



HAL
open science

Analyse, conception et développement d'une interface graphique pour un programme fortran

Camille Rosa

► **To cite this version:**

Camille Rosa. Analyse, conception et développement d'une interface graphique pour un programme fortran. [Stage] IUP GMI. Université d'Avignon et des Pays de Vaucluse (UAPV), FRA. 2003, 31 p. hal-02833503

HAL Id: hal-02833503

<https://hal.inrae.fr/hal-02833503>

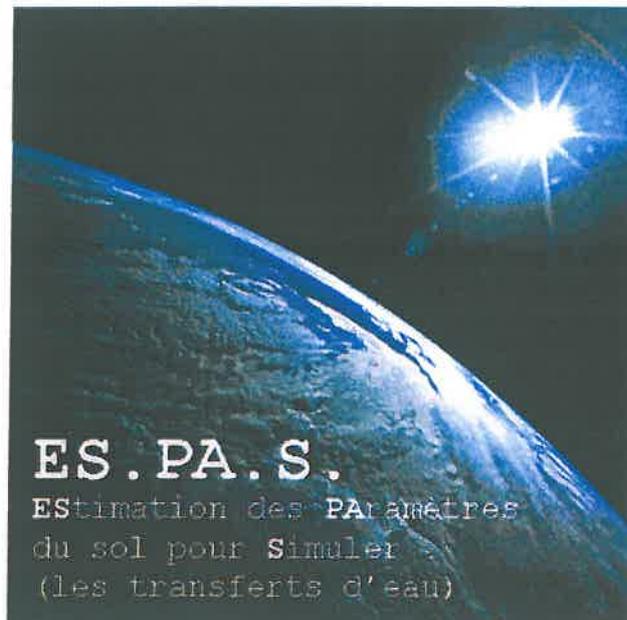
Submitted on 7 Jun 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

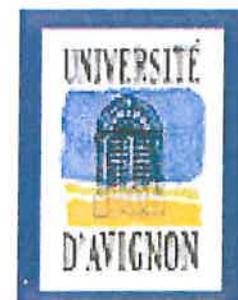
L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Analyse, conception et développement d'une interface graphique pour un programme Fortran



Stagiaire : Cédric ROSA
- IUP3, promotion 2000-2003
Professeur tuteur : Patrick Nasarre
Maitre de stage : Stéphane Ruy



Sommaire

Remerciements.....	4
Avant propos	5
1 Cadre de Travail.....	6
1.1 L'INRA	6
1.2 Le centre d'Avignon	7
1.3 L'unité CSE (Climat, Sol et Environnement).....	8
1.4 L'équipe TMSH (Transferts de Masse en Sols Hétérogènes).....	9
2 Contexte de la mission	11
2.1 Origine des besoins	11
2.2 Définition de la mission	13
2.3 Planification	14
2.4 Organisation	15
2.5 Moyens matériels et logiciels.....	16
3 Réalisation.....	17
3.1 Analyse	17
3.2 Conception	23
3.3 Implémentation	26
3.4 Tests	27
3.5 Problèmes rencontrés	27
3.6 Avantages et inconvénients de l'IHM.....	28
Bilan du stage.....	29

Remerciements

Stéphane Ruy, pour m'avoir permis d'effectuer ce stage, et m'avoir accueilli, dans la bonne humeur, pendant cinq mois au sein de son équipe mais aussi pour sa gentillesse et sa compréhension.

Nathalie Moitrier, pour ses précieux conseils techniques.

L'équipe CSE, pour leur accueil chaleureux.

Jean, Nicolas et Marc, pour leur soutien.

Estelle, qui, comme à son d'habitude m'a beaucoup apportée grâce à ses conseils de rédaction et sa relecture.

Avant propos

Je suis en troisième et dernière année d'IUP informatique à l'université d'Avignon et cette formation se termine par un stage validant notre diplôme.

Mon parcours universitaire m'ayant déjà amené à réaliser un projet dans une entreprise, je voulais changer de lieu de travail. En regardant la liste des stages proposés sur l'Intranet de l'IUP, ce stage est le seul à avoir retenu mon attention au premier coup d'œil. Je n'avais jamais effectué de projet entier lors de mes dernières expériences professionnelles et je voulais acquérir des connaissances en matière de gestion de projet. Ce qui m'a aussi beaucoup intéressé, c'est de connaître les habitudes de travail dans un institut de recherche. Pour finir, je souhaitais aussi développer mes compétences au niveau du développement JAVA avec XML/XSLT.

1 Cadre de Travail

1.1 L'INRA

L'Institut National de la Recherche Agronomique a été créé en 1946 en application de la loi Tanguy Prigent, le ministre de l'agriculture de l'époque. Il s'agit d'un établissement public à caractère scientifique et technologique, placé sous la double tutelle des ministères chargés de la recherche et de l'agriculture.

L'institut a plusieurs rôles, il doit :

- œuvrer au service de l'intérêt public tout en maintenant l'équilibre entre les exigences de la recherche et les demandes de la société.
- Produire et diffuser des connaissances scientifiques et des innovations, principalement dans les domaines de l'agriculture, de l'alimentation et de l'environnement.
- contribuer à l'expertise, à la formation, à la promotion de la culture scientifique et technique, au débat science / société.

Présent sur tout le territoire, il est organisé de la manière suivante :

- 17 départements de recherche touchant l'agriculture, l'alimentation et l'environnement regroupés dans 5 directions scientifiques (cf. figure1).
- 21 centres régionaux répartis en près de 200 sites dans toute la France.
- 257 unités de recherche (dont 128 associées à d'autres organismes comme le CNRS, le CIRAD, universités,...).
- 80 unités expérimentales.
- 131 unités d'appuis et de services.

Sur le plan des moyens financiers, le budget de l'INRA est de 573 millions d'euros (env. 3,7 milliards de francs...)

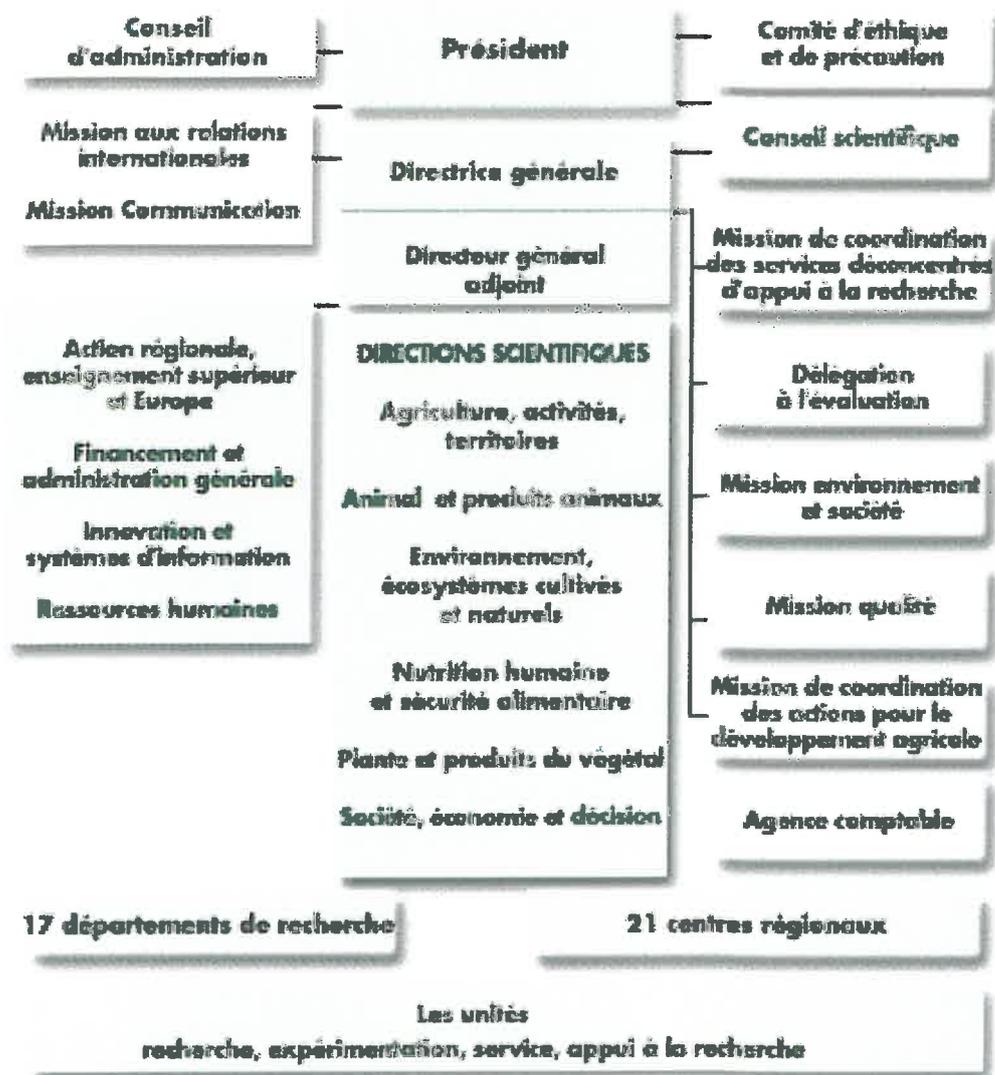


Figure 1 : organigramme INRA

1.2 Le centre d'Avignon

Il a été fondé en 1953 et constitue un maillon important du dispositif de l'INRA tant par sa taille que par les thématiques de recherches qui lui sont confiées.

Sa vocation est aussi bien régionale que nationale. Son implantation vauclusienne lui a conféré traditionnellement une responsabilité vis à vis de l'agriculture méditerranéenne pour la qualité de la production maraîchère et fruitière et pour l'industrie régionale de la transformation agroalimentaire.

Les programmes scientifiques du Centre Inra d'Avignon lui confèrent également une forte implication dans la politique scientifique nationale de l'institut.

Deux pôles majeurs caractérisent le centre :

- Gestion de l'environnement, pour les territoires cultivés et la forêt méditerranéenne.
- Maîtrise de la qualité des produits cultivés et transformés pour la santé du consommateur.

Le centre compte en moyenne 570 agents permanents et 400 temporaires par an. Il dispose d'un budget annuel de 38 millions d'euros, salaires compris et publie annuellement 450 documents.

Le centre compte 13 unités de recherche et 6 unités expérimentales réparties sur 3 régions, majoritairement en PACA (cf. figure 2)



Figure 2 : les implantations du centre d'avignon

1.3 L'unité CSE (Climat, Sol et Environnement)

L'INRA est structuré en départements de recherche, construits sur la base des disciplines scientifiques nécessaires pour couvrir tous les besoins de l'agriculture.

Cette structure se retrouve sur le centre d'Avignon par l'existence des « unités », qui sont sous la tutelle d'un ou plusieurs de ces départements.

L'unité CSE, dirigée par André Chanzy, sous la tutelle du département Environnement et Agronomie, est une jeune unité (2ans) puisqu'elle est le « résultat »

de la « fusion » des deux anciennes unités Science du sol et Bioclimatologie. Elle mène des travaux de recherche sur :

- La description des transferts de masse (eau, gaz, particules) et d'énergie dans le continuum nappe sol plante atmosphère.
- Le couplage des transferts de masse dans le sol avec les cycles bio-géochimiques.
- Le développement de la végétation des écosystèmes cultivés en relation avec le climat, les propriétés du sol et les pratiques agricoles.

Ces travaux ont pour objectifs finalisés :

- La quantification de l'impact environnemental des pratiques agricoles et des épandages de déchets sur la qualité des eaux (eaux de nappe en particulier) et des sols.
- L'optimisation des ressources pour les cultures (irrigation, fertilisation, agriculture de précision) et la mise au point d'itinéraires techniques préservant l'environnement.
- La prévision des rendements et la délimitation des potentialités des zones de production agricole en fonction du complexe agropédoclimatique ou des changements climatiques.

L'unité fonctionne avec 50 agents permanents dont 20 chercheurs, 8 ingénieurs, 14 assistants ingénieurs / techniciens, 3 agents techniques, 3 secrétaires, 3 documentalistes et 14 doctorants.

Elle est composée de 4 équipes : Couplage entre transferts et cycles bio-géochimiques – Transfert de masse et d'énergie dans le sol – Télédétection – Fonctionnement de l'espace agricole.

1.4 L'équipe TMSH (Transferts de Masse en Sols Hétérogènes)

Les travaux de l'équipe s'attachent principalement à la compréhension, à l'analyse et à la mesure des flux de masse et d'énergie (eau, solutés, particules, chaleur) dans les sols et aux interfaces sol-plante et sol-atmosphère. Ils portent aussi sur des développements méthodologiques (modèles, techniques d'estimation de paramètres) et métrologiques (capteurs, systèmes de mesure) associés aux processus étudiés. Ces travaux visent à une

meilleure prise en compte de l'hétérogénéité du sol (naturelle ou induite) dans la modélisation des transferts.

L'équipe gère aussi un site atelier dont l'objectif est une meilleure prédiction de la pollution de nappes souterraines à l'échelle de la parcelle agricole.

Les axes de recherche sont:

- Transfert d'eau, de particules et de polluants de la surface du sol vers les nappes souterraines
 - Partage des flux de masse à la surface du sol en fonction des états de surface et de leur évolution (Stéphane Ruy)
 - Ecoulements préférentiels d'eau et de solutés, lessivages rapides de polluants (L. Di Pietro)
 - Transfert des particules et des éléments traces métalliques associés (B. Cabibel)
 - Modification de la constitution et conséquences sur les transferts hydriques et thermiques (J.C. Fiès)

- Mécanismes de transferts en relation à l'interface sol-plante
 - Transfert d'eau dans le système sol-racines (C. Doussan)
 - Croissance racinaire - mécanique du sol (A. Faure)

- Développements méthodologiques et métrologiques
 - Méthodes géophysiques pour le suivi des écoulements en temps réels (tomographie électrique, potentiels spontanés, capteurs résistifs) - (C. Doussan ; R. Pallut ; S. Ruy ; L. Di Pietro - J.C. Gaudu)
 - Méthodes de caractérisation des propriétés hydrodynamiques (S. Ruy ; C. Doussan ; R. Pallut ; L. Di Pietro)
 - Méthode photogrammétrique pour le suivi de la détection superficielle d'eau (S. Ruy)

2 Contexte de la mission

2.1 *Origine des besoins*

A l'INRA d'Avignon, l'équipe TMSH avait besoin à l'origine de prévoir les transferts (écoulement et évaporation) de l'eau dans le sol. Cela leur permet par exemple d'en déduire les recharges des nappes phréatiques consécutives à une pluie, les ruissellements qui en découlent, ou l'arrivée d'une éventuelle crue lorsque le sol est saturé. Ces études, pour la plupart effectuées dans des organismes de recherche, sont importantes en agronomie, car elles permettent de fournir des conseils précieux aux agriculteurs quant à l'irrigation de leurs terrains. Elles peuvent aussi renseigner des organisations écologiques sur la rapidité du transport par l'eau d'un polluant dans le sol et donc vers la nappe phréatique.

Les chercheurs ont souvent besoin dans leur travail d'avoir recours à la modélisation (conception d'un modèle, c'est-à-dire d'un schéma représentatif choisi en fonction de son utilisation envisagée, suivie de l'élaboration d'un simulateur du système) pour prédire des événements physiques ou pour vérifier une loi expérimentale.

Par exemple, l'équipe TMSH a développé un modèle permettant d'estimer les variations d'humidité dans un sol. Ce modèle est décrit par une équation différentielle non linéaire dite « équation de Darcy-Richards ». Pour la résoudre, il faut connaître les propriétés du sol, en particulier sa courbe de perméabilité (capacité du sol à transmettre l'eau) et sa courbe de rétention (capacité du sol à stocker l'eau). Elle se résout numériquement de façon itérative à l'aide d'un algorithme mis en œuvre dans un programme Fortran.

Ce langage reste un langage très utilisé dans le domaine du calcul numérique par l'industrie et le monde de la recherche. La raison principale est les milliers de bibliothèques contenant des millions de lignes de code Fortran qui ont été développées, testées et validées depuis plus de 30 ans (la première version date de 1954). C'est un investissement lourd qu'il faut continuer d'exploiter.

Les modèles codés directement en Fortran sont adaptés aux chercheurs qui les créent et les utilisent. Mais ces scientifiques, plus souvent formés aux mathématiques qu'à l'informatique, délaissent l'aspect convivial des saisies et ne codent pas souvent avec rigueur.

Depuis un an, l'équipe TMSH a décidé de valoriser son expertise scientifique en développant et en diffusant une application conviviale dénommée ES.PA.S. (ESTimation des PAramètres du sol pour Simuler les transferts d'eau) permettant :

- d'une part de déterminer les paramètres hydrodynamiques d'un sol à partir de mesures expérimentales spécifiques,
- d'autre part d'utiliser ces paramètres dans la simulation de l'écoulement de l'eau dans le sol.

Les différents programmes permettant soit le traitement de données expérimentales pour la détermination des paramètres, soit la simulation existent : ils sont validés par l'expérience de l'équipe et relativement figés. Par contre la convivialité de l'interface et la réunion de ces différents programmes dans une même application ne sont pas encore réalisées.

Le stage s'insère dans ce projet et concerne spécifiquement le programme de simulation. L'objectif principal est de développer une IHM qui donnera à l'utilisateur la possibilité de rentrer les différents paramètres nécessaires à la résolution de l'équation. L'IHM doit aussi permettre de visualiser les résultats de façon graphique.

2.2 Définition de la mission

Le projet informatique a pour but d'encapsuler de manière graphique les trois programmes Fortran qui constituent le logiciel ES.PA.S. afin d'améliorer les conditions d'utilisation de ce dernier. Logiquement, le travail a donc été divisé en trois parties, chacune s'architecturant autour des modules Fortran existants :

- Le premier met en œuvre une méthode particulière de détermination des paramètres hydrauliques qui permettent par la suite de prévoir les mouvements d'eau dans les sols. Ces deux paramètres sont la courbe de perméabilité et la courbe de rétention d'eau dans le sol. Cette méthode, dénommée « méthode de WIND », est basée sur un dispositif expérimental spécifique et sur un algorithme particulier.
- Le deuxième permet de simuler l'évolution de l'humidité en résolvant l'équation qui décrit mathématiquement les mouvements d'eau dans le sol (équation dite de « Darcy-Richards »). Résoudre cette équation nécessite de connaître les paramètres hydrauliques du sol obtenus par exemple à l'aide de la méthode WIND (cf paragraphe précédent) ou grâce à une méthode inverse (cf paragraphe suivant)
- Le dernier met en œuvre une autre méthode particulière. Cette méthode appelée « méthode inverse » permet de retrouver par approximations successives les paramètres du sol à partir de données expérimentales diverses mais non spécifiques à un banc de mesure dédié, contrairement à la méthode de WIND.

Deux étudiants de l'IUP d'Avignon avaient déjà réalisé l'interface de la méthode WIND, mon travail a donc consisté en l'analyse, la conception et la réalisation de l'IHM du programme qui permet de résoudre l'équation de « Darcy-Richards ». Les contraintes de départ étaient donc doubles : il fallait intégrer mon travail autour d'un programme Fortran existant, mais aussi tenir compte de l'IHM développée l'année précédente.

Le but de la partie qui m'a été confiée était de faciliter l'utilisation du modèle de transfert d'eau dans les sols au niveau des entrées/saisies (remplissage des fichiers d'entrées laborieux) puis des visualisation/sorties (présenter des résultats sous une forme graphique mieux exploitable et directement lisible) pour des gens ayant peu de temps pour appréhender rapidement le fonctionnement de l'application ou une faible

connaissance du monde Unix et de la programmation Fortran (les stagiaires par exemple).

Mon stage a donc consisté en la réalisation d'une IHM à savoir :

- 1 – Analyse de l'existant et des besoins
- 2 – Conception
- 3 – Développement

Un planning prévisionnel a donc été réalisé pour estimer le temps de travail et se fixer des objectifs dans le temps.

2.3 Planification

Ce travail a fait l'objet d'une présentation lors d'une réunion avec l'équipe informatique. J'ai tout d'abord prévu de consacrer un tiers du temps à l'analyse et conception, un tiers à la réalisation et le reste pour la rédaction de documentations et les tests.

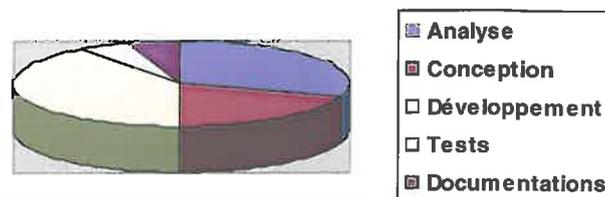


Figure 3 : Planning prévisionnel

J'ai malheureusement été contraint de revoir mes ambitions à la baisse et de réajuster le planning suite à un problème de santé. J'estime le temps perdu à cause de ces ennuis à environ un mois et demi à deux mois. J'ai donc réajusté mon planning en collaboration avec mon maître de stage en mettant l'accent sur l'analyse, la conception et une seule partie de la réalisation.

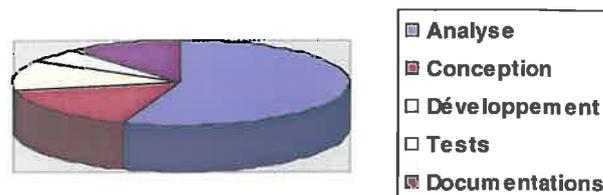


Figure 4 : Planning effectif

2.4 Organisation

Mon responsable étant très occupé, j'ai principalement travaillé de façon autonome. Nous nous sommes mis d'accord sur le fait de gérer mon travail comme je l'entendais, tout en lui montrant les différents documents mais nous essayions toutefois de faire un point sur mon avancement toutes les semaines ou tous les 15 jours et tous les mois avec Nathalie Moitrier, chargée d'une partie du développement informatique de l'unité.

De cette manière, je passais le voir régulièrement pour lui montrer les différents documents réalisés et discuter des choix faits. On se voyait également quand je me trouvais face à un problème qu'il pouvait m'aider à résoudre (par exemple sur les problèmes de compréhension au niveau scientifique du modèle).

La présence sur le site est obligatoire de 10h00 à 12h00 et de 14h00 à 16h00. Ensuite chaque collaborateur gère son temps de travail en fonction de ses contraintes ou de ses envies personnelles. Cela permet à chacun de travailler à son rythme, de ne pas avoir à interrompre le cours de son raisonnement, mais aussi de moduler ses horaires selon ses activités extérieures. Ce système permet de gérer son temps de travail à la journée et à la semaine de façon à arriver à 35 heures par semaine. De cette manière, on peut faire 8 ou 9 heures de travail dans une journée et faire seulement 6 heures le lendemain. J'ai pu ainsi profiter de ce système pour me rendre à des rendez-vous médicaux hebdomadaires, tout en essayant de ne pas trop perturber mon travail.

L'INRA ayant proposé plusieurs stages au sein de l'unité CSE, nous étions quatre étudiants de l'IUP. Nous avons essayé de nous voir régulièrement afin d'échanger nos points de vue au niveau de l'analyse, de la conception ou de la réalisation de nos projets. Ces rendez-vous étaient assez informels mais nous permettaient de trouver rapidement des éléments de réponses grâce aux connaissances de chacun.

Nous avons aussi réalisé deux réunions avec le département informatique de l'INRA dans lesquelles nous expliquions l'état d'avancement de nos projets.

2.5 Moyens matériels et logiciels

L'INRA étant une structure publique ne disposant pas de moyens financiers importants pour ses stagiaires, j'ai réalisé mon stage sur un ordinateur dépassé technologiquement (Pentium II à 233 Mhz avec 128 Mo de mémoire) qui servait simplement de terminal pour se connecter sur un serveur plus puissant. Le serveur quant à lui était un Bi-Athlon cadencé à 2 Ghz et soutenu par 2 Go de RAM.

Tous les ordinateurs du groupe de travail INRA2000 sont reliés en intranet 100MB. Cette configuration de travail permet d'échanger facilement des fichiers entre les membres de l'équipe grâce au partage de dossier windows. Nous disposons aussi de deux imprimantes laser qui fonctionnent en mode réseau autonome.

Pour les connexions aux serveurs LINUX, j'utilisais principalement deux logiciels :

- FlashFXP : un client FTP en mode graphique sous Windows. FTP est un protocole d'échange de fichiers entre une machine client et une machine serveur. Il est distribué sous la licence shareware et est disponible en version d'évaluation 30 jours à l'adresse Web suivante : <http://www.flashfxp.com/>
- PuTTY : un client SSH qui tourne sous Windows. SSH est un protocole de connexion entre une machine client et une machine serveur. Il permet, comme telnet, de lancer des commandes sur un ordinateur distant, mais possède l'avantage de crypter les données échangées entre les deux machines. Il est distribué gratuitement, il est disponible à l'adresse Web suivante : <http://www.chiark.greenend.org.uk/~sgtatham/putty/>

Bien que l'INRA ait un système de sauvegardes des disques, j'ai effectué de façon hebdomadaire des sauvegardes personnelles qui contenaient tous les documents et les sources écrits depuis le début du stage. J'en ai profité pour versionner mes sauvegardes de façon à pouvoir revenir en arrière en cas de besoin. Je n'ai pas pu utiliser le système de gestion des versions de l'INRA car il est en complète restructuration.

3 Réalisation

3.1 Analyse

L'analyse a été la première mission qui m'a été confiée. Elle a été l'étape la plus importante et la plus longue de mon travail.

L'analyse a été assez complexe étant donnée la multitude de données à lire et à écrire dans les fichiers textes.

De plus, le caractère très scientifique de ces données m'a, au début, assez désorienté. Heureusement, j'ai pu me baser sur des tableaux récapitulatifs des données d'entrées et de leurs contraintes respectives, réalisés par mon maître de stage.

3.1.1 Méthode

L'analyse s'est déroulée en trois temps :

- J'ai tout d'abord cherché à comprendre comment fonctionnait le programme Fortran, quels étaient les différents paramètres et la structure des fichiers d'entrées/sorties (analyse de l'existant)
- ensuite, j'ai cherché à savoir quels étaient les besoins et les cas d'utilisation de l'IHM (analyse des besoins),
- pour finir, je me suis attaché à comprendre comment les stagiaires de l'année passée avaient organisé leur travail (analyse de l'IHM existante).

3.1.2 Analyse du programme Fortran

Actuellement deux méthodes sont utilisées pour se servir du modèle :

- L'utilisateur lance l'application X-Windows pour se connecter à une machine UNIX en mode graphique. Ensuite, il édite d'anciens fichiers ASCII (correspondants aux données d'entrée nécessaires au modèle) avec un éditeur de texte, puis lance le programme directement depuis une console. Pour finir, il faut qu'il édite les fichiers résultats générés par l'application et qu'il importe certains fichiers dans un programme appelé S-PLUS pour obtenir une visualisation graphique des résultats.

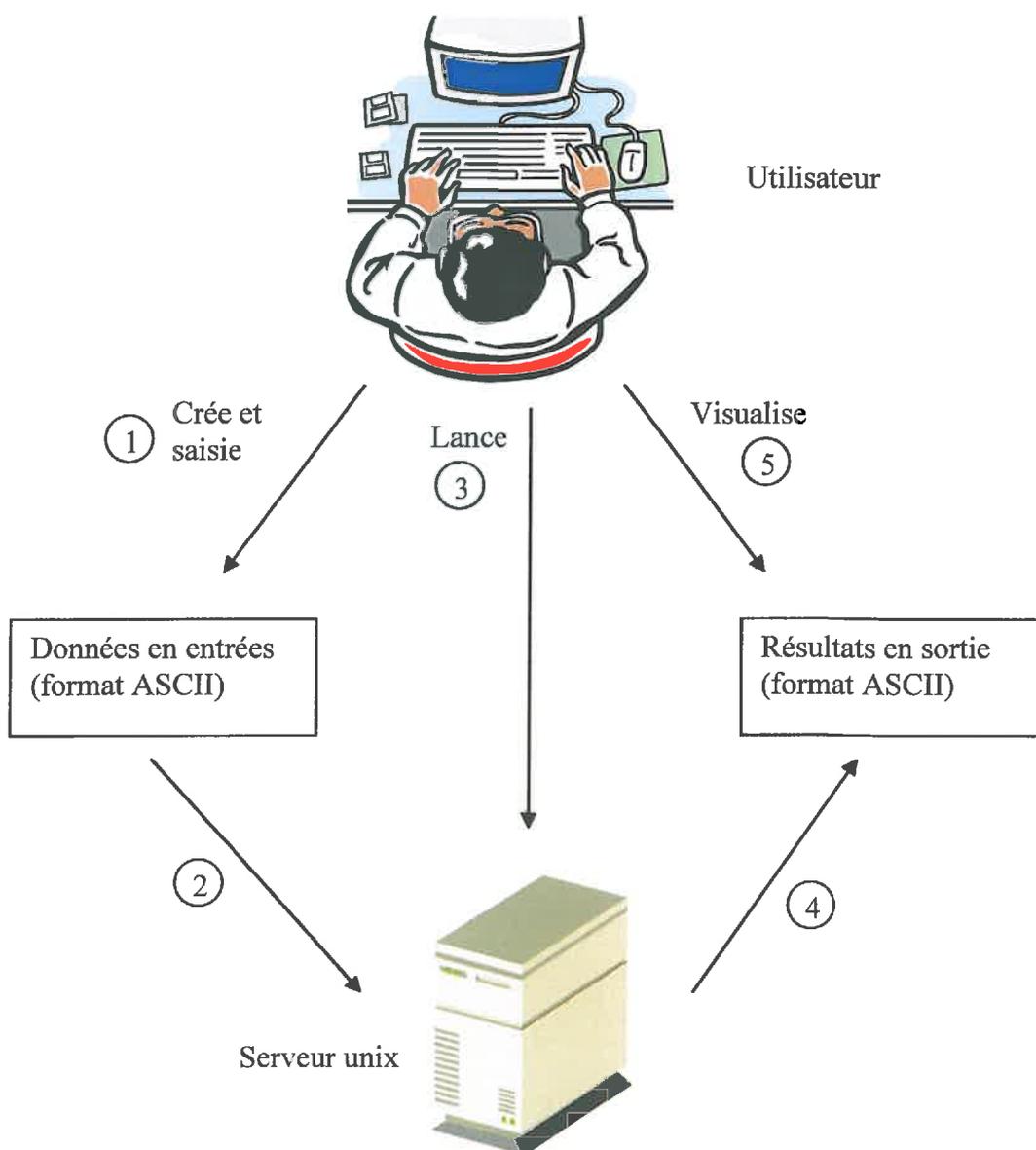


Figure 5 : Ancien fonctionnement

- A partir d'anciens fichiers ASCII (correspondants aux données d'entrée nécessaires au modèle), l'utilisateur saisit les valeurs qui permettront au programme de résoudre l'équation. Ensuite, il exporte ces fichiers via FTP sur un serveur UNIX et lance l'application Fortran (elle aussi située sur le serveur UNIX). Il peut pour finir visualiser les résultats en important via FTP les fichiers de sortie.

L'analyse du programme Fortran existant nous permet aussi de nous rendre compte de ses limites et des contraintes qu'il impose:

- nécessité de partir d'anciens fichiers ASCII,
- risques d'erreurs induits : mélange de paramètres, changement de la structure, ...
- vérification de la cohérence des différents paramètres non assurée,
- utiliser plusieurs outils afin d'éditer puis de visualiser ces résultats.

3.1.3 Analyse des besoins

Cette partie de l'analyse à consister à décrire les fonctions souhaitées de l'IHM et leurs contraintes dans différents scénarii tout en prenant en compte les dépendances entre les différentes étapes. Ainsi, certaines étapes ne sont accessibles que lorsque les étapes « mères » ont été validées et inversement, invalider certaines étapes va conduire à invalider toutes les étapes « filles ».

Pour modéliser tout cela, j'ai rédigé un document d'analyse contenant les scénarii (« use cases », cf figure 7) et des diagrammes reprenant ces cas d'utilisation pour avoir une vision plus globale des besoins.

Exemple de scénario :

Saisies des données (4)

Préalable : scénario « Nouvelle simulation (3) » ou « reprise d'un traitement » déroulé

Evènement déclencheur : déclenché automatiquement par le scénario « Nouvelle simulation (3) » ou lorsque l'utilisateur demande à y revenir

<p>Système :</p>	<p>Propose 7 choix. Un choix n'est possible que si le scénario précédent a été déroulé.</p> <ul style="list-style-type: none"> - Saisie de la description du système (6) - Saisie de la géométrie du système (7) - Saisie des propriétés physiques du système (8) - Saisie des conditions initiales du système (9) - Saisie des conditions limites du système (10) - Saisie des paramètres numériques du système (11) - Saisie des paramètres des observations (12)
<p>Utilisateur :</p>	<p>Choisit puis remplit les saisies les une après les autres Valide pour passer à la suivante, ou annule</p>
<p>Système :</p>	<p>Si un choix est effectué</p> <p style="padding-left: 20px;">Si la saisie précédente a été réalisée Déroule le scénario « saisie des ... » correspondant à l'option sélectionnée (6, 7, 8, 9, 10, 11 ou 12)</p> <p style="padding-left: 20px;">Sinon Informe l'utilisateur qu'il doit d'abord réaliser la saisie précédente</p> <p style="padding-left: 20px;">Finsi</p> <p style="padding-left: 20px;">Sinon Retour à « Nouvelle simulation (3) »</p> <p style="padding-left: 20px;">Finsi</p>

Au niveau des diagrammes, en accord avec le département informatique, je n'ai pas utilisé de méthode classique car elles sont mal adaptées à une lecture par des non-informaticiens. J'ai donc choisi une méthode plus compréhensible qui schématisait les différentes parties du programme et leurs liens respectifs. Déjà formalisée, cette méthode développée l'année passée permet aussi de garder l'homogénéité entre les deux projets. C'est un moyen de communication cohérent et compréhensible par tous.

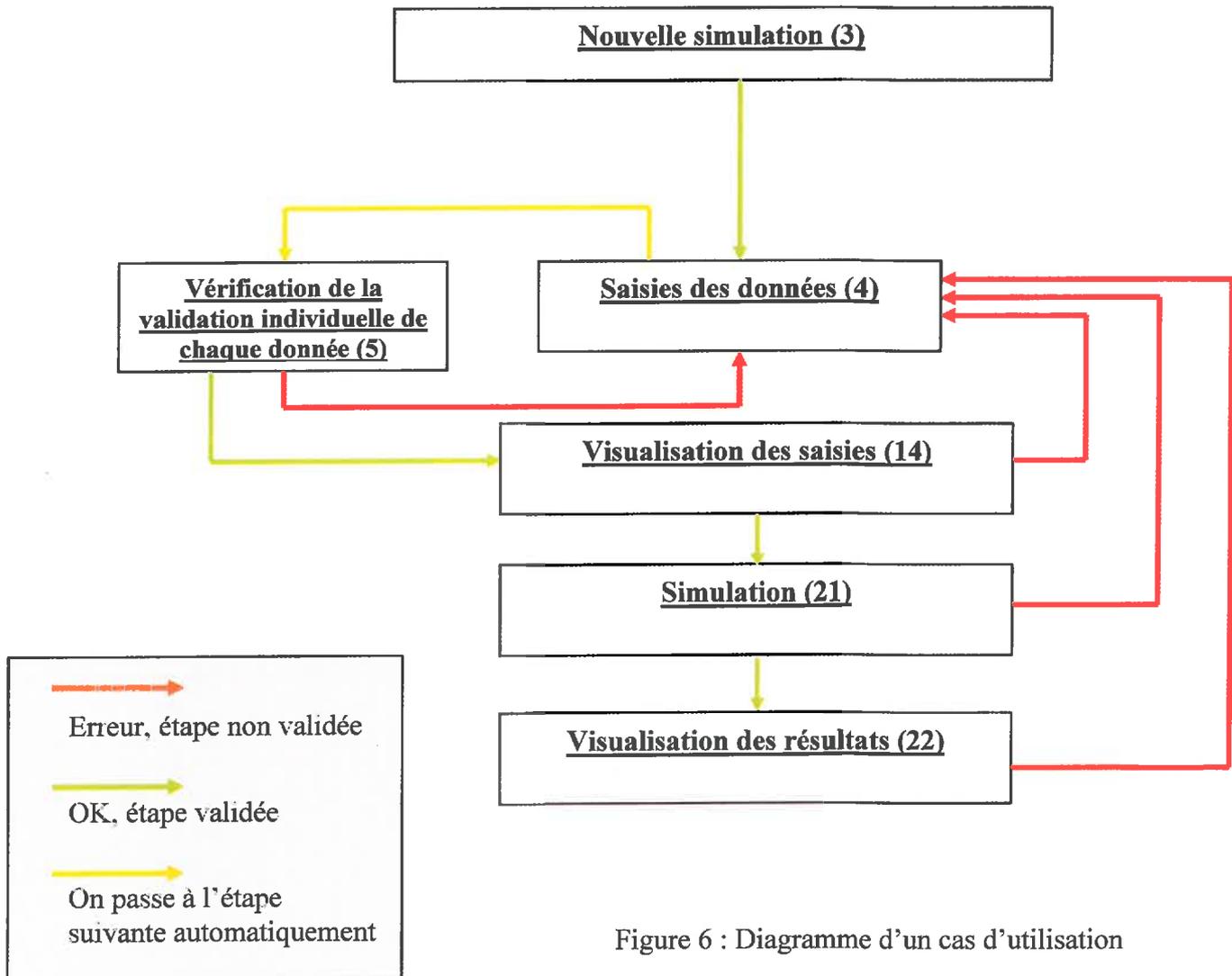


Figure 6 : Diagramme d'un cas d'utilisation

Voici un comment l'IHM s'interfacera avec le programme Fortran :

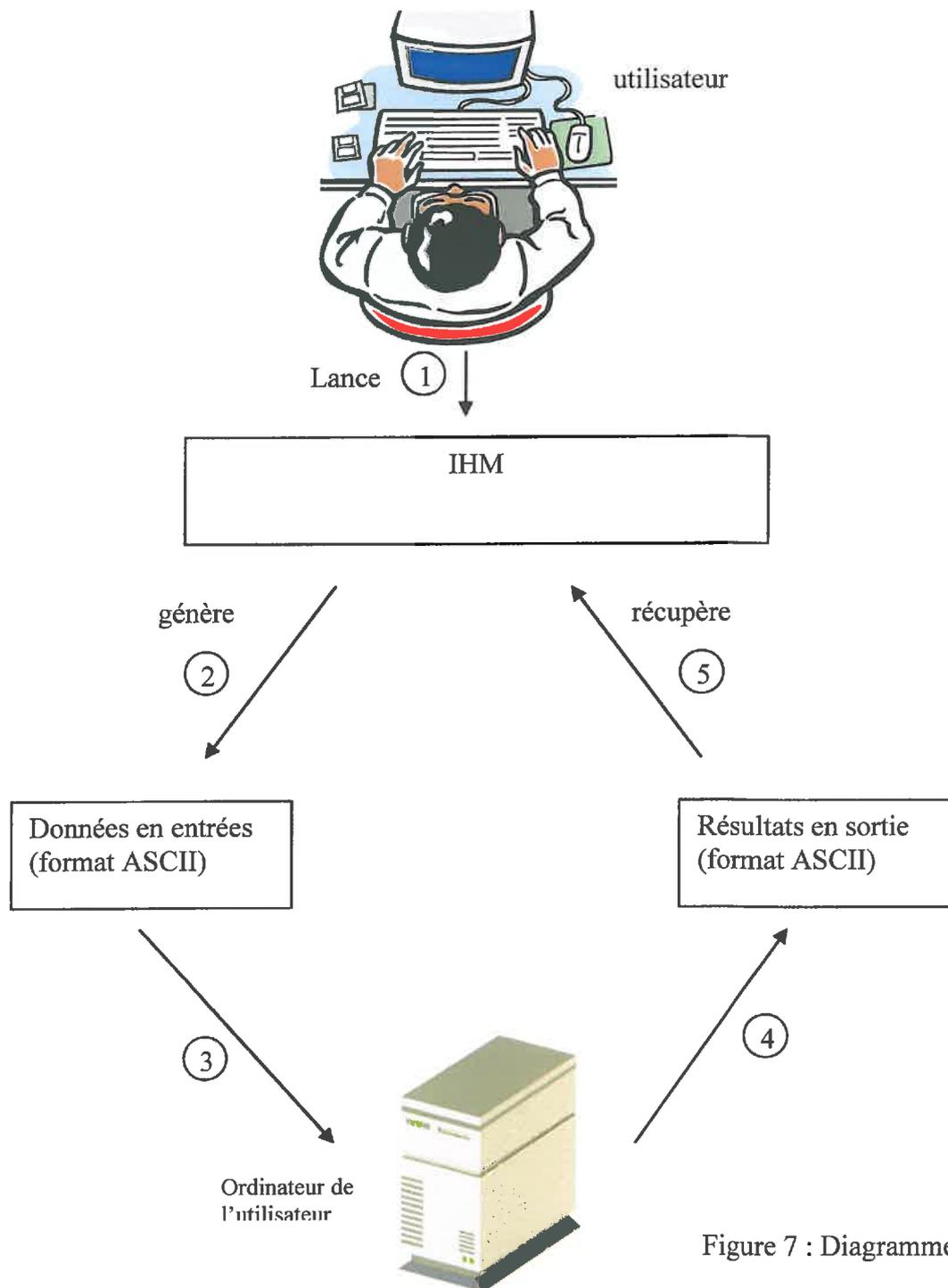


Figure 7 : Diagramme de classe

3.2 Conception

Le module d'interfaçage pour WIND ayant déjà été développé, je me suis basé sur les choix de conception qui avaient été fait à l'époque. Il a cependant fallu que je les comprenne et que je vérifie si ces choix étaient encore cohérents avec le travail qu'on me demandait de faire. Une analyse des autres solutions possibles a tout de même été faite.

3.2.1 Choix des langages de programmation

Au niveau du langage de programmation, le choix qui avait été fait l'année passée était le JAVA. J'étais donc persuadé de développer dans ce langage que j'affectionnais particulièrement. Cependant, à la première réunion avec le groupe informatique de l'INRA, je me suis aperçu que ce langage n'était pas le seul utilisable et qu'il fallait que je justifie ce choix. J'ai donc essayé d'autres langages comme le C++ avec les bibliothèques de développement WX-Windows. Cette solution est aussi très pratique car, même si elle demande une recompilation du programme pour chaque environnement, elle est portable sous tous les environnements et est plus rapide que le JAVA au niveau des temps d'exécution. Cependant pour mettre en œuvre ce langage, il aurait fallu tout re-développer et l'INRA n'avait pas les moyens de se lancer dans une telle aventure.

Le choix du XML et du XSLT au niveau du passage des paramètres a été directement dicté par la généricité induite par ces technologies mais aussi par la simplicité avec laquelle mon maître de stage pourrait effectuer des modifications ultérieurement.

XML est une évolution du langage SGML permettant aux concepteurs de documents HTML de définir leurs propres marqueurs, dans le but de personnaliser la structure des données qu'ils comptent présenter.

Le langage de transformation des données (XSLT, eXtensible Stylesheet Transformation) permet de transformer la structure des éléments XML. Un document XML peut être représenté comme une structure arborescente. Ainsi XSLT permet de transformer les documents XML à l'aide de feuilles de style contenant des règles appelées template rules (ou règles de gabarit en français). Le processeur XSLT

(composant logiciel chargé de la transformation) crée une structure logique arborescente (on parle d'arbre source) à partir du document XML et lui fait subir des transformations selon les template rules contenues dans la feuille XSL pour produire un arbre résultat représentant, par exemple, la structure d'un document HTML.

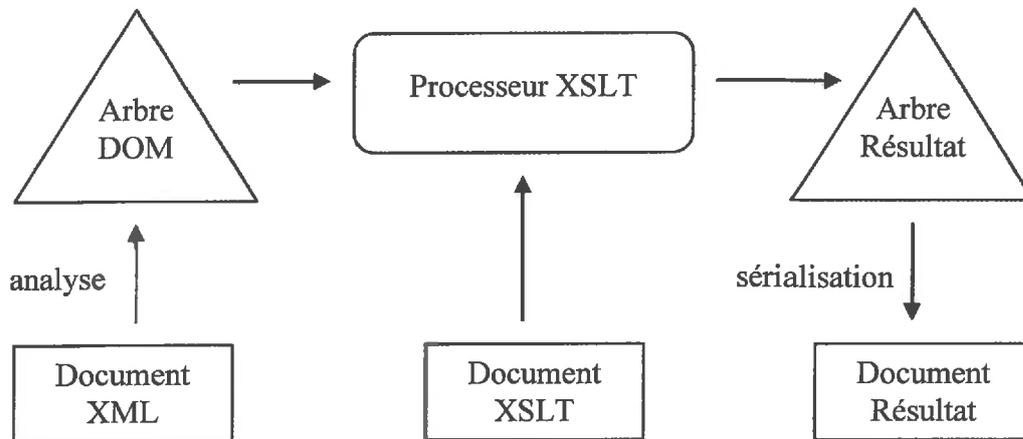


Figure 9 : Transformation de document XML en un autre type de document grâce à XSLT.

Cela suit le choix d'utilisation du XML qui est que le programme Java ne doit définir aucun des paramètres pour faire évoluer l'application plus facilement.

De plus, la traduction de ces fichiers est facilitée. En effet, une personne ne connaissant ni le Java ni le XSLT peut modifier ces feuilles de style pour les traduire.

Pour l'unité où j'ai effectué mon stage, ceci est un gros avantage car les chercheurs n'ont pas beaucoup d'informaticiens à leur disposition. Ils peuvent donc faire évoluer l'application eux-mêmes simplement avec quelques explications.

3.2.2 Choix des outils annexes

Pour la conception et les diagrammes de classes, j'ai utilisé Poseidon (www.gentleware.com) qui est un logiciel issu du projet OpenSource ArgoUML, instigué par Jason Robbins. GentleWare offre une version libre dite « Community Edition ». Cet atelier intègre toutes les fonctionnalités d'un modélisateur UML (Unified Modeling Language), à savoir le support total d'un langage de programmation (Java), le respect des diagrammes de la norme UML, le support du format de sauvegarde XMI, l'exportation des diagrammes sous différents formats (gif, ps, svg, eps), l'internationalisation de l'application dans différentes langues (dont le français), la

génération de code, la rétro-conception (« Reverse-engineering »), la fusion avec les EDI (Environnement de Développement Intégré) NetBeans et Sun Forte.

Pour la génération, un nouveau fichier Java est créé pour chaque classe ou interface à partir du modèle, en réutilisant les fichiers existants. Le code résultant présente un squelette de chaque classe et interface du projet.

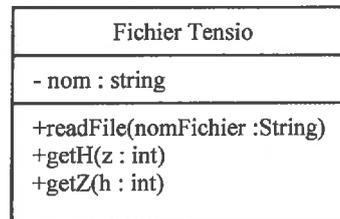


Figure 8 : Diagramme de classe

Pour le développement, j'avais toujours utilisé JBuilder de Borland qui s'est révélé être un bon outil mais qui est malheureusement trop coûteux à tous les sens du terme. Tout d'abord il est gourmand en ressources système et mémoire mais il est aussi assez cher. Le projet d'IHM n'ayant pas de financement spécifique pour acheter ce genre de logiciel, il a fallu partir à la recherche d'un autre environnement de travail, libre de droits. J'en ai trouvé un qui est très agréable à utiliser. Il s'agit de l'IDE Netbeans disponible à l'adresse suivante : <http://www.netbeans.org>. NetBeans, créé à l'initiative de Sun Microsystems (Noyau de Forte4J/SunOne). Il présente toutes les caractéristiques indispensables à un RAD (Atelier de Développement Rapide) Java de qualité. En licence OpenSource, NetBeans 3.5 permet de développer et déployer rapidement et gratuitement des applications graphiques Swing, des Applets, des JSP/Servlets, des architectures J2EE, dans un environnement fortement customisable. NetBeans possède un noyau robuste, et un système de pluggins performant.

Cet environnement est libre d'utilisation, il prend en charge de nombreux langages (Java, C, C++, XML et HTML) mais il est principalement développé pour les outils JAVA. Il a été écrit en JAVA, ce qui lui confère la possibilité d'être multi plate-formes : il peut être utilisé sous toutes les plates-formes pour lesquelles JAVA existe. Il intègre de nombreuses fonctions intéressantes qui facilitent le développement (<http://www.netbeans.org/products/ide/features.html>) dont la possibilité d' « auto-completion » du code JAVA et celle de visualisation rapide de la documentation des fonctions utilisées.

3.3 Implémentation

Le but de l'application était de fournir une surcouche graphique au programme Fortran qui réalise son travail grâce à de multiples fichiers d'entrée ASCII. Avant de lancer l'application Fortran, il fallait donc que l'IHM crée ces fichiers à partir des données saisies par l'utilisateur. En XSLT, il est donc possible de récupérer la valeur d'un nœud précis ou bien de réaliser un traitement sur un ensemble de nœuds grâce à des boucles.

Plusieurs méthodes ont été définies dans ce langage, par exemple pour faire des opérations comme les additions, soustractions, ... ou bien le formatage de nombres, le test de valeur, l'écriture de texte...

Voici un morceau d'un document XSLT que j'ai réalisé pour créer un des fichiers d'entrées du programme Fortran :

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet xmlns:"http://www.w3.org/1999/XSL/Transform" version="1.0">
  <xsl:output method="text" indent="no"/>
  <xsl:template match="TRAITEMENT">
    <xsl:text>
*****
Option d'impression de ce fichier (1 ou 0, i3)
    </xsl:text>
    <xsl:choose>
      <xsl:when test="ENSEMBLE[@id='impression']/CHOIX[@id='iooptimp']/VAL !="">
        <xsl:value-of
select="ENSEMBLE[@id='impression']/CHOIX[@id='iooptimp']/VAL"/>
      </xsl:when>
      <xsl:otherwise>
        <xsl:value-of
select="ENSEMBLE[@id='impression']/CHOIX[@id='iooptimp']/DEFAULT"/>
      </xsl:otherwise>
    </xsl:choose>
    <xsl:text>
*****
Choix des modalités d'ajustement des profils hydriques
1=Polynome+Spline 2=Van Genuchten 3=Logistique
*****
    </xsl:text>
    <xsl:choose>
      <xsl:when test="ENSEMBLE[@id='ajustph']/CHOIX[@id='
iopmodph']/VAL !="">
        <xsl:value-of
select="number(ENSEMBLE[@id='ajustph']/CHOIX[@id='iopmodph']/VAL)+1"/>
      </xsl:when>
      <xsl:otherwise>
        <xsl:value-of
select="number(ENSEMBLE[@id='ajustph']/CHOIX[@id='iopmodph']/DEFAULT)+1"/>
      </xsl:otherwise>
    </xsl:choose>
  </xsl:template>
</xsl:stylesheet>
```

Cette feuille de style appliquée au document XML qui contient tous les paramètres de l'utilisateur permet d'arriver au résultat suivant:

```
*****
Option d'impression de de fichier (1 ou 0, i3)
  1
*****
Option generale de l'utilisation des courbes de retrait par horizon i3 :
1= la description des courbes de retrait par horizon est utilisée
0= la description des courbes de retrait est ignorée
*****
  0
```

L'utilisation du XSLT permet, de même que le XML, une évolution plus simple puisque la mise à jour d'une feuille de style ne nécessite pas la compilation du programme Java afin que les modifications soient prises en compte.

Si le texte évolue, il suffira simplement de modifier les intitulés compris entre les balises <xsl:text>. Cette solution est idéale car même un non-informaticien peut la mettre en œuvre facilement.

3.4 Tests

Je n'ai pas eu de tests particuliers à effectuer car j'ai très peu développé. Cependant, après chaque réalisation d'un fichier XSLT, je vérifiais que la génération du fichier ASCII était correcte. J'ai donc fait beaucoup de tests unitaires à partir de fichiers que je créais pour l'occasion. Il faudra cependant que mon successeur fasse les tests d'intégration quand il intégrera mon travail au sein du projet ES.PA.S.

3.5 Problèmes rencontrés

Les problèmes rencontrés ont été surtout des problèmes de compréhension avec mon maître de stage. Je n'avais pas de connaissances en agronomie et lors de l'analyse des fichiers d'entrées/sorties je me suis souvent trouvé confronté à du vocabulaire que je ne comprenais pas. Il était alors indispensable de faire le point avec un chercheur afin qu'il m'explique le vocabulaire mais aussi qu'il me fasse entrevoir les phénomènes physiques qui étaient en jeu. Le problème est aussi apparu dans le sens inverse, c'est-à-dire qu'il a fallu que nous mettions en place un système de communication simple et pauvre en vocabulaire informatique avec les chercheurs. Nous avons décidé de réaliser des diagrammes très simples pour qu'ils puissent être compris par tous.

3.6 Avantages et inconvénients de l'IHM

A première vue, il semble évident que l'encapsulation graphique de tels outils ne peut avoir que de bons côtés : il permet de saisir de façon claire et précise les paramètres d'entrée, de vérifier leurs vraisemblances ou de déceler d'éventuelles erreurs de saisie mais aussi de visualiser graphiquement les données de sorties. Cependant, ce système n'est pas exempt de défaut car il est architecturé autour d'un code Fortran figé : ainsi, toute modification du code ou ajout d'une fonction dans un module entraînera inévitablement des modifications au niveau de l'interface JAVA.

Bilan du stage

Au début du stage, nous étions bien conscients que je ne pourrais pas finir tout seul en cinq mois tout ce qu'il restait à faire pour distribuer ES.PA.S. dans une version définitive. De plus, ce stage a été ponctuellement dérangé par mes problèmes de santé, je n'ai donc pas pu arriver jusqu'où je l'espérais. Toutefois, ma mission permettra, je l'espère, à un prochain stagiaire d'être soulagé de l'analyse, de la conception et d'une partie du développement. Les solutions techniques étant déjà mises en place, je pense que l'avancement pourrait être assez rapide, j'estime le temps restant à la réalisation à environ deux mois (avec la prise de connaissance des documents existants).

L'apport pour l'I.N.R.A. est assez important car ces projets de valorisation leurs permettront de diffuser plus largement tous leurs modèles mais aussi d'avoir des outils faciles à prendre en main pour des stagiaires ayant peu de temps pour s'adapter.

La vie en entreprise est très différente de celle de l'I.U.P. : même si j'en avais déjà fait l'expérience lors de mon stage de fin d'étude d'I.U.T., et lors de mon stage d'I.U.P. 2, il faut quelques temps pour s'habituer à travailler huit heures par jour sur le même projet. Même si le volume horaire est sensiblement identique à celui de l'I.U.P., les activités et les disciplines universitaires plus variées effacent un peu cette impression de routine.

Ayant effectué mes stages précédents respectivement dans une multinationale et une petite structure publique, je souhaitais cette fois découvrir un autre milieu : la structure publique importante. Cette connaissance de tous ces milieux est, je pense, indispensable : elle permet d'orienter ses futurs choix professionnels.

J'ai été très bien accueilli au sein de l'unité. Mon intégration a été d'autant plus rapide que je connaissais déjà trois autres stagiaires, qui étaient aussi des étudiants de l'I.U.P. en stage de troisième année.

Les difficultés que j'ai pu rencontrer ont été seulement d'ordre de compréhension technique, surtout au niveau des concepts agronomiques. Je n'ai pris que très peu de retard par rapport au calendrier initial si on tient compte du temps perdu à cause de mes ennuis de santé.

Arrivant à la fin de ma formation, je pense avoir fait le bon choix au niveau de mon orientation. L'I.U.T. m'a permis d'acquérir les compétences techniques qui m'ont été nécessaires tout au long de ce projet et l'I.U.P. a apporté de la rigueur et de l'organisation dans mon travail. Grâce à cela, je me suis senti opérationnel en arrivant à l'I.N.R.A.: je connaissais la plupart des outils et j'ai pu m'adapter très rapidement au travail que l'on me demandait.

Les différents types de projets (seul ou en groupe) qui ont jalonné mon parcours universitaire m'ont beaucoup apporté et c'est certainement cela qui a facilité mon intégration dans le groupe et m'a aidé à respecter les contraintes de temps qui m'étaient imparties.

Je trouve que notre formation est très efficace grâce à la diversité des cours qui y sont enseignés. Elle nous permet de nous adapter à tous les travaux informatiques, qu'ils soient orientés vers base de données, programmation, gestion ou communication et multimedia.

Ce stage a été pour moi une expérience très enrichissante, autant au niveau professionnel (acquisition de compétences importantes en analyse et en gestion de projet) qu'au niveau du travail dans un cadre public et hiérarchisé aux allures de « multinationale ».

J'ai pu mesurer, comme je l'espérais, la différence qui existe entre une multinationale (stage d'I.U.T.), un petit groupe de travail aux moyens plus limités (stage d'I.U.P. 2) et une structure publique importante. Le travail dans une multinationale est très ciblé et compartimenté : on a peu de marge de manœuvre et on ne peut pas prendre d'initiatives. Dans une petite structure, le travail demande un investissement colossal, mais il est plus gratifiant : on voit vraiment pour qui et pour quoi on réalise le projet. A

l'I.N.R.A., j'ai pu prendre des initiatives mais aussi voir les chercheurs pour qui je travaillais.

Je pense être plus attiré vers les grosses structures, qui permettent bien souvent d'avoir une meilleure qualité de vie : les heures sont comptabilisées et contrôlées, ce qui permet de ne pas « être happé » par un projet et de passer trop de temps dessus.

Pour finir, j'envisage de poursuivre mes études en DESS TAI à l'université d'Avignon car je souhaite acquérir une spécialisation qui me permettra, je l'espère, de mieux m'insérer dans le milieu professionnel que je souhaite.