right_baseline	view	gauthier			
public	view_med_angina_baseline	view	gauthier			
public	view_med_angiotensin_baseline	view	gauthier			
public	view_med_ankle_reflex_left_baseline	view	gauthier			
public	view_med_ankle_reflex_right_baseline	view	gauthier			
public	view_med_antihypertensive_other_baseline	view	gauthier			
public	view_med_antiplatelet_other_baseline	view	gauthier			
public	view_med_aspirin_baseline	view	gauthier			
public	view_med_atrial_fibrillation_baseline	view	gauthier			
public	view_med_background_retinopathy_left_baseline	view	gauthier			
public	view_med_background_retinopathy_right_baseline	view	gauthier			
public	view_med_beta_blocker_baseline	view	gauthier			
public	view_med_blindness_baseline	view	gauthier			
public	view_med_calcium_baseline	view	gauthier			
public	view_med_cardiovascular_death_total_1	view	Gauthier			
public	view_med_cardiovascular_death_total_1_date	view	Gauthier			
public	view_med_cataract_left_baseline	view	gauthier			
public	view_med_cataract_right_baseline	view	gauthier			
public	view_med_cholesterol_lowering_other_baseline	view	gauthier			
public	view_med_chronic_ulceration_left_baseline	view	gauthier			
public	view_med_chronic_ulceration_right_baseline	view	gauthier			
public	view_med_coronary_artery_bypass_baseline	view	gauthier			
public	view_med_dementia_baseline	view	gauthier			
public	view_med_diuretic_other_baseline	view	gauthier			
public	view_med_dorsalis_pedis_pulse_left_baseline	view	gauthier			
public	view_med_dorsalis_pedis_pulse_right_baseline	view	gauthier			

public	view_med_erection_dysfunction_baseline	view	gauthier			
public	view_med_event_1_first	view	Gauthier			
public	view_med_gliclazide_baseline	view	gauthier			
public	view_med_glinide_baseline	view	gauthier			
public	view_med_heart_failure_baseline	view	gauthier			
public	view_med_hmg_coa_baseline	view	gauthier			
public	view_med_hormone_replacement_therapy_baseline	view	gauthier			
public	view_med_hypertension_baseline	view	gauthier			
public	view_med_hypertension_vs_clean_controls_baseline	view	gauthier			
public	view_med_hypertension_vs_super_controls_baseline	view	gauthier			
public	view_med_insulin_bed_baseline	view	gauthier			
public	view_med_insulin_day_baseline	view	gauthier			
public	view_med_knee_reflex_left_baseline	view	gauthier			
public	view_med_knee_reflex_right_baseline	view	gauthier			
public	view_med_laser_therapy_left_baseline	view	gauthier			
public	view_med_laser_therapy_right_baseline	view	gauthier			
public	view_med_left_ventricular_hypertrophy_baseline	view	gauthier			
public	view_med_light_touch_sensation_left_baseline	view	gauthier			
public	view_med_light_touch_sensation_right_baseline	view	gauthier			
public	view_med_macro_albuminuria_baseline	view	gauthier			
public	view_med_macro_albuminuria_event_1	view	Gauthier			
public	view_med_macro_albuminuria_event_1_date	view	Gauthier			
public	view_med_macro_vascular_event_1	view	Gauthier			
public	view_med_macro_vascular_event_1_date	view	Gauthier			
public	view_med_macular_oedema_baseline	view	gauthier			
public	view_med_macular_oedema_left_baseline	view	gauthier			
public	view_med_macular_oedema_right_baseline	view	gauthier			
public	view_med_major_cardiovas_disease_event_1_first_vs_super_cont rol	view	Gauthier			

public	view_med_major_cardiovascular_baseline_vs_clean_controls	view	gauthier			
public	view_med_major_cardiovascular_disease_event_1	view	Gauthier			
public	view_med_major_cardiovascular_disease_event_1_date	view	Gauthier			
public	view_med_major_cardiovascular_disease_event_1_vs_super_controls	view	Gauthier			
public	view_med_major_cardiovascular_total_1_vs_super_controls	view	Gauthier			
public	view_med_major_cerebrovascular_event_1	view	Gauthier			
public	view_med_major_cerebrovascular_event_1_date	view	Gauthier			
public	view_med_major_cerebrovascular_total_1	view	Gauthier			
public	view_med_major_coronary_heart_disease_event_1	view	Gauthier			
public	view_med_major_coronary_heart_disease_event_1_date	view	Gauthier			
public	view_med_metformin_baseline	view	gauthier			
public	view_med_micro_albuminuria_baseline	view	gauthier			
public	view_med_micro_albuminuria_event_1	view	Gauthier			
public	view_med_micro_albuminuria_event_1_date	view	Gauthier			
public	view_med_micro_vascular_event_1	view	Gauthier			
public	view_med_micro_vascular_event_1_date	view	Gauthier			
public	view_med_myocardial_infarction_baseline	view	gauthier			
public	view_med_myocardial_infarction_event_1	view	Gauthier			
public	view_med_myocardial_infarction_event_1_date	view	Gauthier			
public	view_med_new_worsening_nephropathy_event_1	view	Gauthier			
public	view_med_new_worsening_nephropathy_event_1_date	view	Gauthier			
public	view_med_new_worsening_retinopathy_event_1	view	Gauthier			
public	view_med_new_worsening_retinopathy_event_1_date	view	Gauthier			
public	view_med_nitrates_baseline	view	gauthier			
public	view_med_non_cardiovascular_death_total_1	view	Gauthier			
public	view_med_only_myocardial_infarction_baseline	view	gauthier			
public	view_med_ophthalmologist_examination_baseline	view	gauthier			
public	view_med_oral_anticoagulant_baseline	view	gauthier			

public	view_med_perindopril_baseline	view	gauthier			
public	view_med_peripheral_revascularisation_baseline	view	gauthier			
public	view_med_posterior_tibial_pulse_left_baseline	view	gauthier			
public	view_med_posterior_tibial_pulse_right_baseline	view	gauthier			
public	view_med_proliferative_retinopathy_baseline	view	gauthier			
public	view_med_proliferative_retinopathy_left_baseline	view	gauthier			
public	view_med_proliferative_retinopathy_right_baseline	view	gauthier			
public	view_med_q_waves_mi_diagnostic_baseline	view	gauthier			
public	view_med_randomization_date	view	gauthier			
public	view_med_regular_help_requirement_left_baseline	view	gauthier			
public	view_med_retinal_photocoagulation_baseline	view	gauthier			
public	view_med_stroke_baseline	view	gauthier			
public	view_med_stroke_event_1	view	Gauthier			
public	view_med_stroke_event_1_date	view	Gauthier			
public	view_med_sulphonylurea_other_baseline	view	gauthier			
public	view_med_thiazide_baseline	view	gauthier			
public	view_med_thiazolidinedione_baseline	view	gauthier			
public	view_med_transient_ischaemic_attack_baseline	view	gauthier			
public	view_pc1_1904_eigenstrat	view	Gauthier			
public	view_pc2_1904_eigenstrat	view	Gauthier			
public	view_pheno_1	view	intégration			
public	view_pheno_2	view	intégration			
public	view_pheno_3	view	intégration			
public	view_pop_men	view	gauthier			
public	view_pop_women	view	gauthier			
public	view_pph_age_baseline	view	gauthier			
public	view_pph_albumin_creatinine_ratio_baseline	view	gauthier			
public	view_pph_alt_baseline	view	gauthier			

public	view_pph_average_cigarettes_baseline	view	gauthier			
public	view_pph_cholesterol_baseline	view	gauthier			
public	view_pph_creatinine_baseline	view	gauthier			
public	view_pph_creatinine_clearance_baseline	view	gauthier			
public	view_pph_current_beer_baseline	view	gauthier			
public	view_pph_current_cigarettes_baseline	view	gauthier			
public	view_pph_current_drinker_baseline	view	gauthier			
public	view_pph_current_pipe_cigar_baseline	view	gauthier			
public	view_pph_current_spirits_baseline	view	gauthier			
public	view_pph_current_wine_baseline	view	gauthier			
public	view_pph_diabetes_children_baseline	view	gauthier			
public	view_pph_diabetes_parents_baseline	view	gauthier			
public	view_pph_diabetes_siblings_baseline	view	gauthier			
public	view_pph_diabetes_years_baseline	view	gauthier			
public	view_pph_diagnosis_age_baseline	view	gauthier			
public	view_pph_diastolic_bp_baseline	view	gauthier			
public	view_pph_dietician_baseline	view	gauthier			
public	view_pph_education_age_baseline	view	gauthier			
public	view_pph_ever_smoked_baseline	view	gauthier			
public	view_pph_exercise_baseline	view	gauthier			
public	view_pph_exercise_mild_baseline	view	gauthier			
public	view_pph_exercise_moderate_baseline	view	gauthier			
public	view_pph_exercise_vigorous_baseline	view	gauthier			
public	view_pph_glucose_baseline	view	gauthier			
public	view_pph_gums_baseline	view	gauthier			
public	view_pph_hba1c_baseline	view	Gauthier			
public	view_pph_hdl_baseline	view	gauthier			
public	view_pph_heart_rate_baseline	view	gauthier			
public	view_pph_height_baseline	view	gauthier			

public	view_pph_hip_baseline	view	gauthier			
public	view_pph_last_smoked_age_baseline	view	gauthier			
public	view_pph_ldl_baseline	view	gauthier			
public	view_pph_mmse_baseline	view	gauthier			
public	view_pph_past_beer_baseline	view	gauthier			
public	view_pph_past_drinker_baseline	view	gauthier			
public	view_pph_past_spirits_baseline	view	gauthier			
public	view_pph_past_wine_baseline	view	gauthier			
public	view_pph_potassium_baseline	view	gauthier			
public	view_pph_pulse_pressure_baseline	view	gauthier			
public	view_pph_sex_baseline	view	gauthier			
public	view_pph_sodium_baseline	view	gauthier			
public	view_pph_systolic_bp_baseline	view	gauthier			
public	view_pph_teeth_baseline	view	gauthier			
public	view_pph_triglycerides_baseline	view	gauthier			
public	view_pph_visual_acuity_1l_baseline	view	Gauthier			
public	view_pph_visual_acuity_1r_baseline	view	Gauthier			
public	view_pph_visual_acuity_2l_baseline	view	Gauthier			
public	view_pph_visual_acuity_2r_baseline	view	Gauthier			
public	view_pph_waist_baseline	view	gauthier			
public	view_pph_weight_baseline	view	gauthier			
public	view_valid_caucasian	view	gauthier			
public	visit	table	Gauthier	0	7	No
public	visit_id_seq	sequence	Gauthier	1	9	Yes

2. Description de la base de données « Experiment »

Schema	Name	Type	Owner
public	TEST_AMELIE	view	integration
public	assigned_to	view	gauthier
public	contact	table	gauthier
public	experiment	table	gauthier
public	experiment_parameter	table	gauthier
public	originates_from	view	gauthier
public	parameter	table	gauthier
public	parameter_type	table	gauthier
public	phenotype_views	table	gauthier
public	sid	view	gauthier
public	status	table	gauthier
public	template_descriptor	table	integration
public	template_descriptor_parameters	table	integration
public	test_amelie	view	integration

ANNEXE V

CONFIGURATION DU SYSTÈME

ANNEXE V

1- Les étapes de configurations du système

a) Installation des packages utiles pour le système.

Pour faire fonctionner le système, la version de Python_2.7 et de Pylons_1.0 sont les minimums nécessaires.

Une fois python est installé et configuré tapez le terminal Linux cette commande.

```
easy_install -U Pylons
```

Dans le cas l'outil easy_install n'est pas installé, télécharger le package sur internet puis suivre le guide de son installation.

Maintenant l'environnement est installé, il faut paramétrer le profil de l'utilisateur qui doit utiliser le système en modifiant le fichier .bashrc, .bash_profile et .pgpass

Par exemple.

Le fichier /home/gauthier/.bashrc doit contenir les instructions suivantes;

```
# /etc/skel/.bashrc
#
# This file is sourced by all *interactive* bash shells on startup,
# including some apparently interactive shells such as scp and rcp
# that can't tolerate any output.  So make sure this doesn't display
# anything or bad things will happen !

# Test for an interactive shell.  There is no need to set anything
# past this point for scp and rcp, and it's important to refrain from
# outputting anything in those cases.
if [[ $- != *i* ]] ; then
    # Shell is non-interactive.  Be done now!
    return
fi

# Put your fun stuff here.

# Add local installs to PATH
```

```
#Chemins
MYPATH=.:...:$HOME:$HOME/software/bin/biogenix:$HOME/software/bin/prognomix
:$HOME/software/bin/external
PATH=$MYPATH:$PATH
```

```
# environment variables
PYTHONPATH=.:...:$HOME/software
#PYTHONPATH=.:...:$HOME/software/wrapper/dev:$HOME/software/domain_checker/
dev:$HOME/software/lib:$HOME/software/lib/biogenix
```

```
export PROGNOPIX_DB_NAME=prognomix_test
export PROGNOPIX_DB_HOST=10.180.121.31
export PROGNOPIX_DB_PORT=5432
export PROGNOPIX_DB_USER=integration
export PROGNOPIX_CONN_STR="-d $PROGNOPIX_DB_NAME -h $PROGNOPIX_DB_HOST -p
$PROGNOPIX_DB_PORT -U $PROGNOPIX_DB_USER"
```

Dans le fichier /home/Gauthier/.bash_profile, il faut mettre :

```
[[ -f ~/.bashrc ]] && . ~/.bashrc
```

Le fichier /home/Gauthier/.pgpass , il faut mettre :

```
10.180.121.31:5432:experiment:integration:postgres
10.180.121.31:5432:experiment:integration:postgres
10.180.121.31:5432:experiment:integration:postgres
10.180.121.31:5432:experiment:integration:postgres
10.180.121.31:5432:prognomix_test:integration:postgres
```

Pour exécuter l'environnement il faut d'abord s'assurer de bien configurer le fichier /home/gauthier/Prognomix_DSS_2.09/development.ini. Ici la configuration nécessaire.

```
#
# biogenix - Pylons development environment configuration
#
# The %(here)s variable will be replaced with the parent directory of this
file
#
[DEFAULT]
debug = true
# Uncomment and replace with the address which should receive any error
reports
#email_to = you@yourdomain.com
smtp_server = localhost
error_email_from = paste@localhost

[server:main]
use = egg:Paste#http
host = 10.180.121.31 -> Ici c'est l'adresse IP du serveur.
port = 5000 -> Ici c'es le port d'écoute du serveur
```

[app:main] -> Le contenu de cette partie ne dois pas être modifié

```

use = egg:biogenix
full_stack = true
cache_dir = %(here)s/data
beaker.session.key = biogenix
beaker.session.secret = somesecret

#####
#
# If you'd like to fine-tune the individual locations of the cache data
# dirs
# for the cache data, or the session saves, un-comment the desired
# settings
# here:
#beaker.cache.data_dir = %(here)s/data/cache
#beaker.session.data_dir = %(here)s/data/sessions

authkit.setup.method = form, cookie
authkit.cookie.secret = secret string
authkit.cookie.signoutpath = /auth/signout
authkit.cookie.params.expires = 3600
authkit.cookie.enforce = True
authkit.cookie.includeip = True

# -> Ci-dessous c'est la configuration de la liste des utilisateurs du
#système.
authkit.form.authenticate.user.data = johanne:johanne researcher
                                     ghislain:ghislain researcher
                                     pavel:pavel pi
                                     carole:carole secretary
                                     maxime:maxime analyst
                                     johanna:johanna analyst
                                     ettore:ettore researcher
                                     god:god administrator

pool.database = experiment #-> C'est la base de données par défaut
pool.user = intégration #-> L'utilisateur par défaut
pool.host = 10.180.121.31 # -> L'adresse du serveur
pool.password = postgres # -> Le mot de passé pour la base de données

#-----

Une fois le système est configuré, il faut utiliser cette commande pour
l'exécuter.
Dans /home/Gauthier/Prognomix-DSS_2.09/

Utiliser la commande :
./start.sh : pour lancer le système à partir d'un fichier script.

Ou

paster serve --reload development.ini

à note c'est le même script qu'on retrouve dans le fichier
/home/gauthier/Pronomix-DSS_2.09/start.sh

```


ANNEXE VI

FONCTIONNALITÉS DU SYSTÈME

ANNEXE VI

1- L'écran principal après le lancement du système.

god(administrator) | [help](#)

LogBook
Some experiments are present on the disk but not in the database. Click Add Experiments to index them in the database.

Compare experiments with id: and

id	Name	Priority	Inscription	Start	Stop	assigned	originates	Status
6	snp_compare_2	None	None	None	None	None	None	completed
5	boby1	M	2014-11-24 00:00:00	2014-11-24 17:26:48.202739	None	Dominic Letarte	Pavel Hamet	completed
4	boby	M	2014-11-24 00:00:00	2014-11-24 17:25:43.878119	None	Francois Gauthier	Pavel Hamet	completed
3	test_final	M	2009-05-05 00:00:00	2009-05-05 16:08:11.18085	None	Francois Gauthier	Pavel Hamet	completed

Cet écran présente toutes les fonctionnalités du système.

3.10 1- LogBook

Cette fonctionnalité affiche la liste des expériences ajoutées dans le système. Il permet d'ajouter également des expériences dans le système. Entre autres, il permet de comparer deux expériences qui ont été effectuées.

LogBook

Some experiments are present on the disk but not in the database. Click **Add Experiments** to index them in the database.

Compare experiments with id: and

id	Name	Priority	Inscription	Start	Stop	assigned	originates	Status
		All				All	All	All
6	snp_compare_2	None	None	None	None	None	None	completed
5	boby1	M	2014-11-24 00:00:00	2014-11-24 17:26:48.202739	None	Dominic Letarte	Pavel Hamet	completed
4	boby	M	2014-11-24 00:00:00	2014-11-24 17:25:43.878119	None	Francois Gauthier	Pavel Hamet	completed
3	test_final	M	2009-05-05 00:00:00	2009-05-05 16:02:11.40025	None	Francois Gauthier	Pavel Hamet	completed
2	Association	L	2007-10-01 00:00:00	2007-12-16 00:00:00	2007-12-29 00:00:00	Ettore Merlo	Pavel Hamet	completed
1	lpsum	M	2008-01-01	2008-01-01	2008-01-10	Francois Gauthier	Francois Gauthier	pending

Quand l'utilisateur compare deux expériences, il va avoir cette fenêtre comme résultat.

Dans cet exemple la comparaison est faite en les expériences 4 et 5.

Compare Experiments #4 and #5

Identical Added Different

Logbook of #4		Logbook of #5	
inscription:	2014-11-24 00:00:00	inscription:	2014-11-24 00:00:00
scheduled_stop:	None	scheduled_stop:	None
name:	boby	name:	boby1
resdir:	boby	resdir:	boby1
sid:	completed	sid:	completed
validity:	None	validity:	None
scheduled_start:	2014-11-24 17:25:43.878119	scheduled_start:	2014-11-24 17:26:48.202739
priority:	M	priority:	M
comments:	fdsa	comments:	afds
assigned_to:	Francois Gauthier	assigned_to:	Dominic Letarte
originates_from:	Pavel Hamet	originates_from:	Pavel Hamet
id:	4	id:	5
description:	dfsa	description:	dsaf
Descriptor of #4		Descriptor of #5	
Stats.base_phenotype:	view_dph_albuminuria_baseline	Stats.base_phenotype:	view_pheno_2
Stats.comp_phenotypes:	view_med_myocardial_infarction_baseline,	Stats.comp_phenotypes:	view_med_myocardial_infarction_baseline,
Stats.output:	descriptive_statistics.csv	Stats.output:	descriptive_statistics.csv
Stats.pids:	test_5_0.pids	Stats.pids:	database_data.py
Prognomix.Database.host:	10.180.121.31	Prognomix.Database.host:	10.180.121.31
Prognomix.Database.name:	prognomix_test	Prognomix.Database.name:	prognomix_test
Prognomix.Database.user:	integration	Prognomix.Database.user:	integration
Prognomix.Database.version:	19	Prognomix.Database.version:	19
Results of #4		Results of #5	

3.11 2- Administrations

Cette fonctionnalité permet de visualiser les rapports des différentes les rencontres effectuées avec les différents chercheurs et le plan d'action mis en place pour la poursuite des travaux.



3.12 3- Define Parameters

Étant donné que le système est développé dans un but d'être flexible d'utilisation. Il permet de définir les paramètres pour la base de données de phénotype.



3.13 4- Search Experiments

Cette fonctionnalité permet de rechercher des expériences effectuées dans le système.

C'est une fonctionnalité très flexible qui permet d'utiliser autant de critères possibles pour faire la recherche. Une fois l'expérience recherchée est trouvée, il est possible de protéger le fichier à télécharger par un mot de passe. Voir l'écran 2.

L'écran 1 :

The screenshot shows the Prognomix DSS web interface. At the top left is the Prognomix logo with the tagline "Prédire pour prévenir Predict to prevent". At the top right, it says "Prognomix DSS Discovery support system". Below the header, there is a navigation menu on the left with items like LOGBOOK, ADMINISTRATION, DEFINE PARAMETERS, SEARCH EXPERIMENTS, SAVE EXPERIMENTS, ASSOCIATION, MACHINE LEARNING, PCA, DESCRIPTIVE STATS, SNP COMPARISON, and DESCRIPTORS. The main content area is titled "Search experiments" and contains a search form with three rows of criteria. Each row has a dropdown for the search attribute, a dropdown for the operator, and a text input for the value or value interval. The criteria are: "id (integer)", "comments (string)", and "id (integer)". Below the criteria are buttons for "Add criteria" and "Remove selected criteria". A "Search" button is at the bottom of the form. At the bottom of the page, it says "Copyright © 2008. All rights reserved".

L'écran 2

The screenshot shows the "Summary" page of the Prognomix DSS interface. It indicates that "2 experiments selected". The first experiment is named "boby" and consists of "1 files [3.0 KB]: descriptor.dat, and 3 others files [3.2 KB] with raw data". The second experiment is named "boby1" and consists of "1 files [3.0 KB]: descriptor.dat, and 3 others files [5.8 KB] with raw data". Below the summary is a "Download" section. It has a checkbox for "protect archive with password:" followed by a text input field. There are two radio button options: "Results.exe self-extract zip file [6.0 KB]" and "Results.exe with raw data, self-extract zip file [15.0 KB]". A "Download" button is at the bottom of the section.

3.14 5- Save Experiments

3.15 6- Associations

LOGBOOK

ADMINISTRATION

DEFINE PARAMETERS

SEARCH EXPERIMENTS

SAVE EXPERIMENTS

ASSOCIATION

MACHINE LEARNING

PCA

DESCRIPTIVE STATS

SNP COMPARISON

DESCRIPTORS

New association study

Experiment description

Experiment name: Assigned to: Francois Gauthier Originates from: Pavel Hamet Experiment description:

Experiment comments:

Genotype extraction

Affymetrix 5.0 Pids file: None Affymetrix 6.0 Pids file: None Snp file (affy format): test.affy

Quality filters

Plink maf: Plink mind: Plink geno: Plink hwe:

Association model

Case/Control association Logistic regression association

Phenotype extraction

Phenotype(s) name(s):

- view_dph_albuminuria_baseline
- view_dph_albuminuria_extreme_baseline
- view_dph_have_complications_baseline
- view_pheno_1
- view_pheno_2
- view_pheno_3

Covariate(s) name(s):

- view_dph_albuminuria_baseline
- view_dph_albuminuria_extreme_baseline
- view_dph_bmi_baseline
- view_dph_have_complications_baseline
- view_pheno_1
- view_pheno_2
- view_pheno_3

PCA

Include PCA as covariate ?
 Yes No

Number of PCA to compute: P-value threshold: Random seed:

Distance:

Format association results

P-value cutoff: Output file name:

3.16 7- Machine Learning

LOGBOOK

ADMINISTRATION

DEFINE PARAMETERS

SEARCH EXPERIMENTS

SAVE EXPERIMENTS

ASSOCIATION

MACHINE LEARNING

PCA

DESCRIPTIVE STATS

SNP COMPARISON

DESCRIPTORS

New machine learning study

New case/control prediction study

Experiment name: Assigned to: Originates from: Experiment description:

Experiment comments:

3.17 8- PCA

LOGBOOK

ADMINISTRATION

DEFINE PARAMETERS

SEARCH EXPERIMENTS

SAVE EXPERIMENTS

ASSOCIATION

MACHINE LEARNING

PCA

DESCRIPTIVE STATS

SNP COMPARISON

DESCRIPTORS

New PCA study

Experiment description

Experiment name: Assigned to: Originates from: Experiment description:

Experiment comments:

Genotype extraction

Affymetrix 5.0 Pids file: Affymetrix 6.0 Pids file: Snp file (affy format):

Quality filters

Plink maf: Plink mind: Plink geno: Plink hwe:

Phenotypes extraction

Phenotype(s) name(s):

- view_dph_albuminuria_baseline
- view_dph_albuminuria_extreme_baseline
- view_dph_have_complications_baseline
- view_pheno_1
- view_pheno_2
- view_pheno_3

Unassociated and spaced snps

P-value threshold: Random seed: Distance:

PCA

Number of PCA to compute: PCA plot title: PCA plot output file: PCA output file:

3.18 9- Descriptive Stats

LOGBOOK

ADMINISTRATION

DEFINE PARAMETERS

SEARCH EXPERIMENTS

SAVE EXPERIMENTS

ASSOCIATION

MACHINE LEARNING

PCA

DESCRIPTIVE STATS

SNP COMPARISON

DESCRIPTORS

New descriptive statistics

Experiment description

Experiment name: Assigned to: Originates from: Experiment description:

Experiment comments:

Patients selection

Pids file:

Phenotypes extraction

Base phenotype name: Comparison phenotype set:

Descriptive statistics results

Output file name:
descriptive_statistics

Next

3.19 10- SNP Comparison

LOGBOOK

ADMINISTRATION

DEFINE PARAMETERS

SEARCH EXPERIMENTS

SAVE EXPERIMENTS

ASSOCIATION

MACHINE LEARNING

PCA

DESCRIPTIVE STATS

SNP COMPARISON

DESCRIPTORS

New snp comparison study

Experiment description

Experiment name: _____ Assigned to: Francois Gauthier Originates from: Pavel Hamet Experiment description: _____

Experiment comments: _____

New snp comparison study

Available experiments:

- 8 -- MARKOV_RECESSIVE_COMP_B
- 7 -- MARKOV_RECESSIVE_COMP_B
- 5 -- boby1
- 4 -- boby
- 3 -- test_final
- 2 -- Association

Experiments to compare: _____

Quality filters

P-Value threshold: _____ Intersection threshold: _____ Odds ratio lower than: _____ Odds ratio higher than: _____

Report unique snps ?

Yes No

Chromosome filters

Restrict to chromosome: All Range from: _____ Range to: _____

Snp comparison results

Output file name:
snp_comparison_results

Next

Aide associé pour son utilisation:

Snp comparison analysis help

Purpose:

Given a set of genetic association studies, extract those snps that satisfy *p-value* and/or *odds ratio* filters and that are shared by at least *intersection threshold* of the associations studies.

If requested, also report snps that appear in only one association study.

If requested, only report snps that belong to a specific chromosome, in a specific position range.

Usage:

You must first choose the experiments you wish to compare by transferring them from the *available experiments* menu to the *experiments to compare* menu using the arrow buttons.

P-value threshold: If a threshold is set, the analysis will only include snps with a p-value lower or equal to the p-value threshold.

Intersection threshold: Only snps that are shared by at least *intersection threshold* of all selected experiments will be reported.

OR filters: If set, only report snps with an OR lower than *low OR* **or** higher than *high OR*.

Unique snps: Unique snps refer to those snps that appear in only one of the selected experiments.

Restrict to chromosome: Only report snps belonging to the selected chromosome.

Range from / to: Only report snps in the specified position range. If either *from* or *to* is set, only report snps downstream or upstream of the specified position. This option is only effective if results are restricted to a specific chromosome.

3.20 11- Descriptors

Create A template Descriptor

Create a template

Enter template information

Experiment type : Description :

Choose template parameters

- Prognomix
- Stats
- config
- markov
- options
- prognomix
- python
- signature

Open Template

Template descriptor details

Parameters list

You can enter values in the following fields before generating your descriptor file.

Prognomix.Database.host	<input type="text"/>
Prognomix.Database.name	<input type="text"/>
Prognomix.Database.user	<input type="text"/>
Prognomix.Database.version	<input type="text"/>
Stats.base_phenotype	<input type="text"/>
Stats.comp_phenotypes	<input type="text"/>
Stats.output	<input type="text"/>
Stats.pids	<input type="text"/>
signature.ddd	<input type="text"/>
signature.features.max	<input type="text"/>
signature.features.min	<input type="text"/>
signature.subjects.max	<input type="text"/>
signature.subjects.min	<input type="text"/>

Add Parameter

Add parameter

Choose an existing category or create a new one

Existing category :
Prognomix.Database

New category :

Enter parameter information

Name : _____ Type : string Description : _____

Cet écran permettra aux chercheurs de définir de nouveaux paramètres pour de nouvelles expériences.