

Projet Ingénieur n° 6

Modélisation d'un cube de données d'observations astronomiques

Création d'une **applet java** pour simuler les représentations d'un
cube de données.

Client :

Pr. J. Köppen
de l'Observatoire Astronomique de Strasbourg

Equipe projet :

Bertrand Leriche
Cindy Le Loirec
Marc Olivier Sercki
Sophie Bresson

Rapport public
Première édition
12 Mai 2003

Résumé

Les observations astronomiques d'objets à émission (galaxies, nuages gazeux...) ne permettent pas de déterminer de façon immédiate la forme et la cinématique en trois dimensions de ces objets. En effet, on n'observe que des images projetées (donc en deux dimensions), ainsi que la vitesse radiale (en une dimension) grâce à l'effet Doppler. On n'a alors accès qu'à l'intensité en fonction de ces trois grandeurs, ce qui forme le cube de données.

C'est pourquoi, à la demande du Pr. Köppen de l'Observatoire Astronomique de Strasbourg, nous avons choisi de réaliser un outil pédagogique qui permette, pour les étudiants, de se familiariser avec l'interprétation des observations en manipulant les objets et pour les chercheurs, de tester des modèles, i.e. de vérifier leur conformité avec les observations. Cette réalisation s'inscrit dans le cadre de notre projet ingénieur de 2ème année pour l'ENSPS.

Le produit final est une applet Java interactive et pédagogique, qui à partir d'un modèle géométrique et cinématique d'un objet astronomique, simule les représentations courantes que peut en obtenir un observateur réel. Cette applet s'inscrit plus globalement dans un site Web dont le développement fut une deuxième part de la gestion de ce projet et qui présente en plus de l'applet une aide en ligne sur son fonctionnement.

Abstract

It is not always easy to determine precisely the 3D-geometry and the kinematics of emitting astronomical objects such as galaxies or stellar clouds given only the data cube. This cube, consisting in a two-dimensional representation and the radial velocity (e.g. three parameters) is all astronomers receive from space observations.

Given that, we decided to create, on request of Pr. Köppen from the Astronomical Observatory of Strasbourg, an educational tool aimed at students and astrophysicists. This tool will make it possible for them to get familiar with manipulation of objects and to understand the representations. But it is also designed for astronomers who will have the opportunity to test their models by simulating the representations they could obtain with real observations. Therefore they will be able to compare their models with the ones that originated the representations given by our tool.

We carried out this project as our 2nd year engineering project for the ENSPS. Eventually, we have obtained a Java Applet simulating various representations of an object, given its geometry, density and kinematic behaviour. Besides, this applet is part of a web site describing its functions and offering online help.

I. Introduction

Dans notre ciel, ces points sont aussi petits que des grains de sable voire même invisibles et semblent simples à analyser pour un observateur quelconque. Ce n'est cependant pas le cas. En effet, ces points que nous observons ne sont que des projections à deux dimensions d'objets astronomiques tridimensionnels: il nous manque donc une partie des informations, à priori, pour pouvoir les reconstituer. Cependant, d'autres données nous sont fournies indirectement par l'effet Doppler : c'est la vitesse radiale. Nous obtenons ainsi ce que nous pouvons appeler un « cube de données ».

La cartographie à haute résolution spatiale et spectrale des objets à émission est le moyen le plus riche qui nous permet d'obtenir ce cube de données. Il semble donc possible de constituer un modèle, grâce à ces informations, des objets astronomiques observés. Cependant cette modélisation est délicate pour les chercheurs et d'autant plus pour les étudiants. Le développement d'un outil de simulation est donc nécessaire pour leur faciliter ce travail, et permettre aux étudiants de se familiariser avec l'interprétation des observations astronomiques.

Le but de notre projet a donc été de développer cet outil à travers une applet interactive et pédagogique qui à partir d'un modèle géométrique et cinématique d'un objet astronomique (c'est-à-dire à partir d'un cube de données), simule différentes représentations que pourrait en obtenir un observateur réel.

Les fonctionnalités précises de notre logiciel sont rappelées dans la formulation du cahier des charges qui présentera aussi l'évaluation des différents coûts que nous avons prévus. Puis l'évolution précise du projet sera développée et analysée. Accompagnée d'un exemple d'utilisation de l'applet, une présentation complète de ses possibilités sera également faite. Nous concluons alors sur les développements et optimisations possibles.

II. Sommaire

Résumé	2
Abstract	2
I. Introduction	3
II. Sommaire	4
III. Résumé du cahier des charges	5
A. Définition du produit	5
B. Interlocuteurs, acteurs	5
C. Fonctionnalités du produit, exigences particulières	5
D. Evaluation (fictive) des coûts	6
E. Evaluation des délais	6
IV. Gestion du projet, bilan	7
A. La gestion au cours du temps	7
1. Attribution des tâches	7
2. Planification et déroulement du projet	7
3. Rencontres avec les différents intervenants	9
4. Budget	9
B. Analyse et perspectives	10
1. Difficultés rencontrées et solutions	10
2. Perspectives d'avenir	11
V. Description détaillée de l'applet	12
A. Description générale	12
1. Fonctionnement de l'applet	12
2. Description générale de l'interface	13
B. La création d'un modèle	14
1. Structure d'un modèle	14
2. Les repères	15
3. Le panneau « groupe de modèles »	16
4. Le panneau « modèle »	16
5. Les panneaux « loi »	17
C. Le générateur	18
D. Les représentations	18
1. Le panneau de choix des représentations	18
2. Les panneaux des graphes	19
E. Exemple concret	20
VI. Site Web	25
VII. Conclusion	26
VIII. Annexes	27
Annexe A : Glossaire	27
Annexe B : Bibliographie	28
Annexe C : Contenu du rapport industriel	28
Annexe D : Transparents de R6	28

III. Résumé du cahier des charges

A. Définition du produit

Le produit est une applet interactive et pédagogique, qui à partir d'un modèle géométrique et cinématique d'un objet astronomique, simulera différentes représentations que pourrait en obtenir un observateur réel.

B. Interlocuteurs, acteurs

Ce projet est mené à la demande du Pr.KÖPPEN Joachim de l'Observatoire de Strasbourg, le client. L'équipe projet chargée de le mener à son terme est constituée de :

LERICHE Bertrand chef de projet

LE LOIREC Cindy responsable de la communication

BRESSON Sophie responsable de la documentation et du budget

SERCKI Marc-Olivier responsable technique

Cette équipe est encadrée au sein de l'ENSPS par le Pr.GOLTZENE Alfred. Elle est encadrée dans son exercice par Mr.CHRETIEN Jean, le Pr.BALLADORE Jean-Louis, et le Pr.SZYCHOWIAK Pyotr ; elle est également évaluée dans sa démarche de gestion de projet par Mr.BAUER Jacques et Mr.FLECK Claude.

C. Fonctionnalités du produit, exigences particulières

Les adjectifs suivants devront être applicables au produit :

- ✓ **Evolutif** : l'architecture de ce programme doit être suffisamment adaptable pour permettre d'ajouter ultérieurement de nouveaux modèles ainsi que de nouvelles représentations avec le minimum possible de modifications; dans cette optique, la méthode de simulation à implémenter est celle dite de Monte Carlo.
- ✓ **Flexible** : les différents modèles doivent pouvoir être combinés aisément afin d'en créer de plus complexes basés sur ceux existants.
- ✓ **Opérationnel** : ces conditions réalisées, il faut également implémenter quelques modèles et représentations afin notamment de démontrer la fonctionnalité de l'applet mais aussi de pouvoir en commencer l'exploitation à des fins pédagogiques.
- ✓ **Convivial** : étant donné l'application visée pour cette applet, elle doit être la plus simple d'utilisation possible, tout en permettant de tester des configurations variées au sein d'un modèle.
- ✓ **Interactif** : les interactions avec l'utilisateur passeront par une interface graphique, aussi bien pour l'acquisition que pour la représentation des données ; la simulation doit être conçue pour que l'utilisateur puisse décider en cours de calcul si elle doit être interrompue.

D. Evaluation (fictive) des coûts

Cette évaluation est faite dans l'hypothèse où tout service serait effectivement facturé.

Le matériel et logiciel nécessaires au projet sont les suivants :

- ✓ un ordinateur par personne : 4*1500€ soit 6000€,
- ✓ le JAVA Software Development Kit v.1.05 dit 'distribuable', dont la licence autorise gratuitement l'utilisation normale,
- ✓ la documentation diverse, sur différents supports dont le coût ne saurait dépasser 500€.

Par ailleurs, la location des locaux coûterait environ 3000€(6mois à 500€).

Le budget matériel s'élèverait donc à 9000€ (cf. partie budget pour plus de détails).

E. Evaluation des délais

De façon très grossière, il faudra environ, à partir du 24 octobre 2002, date d'attribution du sujet :

- ✓ 3 semaines de définition du projet,
- ✓ 1 mois et demi de recherche documentaire, évaluation de l'existant, identification des tâches, mise en place du planning de développement, définition de la structure de l'applet, apprentissage des spécificités à la programmation JAVA,
- ✓ 2 à 3 mois de développement de l'applet : définition de tests à effectuer en cours de développement, interface/acquisition, simulation, interface/représentation, un modèle,
- ✓ 1 mois et plus : implémentation de propriétés supplémentaires au modèle, de représentations supplémentaires, tests et corrections.

Dans ces estimations, il serait possible de fournir un prototype fonctionnel dans le courant de mars, et une version définitive en mai 2003(cf. partie planning pour plus de détails).

IV. Gestion du projet, bilan

A. La gestion au cours du temps ...

1. Attribution des tâches

La répartition du travail s'est faite dès les premières semaines de l'activité projet, principalement en suivant les préférences et aspirations de chacun. De plus le découpage en modules de la partie programmation nous a aidé pour la mise en place du travail.

Chacun a donc pu occuper le rôle qui lui convenait et s'épanouir dans sa partie du projet.

2. Planification et déroulement du projet

Le projet s'est déroulé du 24 Octobre 2002 au 12 Mai 2003. Durant ces 7 mois d'activité, des réunions (revues de projet) étaient programmées pour nous permettre de présenter les dernières avancées et ainsi nous avons bénéficié de points de repères précieux pour rester à jour.

De plus nous avons suivi plusieurs réunions de Gestion des Ressources Humaines (GRH) avec des consultants pour nous aider à mieux gérer la partie humaine du projet. Ces réunions, ayant lieu avant chaque revue de projet, nous avons donc l'opportunité de nous y préparer en relevant les points de difficulté rencontrés. Ces revues de projet nous ont également permis de faire des bilans intermédiaires et nous avons pu rester à jour dans notre planification initiale du travail.

Concernant le planning, nous avons agi de manière méthodique. Nous nous sommes tout d'abord consacrés à la reformulation du cahier des charges afin de partir sur de bonnes bases, puis nous nous sommes partagés le travail ce qui s'est révélé très simple étant donné la forme modulaire du projet. Durant trois mois nous avons programmé chacun notre partie puis nous avons procédé à la mise en commun des différentes parties de code. Depuis fin mars, nous avons repris une activité de groupe à proprement parler puisque nous travaillons ensemble afin d'optimiser l'applet. La gestion du temps s'est avérée efficace même si la mise en commun a été délicate.

Pour mieux comprendre comment nous avons réparti le travail cette année, nous présentons sur la page suivant le planning général qui comporte les grandes étapes de notre activité.

Voici le planning pour la période du 24 Octobre 2002 au 26 Mai 2003 avec tout d'abord le nom et la durée des différentes tâches :

Nom de la tâche	Durée	Début	Fin
définition du projet	15 jours	Jeu 24/10/02	Mer 13/11/02
mise en place	31 jours	Mer 13/11/02	Mer 25/12/02
programmation des modules	49 jours	Mer 25/12/02	Lun 03/03/03
générateur	32 jours	Mer 25/12/02	Jeu 06/02/03
modélisation	45 jours	Mar 24/12/02	Lun 24/02/03
page web	32 jours	Ven 10/01/03	Lun 24/02/03
interface	44 jours	Mer 25/12/02	Lun 24/02/03
représentation	44 jours	Mer 25/12/02	Lun 24/02/03
mise en place du prototype	10 jours	Lun 03/03/03	Ven 14/03/03
développement du prototype	27 jours	Mar 25/03/03	Mer 30/04/03
rédaction des rapports	28 jours	Jeu 17/04/03	Lun 26/05/03

Tableau 1 : Planning du 24 octobre 2002 au 26 mai 2003

Puis voici le planning représenté par un diagramme de Gantt :

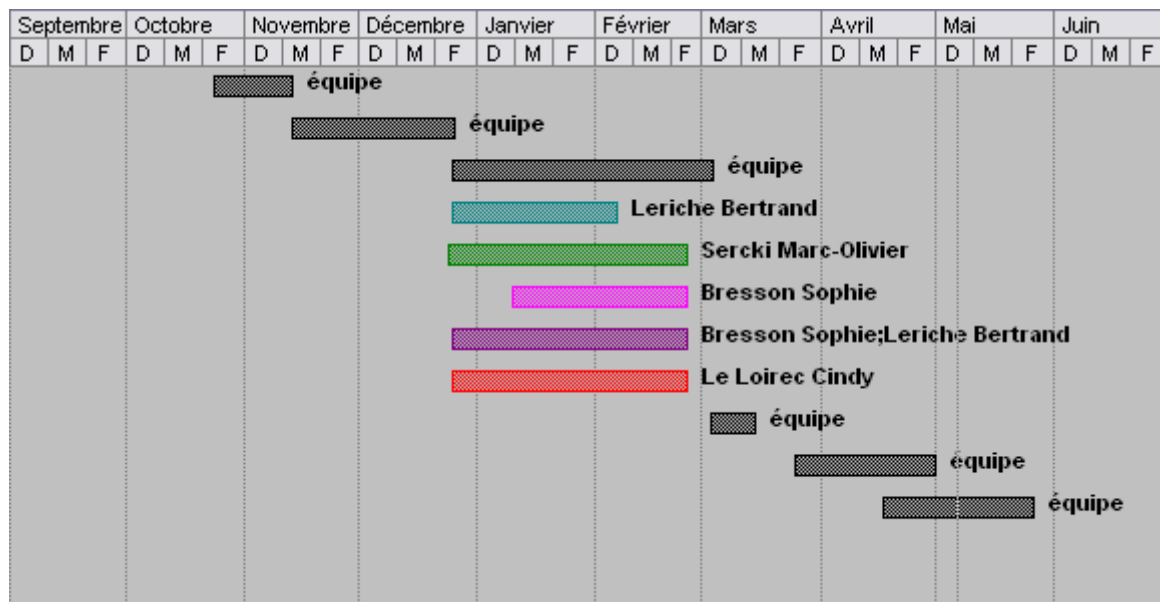


Tableau 2 : Diagramme de Gantt du planning

3. Rencontres avec les différents intervenants

Tout au long de l'année, nous avons maintenu un très bon contact avec notre client M. Köppen. Nous avons établi un premier contact par email dès le 30 octobre grâce auquel nous avons demandé un premier rendez-vous pour se rencontrer et connaître ses exigences. Ensuite et jusqu'en mai nous avons maintenu un contact permanent par email et par des rendez-vous à l'Observatoire de Strasbourg.

Nous avons eu avec lui quatre réunions au total qui ont duré en moyenne 3 à 4 heures et dont voici un bref résumé :

<i>1^{ère} réunion :</i>	7 novembre 2002	Au cours de cette entrevue nous avons fait connaissance, défini les points importants du produit final ainsi que ses futures applications. Nous avons également mis au clair les techniques numériques que nous allons utiliser.
<i>2^{ème} réunion :</i>	13 février 2003	Nous avons précisé les points qui nous posaient problème principalement dans la façon de programmer.
<i>3^{ème} réunion :</i>	27 mars 2003	Nous avons fait la démonstration d'un tout premier prototype, posé des questions sur le contenu de l'applet et nous nous sommes renseignés sur ce qu'il y avait à ajouter.
<i>4^{ème} réunion :</i>	24 avril 2003	Nous avons fait la démonstration d'un prototype quasi-définitif. La majeure partie de la rencontre a été consacrée à des tests et des ajustements mineur sur le code source.

Tableau 3 : Liste des réunions avec Mr. Köppen

Pour nous aider à gérer le projet, nous avons pris conseil à plusieurs reprises auprès de notre tuteur M. Goltzené. Il nous a notamment aidé dans le choix des méthodes numériques et aussi pour l'aspect présentation lors des revues de projet.

Enfin, au mois de décembre nous avons rencontré M. Leroy de l'ENSPS pour clarifier certaines questions au sujet de la méthode de Monte-Carlo.

4. Budget

La gestion de notre budget s'est conduite dès le départ de l'activité projet, en novembre 2002 pour s'achever en mai 2003. Cependant, étant donnée la nature essentiellement informatique du projet, nous n'avons pas eu à gérer de budget réel conséquent. Nous présentons donc dans le tableau suivant un budget découpé en deux parties; l'une réelle comprenant les frais effectivement engagés par les membres de l'équipe, et l'autre fictive présentant ce qu'aurait coûté ce projet s'il avait été conduit dans le cadre d'un travail en entreprise par exemple. Par rapport à ce qui a été annoncé en début d'activité, le budget final est sensiblement le même. Certains éléments ajoutés en cours d'année ont un peu augmenté le total.

Pour commencer nous présentons le budget sous l'angle partie imputable à l'école / partie imputable au client.

Voici donc les frais engagés par chacune des deux parties :

Partie imputable au client :	Frais de personnel	6 340 €
Partie imputable à l'école :	Frais matériels	18 000 €
Total		24 530 €

Tableau 4 : Frais engagés par les deux parties

Budget global :

FRAIS REELS		
Déplacements chez le client	4 visites en 7 mois	30
Internet	2 h par semaine	60
Téléphone	1/2 h par semaine	30
Fournitures	Papeterie, photocopies ...	20
Documentation	Livres sur Java,...	50
TOTAL		190 €

FRAIS FICTIFS		
Budget Matériel		
4 ordinateurs	Amortissement sur 7 mois	750
1 Imprimante laser	Amortissement sur 7 mois	900
Logiciels	Pack Office (4 licences)	740
Microsoft FrontPage	1 licence	270
Compilateur Java	JDK 1.05	0
Documentation	Livres, ...	500
Locaux	Une salle pendant 7 mois	3 000
Sous total		6 160 €
Budget Personnel		
4 Ingénieurs	30 € / heure	18 000
Sous total		18 000 €
TOTAL		24 160 €

TOTAL (en euros)	24 350 €
-------------------------	-----------------

Tableau 5 : Budget global

B. Analyse et perspectives

1. Difficultés rencontrées et solutions

Pour nous le bilan humain est dans l'ensemble très positif. En effet, grâce au travail de groupe nous avons retiré un certain nombre d'enseignements. Par exemple, comme notre travail consistait en la programmation d'une applet en langage java, certains d'entre nous qui en savaient plus dans ce domaine ont pu aider les autres et nous avons mis en place un bon système de transfert de compétences. Cela s'est fait au travers de réunions, d'échanges oraux et écrits.

Cependant, ce qui nous a paru plus difficile au niveau du travail en équipe fut l'accord des différentes méthodes de travail de chacun. En effet, deux membres du groupe projet venant de l'université et deux autres étant issu de l'école, nous avons du faire face à des divergences d'opinion dans la manière d'aborder le travail et dans la façon de préparer les revues de projet. Néanmoins, après discussions nous parvenions à nous entendre et à nous enrichir de nos différences.

Il est à noter également que le travail commun a été compliqué par l'éclatement en groupes de TD ou TP différents. Nous avons donc eu à mettre en place des réunions hebdomadaires à des horaires convenant à tous pour nous rencontrer. Enfin, dans la partie finale de la mise au point du prototype nous communiquions essentiellement par email avec la création d'un compte mail accessible à chacun et ainsi les informations circulaient très facilement.

Afin de poursuivre cette analyse, évoquons les cours qui ont été mis en place pour nous aider dans l'activité de gestion de projet. En particulier les réunions de GRH nous ont grandement aidé à gérer le côté humain du projet et à réagir dès l'apparition des problèmes. Par exemple, nous nous sommes rapidement aperçu de la relative inefficacité de réunions non préparées et avons ainsi mis en place un système de fiches qui récapitulaient notre travail passé et nous permettait de nous concentrer sur le travail à venir.

Nous avons également mis en pratique certains éléments du cours de gestion financière pour la partie budget. Et enfin, sur un plan plus technique, nous avons utilisé nos connaissances acquises en cours de programmation pour la construction des diagrammes de classe grâce au formalisme UML.

Sur un plan matériel, nous avons toujours eu à notre disposition le matériel souhaité et n'avons donc pas connu de problèmes majeurs.

Pour conclure cette analyse du projet, nous avons pu constater à quel point il est important de bien définir les objectifs dès le départ pour ensuite effectuer un travail valable et ne pas se perdre dans des activités inutiles. Pour définir ce qui était attendu nous avons eu la chance d'avoir un cahier des charges très bien décrit par notre client lui-même. Nous avons eu cependant tenu à effectuer une reformulation de ce cahier des charges pour nous assurer que nous avons bien compris les exigences du client et la problématique. Ce travail conséquent nous a semblé utile et nous avons pu vérifier par la suite qu'il s'était avéré essentiel.

2. Perspectives d'avenir

Nous avons dès le départ comme consigne de travailler sur une applet dont le code pourrait être facilement complété par d'autres. Ainsi M. Köppen a-t-il insisté pour que le code source reste ouvert et permette ainsi à une autre équipe de pouvoir reprendre le projet et compléter l'applet avec des nouvelles lois (géométrie, densités, vitesses). Nous avons donc mis en place une documentation tout au long de la partie programmation et dans l'avenir il sera aisé d'ajouter des lois, des paramètres supplémentaires et/ou des options à l'applet.

V. Description détaillée de l'applet

A. Description générale

1. Fonctionnement de l'applet

LA raison d'être de l'applet a été précédemment définie dans le chapitre « Cahier des charges » et dans l'introduction. Voyons maintenant comment se sont traduites ces exigences sur le produit final. Nous dégagerons ici quatre fonctions principales dans l'applet :

- ✓ Une interface graphique permettant la configuration de l'applet,
- ✓ Un générateur de points aléatoires,
- ✓ Un module de modélisation d'un objet stellaire, avec sa cinématique, soit un cube de données,
- ✓ Un module de représentation, simulant différentes observations possibles du cube de données.

Chacune de ces fonctions fera l'objet d'une description détaillée dans le paragraphe suivant. Pour le moment, nous allons décrire la façon dont elles s'articulent au sein du programme.

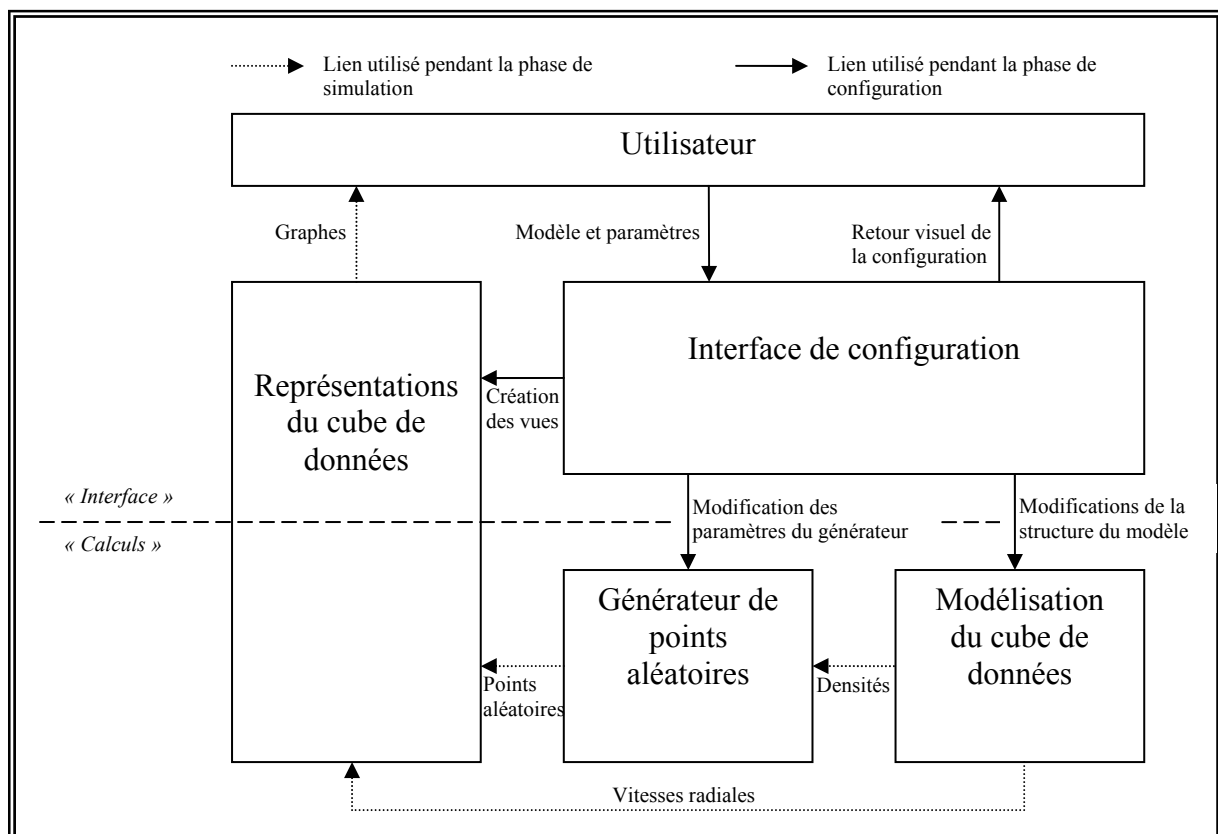


Figure 1 : Présentation synthétique de l'organisation de l'applet

Afin d'interpréter correctement ce schéma (Figure 1), rappelons que le principe de production des graphes utilise la méthode dite de Monte-Carlo : en résumé, cela consiste à générer successivement des points aléatoires (avec une distribution imposée par le modèle) et

à ajouter leur contribution au graphe ; on peut prouver que lorsque le nombre de points tend vers l'infini (notion dont l'application effective dépend notamment de la précision voulue), le graphe obtenu est le même que celui obtenu par la méthode exacte, c'est-à-dire celle qui consiste à calculer directement, en chaque point du graphe, la valeur qui doit être prise.

Après cette mise au point, indiquons qu'un cycle de simulation a lieu comme suit :

- ✓ Le module de représentation requiert n points supplémentaires pour ses graphes,
- ✓ Le générateur sort ces n points, en faisant appel aux densités de points imposées par le modèle,
- ✓ Le module représentation met à jour ses graphes avec ces points, en utilisant une information supplémentaire, la vitesse radiale fournie par le modèle.

Ainsi, une utilisation normale de l'applet consistera à :

- ✓ Créer un modèle,
- ✓ Configurer le générateur,
- ✓ Sélectionner la ou les représentations voulues,
- ✓ Lancer la simulation.

2. Description générale de l'interface

L'interface de l'applet se compose de trois composantes (cf. Figure 2) :

- ✓ Une barre de boutons en haut, fournissant des fonctions touchant à l'applet en général,
- ✓ Une liste d'objets paramétrables à gauche,
- ✓ Un panneau variable au centre : son contenu varie en fonction de l'objet sélectionné dans la liste. Selon le cas, il présente un panneau de configuration, ou un graphe (représentation du cube de données).

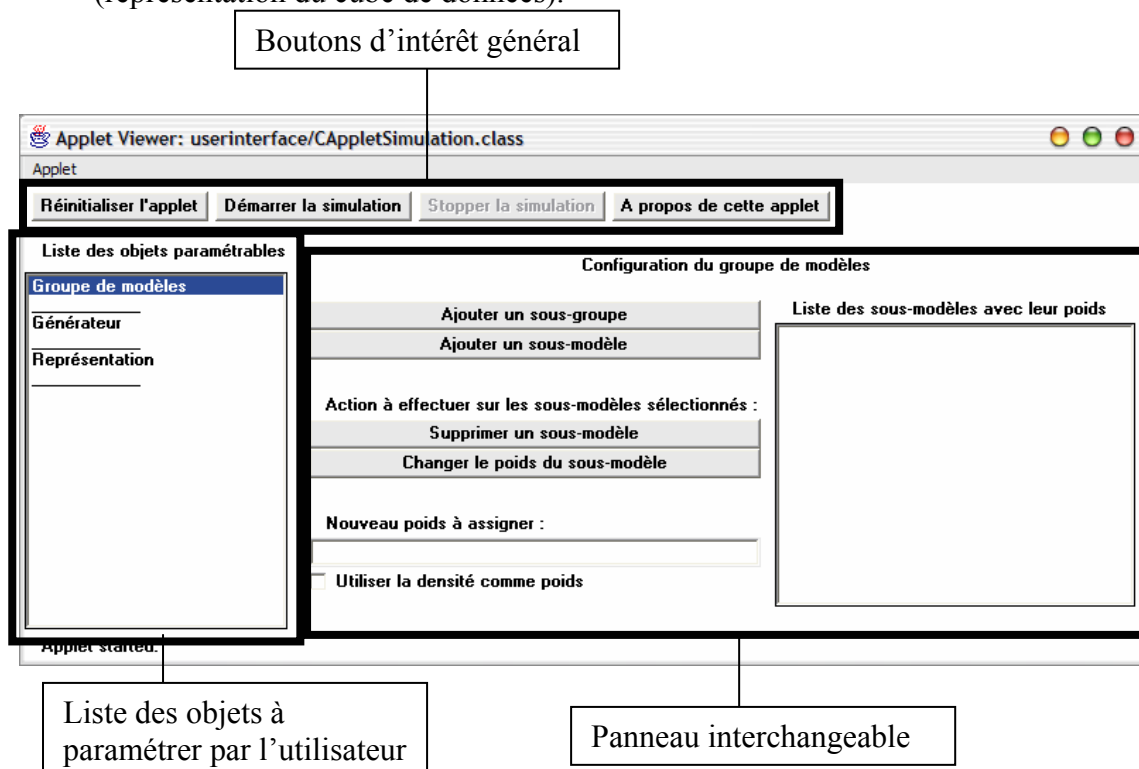


Figure 2 : Elements principaux de l'interface

Lorsque de nouveaux objets sont construits pour convenir aux besoins de l'utilisateur, des éléments sont ajoutés à la liste de gauche afin de permettre leur configuration. Selon leur nature, sous-modèle, loi, graphe..., ils apparaîtront soit sous « Groupe de modèles », soit sous « représentation ».

B. La création d'un modèle

1. Structure d'un modèle

Dans cette applet, le modèle cinématique d'un objet stellaire est un ensemble d'objets organisés hiérarchiquement, c'est-à-dire sur un arbre hiérarchique. Ces objets sont répertoriés dans le tableau suivant :

Modèle autonome : c'est un nœud de l'arbre.	Loi (composant d'un modèle complet) : c'est une feuille de l'arbre.
Modèle complet : il doit regrouper au moins une loi de chaque type, mais aucun modèle.	Loi de vitesse : elle fournit le champs de vitesse d'un modèle complet.
Groupe de modèles : il doit regrouper un ou plusieurs modèles.	Loi de géométrie : elle définit l'espace qu'englobe un modèle complet.
	Loi de densité : elle définit la densité en tout point à l'intérieur du modèle complet.

Tableau 6 : Lois et modèles

Voici graphiquement comment s'articulent ces objets :

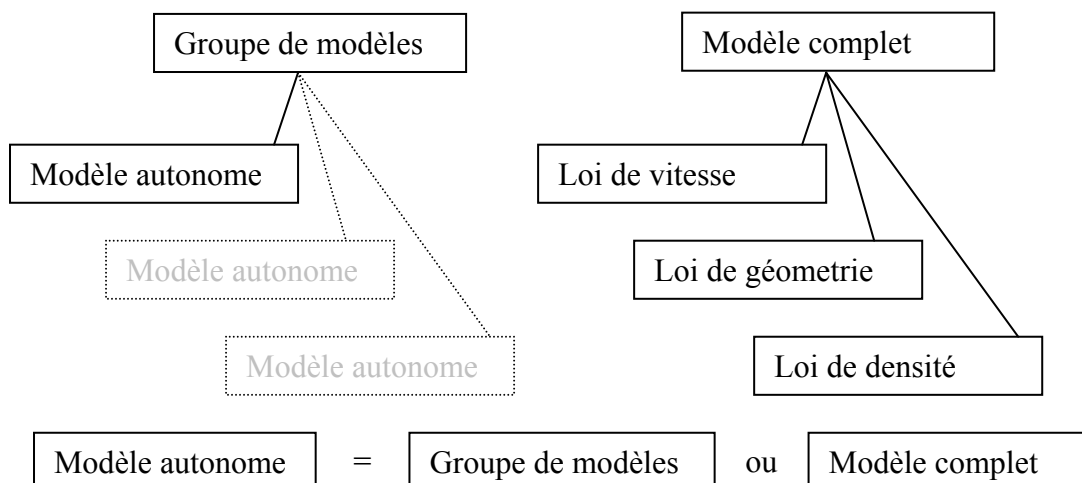


Figure 3 : Hiérarchie des objets format un modèle

En plus de ce qui a été défini dans le tableau ci-dessus, s'appliquent les conditions suivantes :

- ✓ La racine de l'arbre est obligatoirement un objet « groupe de modèles ».
- ✓ Dans un modèle complet, quand plusieurs lois de vitesses sont concurrentes en un point, les vecteurs vitesses sont ajoutés. De même pour les lois de densité.
- ✓ Dans un groupe de modèles, quand plusieurs lois de vitesses sont concurrentes, elles sont ajoutées, coefficientées par la densité correspondante ou bien un poids prédéfini, au point considéré.
- ✓ Quand plusieurs géométries sont utilisées pour un même objet « modèle complet », l'espace résultant est la réunion des espaces définis par chacune des lois, à l'exception notable qui suit : si l'utilisateur le précise, une géométrie peut définir un espace à **exclure** au lieu d'un espace à **inclure**. Dans ce cas précis, en tout point de l'espace à exclure, les densités et vitesses renvoyées par l'objet « modèle complet » qui le contient seront systématiquement nulles.
- ✓ En l'absence d'une loi de vitesse, le champ de vitesse est nul ; sans loi de densité, la densité est nulle, et sans géométrie l'espace englobé par le modèle est vide.

2. Les repères

Chaque type d'objet présenté ci-dessus dispose d'un repère local, défini par rapport au repère de son objet parent, c'est-à-dire le modèle ou groupe de modèle (le nœud) immédiatement supérieur dans « l'arbre » que forme le modèle dans son ensemble.

Pour définir un repère, on a besoin de trois triplets de réels :

- ✓ Les coordonnées (x, y, z) de l'origine locale O' exprimées dans le repère parent,
- ✓ Trois angles de rotation autour des axes du repère parent,
- ✓ Les normes des trois vecteurs locaux exprimées par rapport aux vecteurs directeurs originaux correspondants (voir figure)

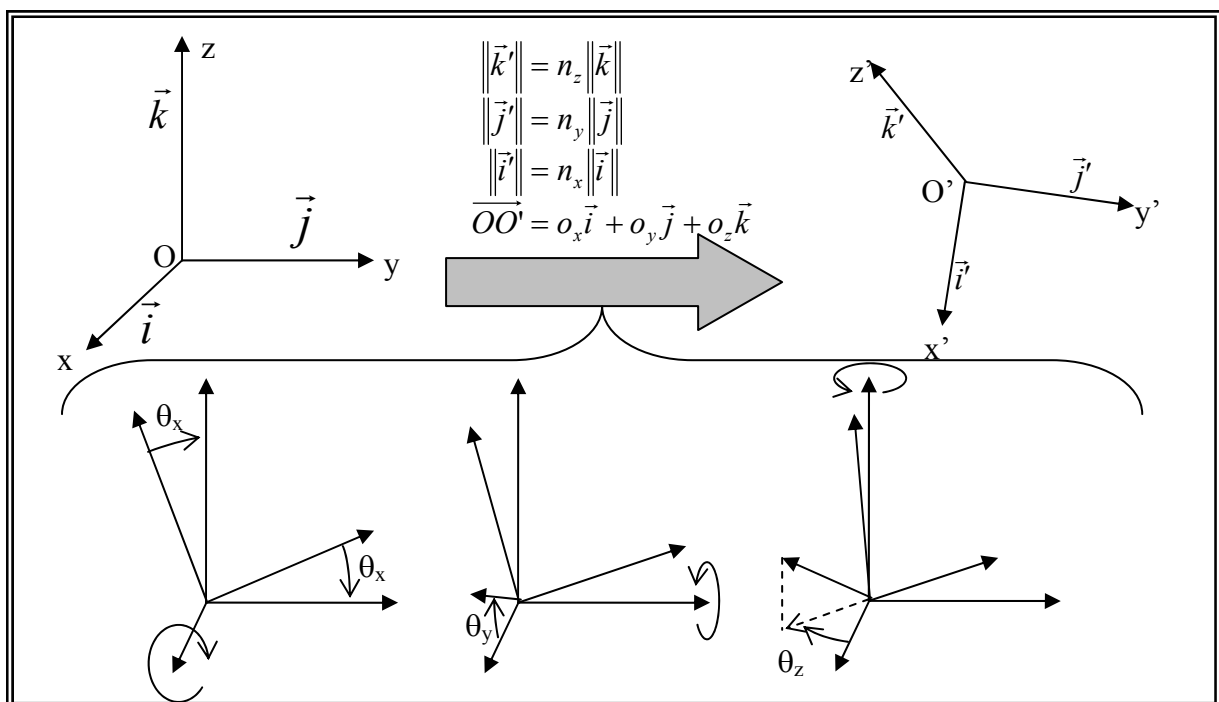


Figure 4 : Schéma explicitant le passage du repère parent au repère local

3. Le panneau « groupe de modèles »

Ce panneau (voir Figure 5) permet de créer un sous-groupe ou un sous-modèle (modèle complet) en appuyant sur le bouton approprié. On peut également sélectionner dans la liste un élément existant afin de le supprimer, ou de changer son poids associé.

La case à cocher « Utiliser la densité comme poids » détermine la façon dont sont combinées les vitesses : on peut soit utiliser les poids définis manuellement pour chaque modèle, soit utiliser la répartition des densités en chaque point, pour pondérer les différentes vitesses correspondant aux différents sous-modèles.

Configuration du groupe de modèles

Ajouter un sous-groupe	Liste des sous-modèles avec leur poids	
Ajouter un sous-modèle		
Action à effectuer sur les sous-modèles sélectionnés :		
Supprimer un sous-modèle	<div style="border: 1px solid black; height: 150px;"></div>	
Changer le poids du sous-modèle		
Nouveau poids à assigner :		
<input type="text"/>		
<input type="checkbox"/> Utiliser la densité comme poids		

Figure 5 : Panneau de configuration d'un groupe de modèles

4. Le panneau « modèle »

Ce panneau (voir Figure 6) permet de choisir les différentes lois de vitesse, géométrie, et densité associée à un modèle complet. Pour ce faire, on sélectionne la loi dans la liste de gauche, et on appuie sur « ajouter ».

Pour supprimer une loi existante dans le modèle, on la sélectionne dans la liste de droite, et on appuie sur « supprimer ».

Les cases d'édition en bas du panneau permettent de définir le repère relatif de ce modèle par rapport au repère du groupe de modèles « père ». On y définit ainsi les trois vecteurs directeurs des trois axes du repère, ainsi que la position de l'origine.

Configuration du modèle

Ajouter une loi	Supprimer un sous-modèle	
Sélectionner la loi à ajouter : Géométrie anneau Géométrie coquille Géométrie disque Géométrie jet Géométrie sphérique Géométrie paraboloides Géométrie paraboloides double Densité Constante Densité Exponentielle Densité Gaussienne Densité Constante Axiale Densité Exponentielle Axiale Densité Gaussienne Axiale Vit aléatoire isotrope (bruit) Vit en expansion radiale constante	Action à effectuer sur les sous-modèles sélectionnés :	
Repère local par rapport au repère parent :		
Orientation : X: <input type="text" value="0"/> Y: <input type="text" value="0"/> Z: <input type="text" value="0"/>	Norme : X: <input type="text" value="1"/> Y: <input type="text" value="1"/> Z: <input type="text" value="1"/>	Origine : X: <input type="text" value="0"/> Y: <input type="text" value="0"/> Z: <input type="text" value="0"/>

Figure 6 : Panneau de configuration d'un modèle

5. Les panneaux « loi »

Ces panneaux (voir l'exemple Figure 7) permettent de définir des paramètres numériques spécifiques à chacune des lois utilisées. Elles proposent également de modifier le repère local (cf. V.B.2 Les repères).

Selon les cas, on peut également y voir un aperçu de la loi avec les paramètres en cours, ainsi qu'un dessin expliquant l'effet de ces paramètres.

Dans le cas d'une géométrie, il y a également une case à cocher permettant d'indiquer si l'espace défini est à inclure ou exclure (cf. V.B.1 Structure d'un modèle).

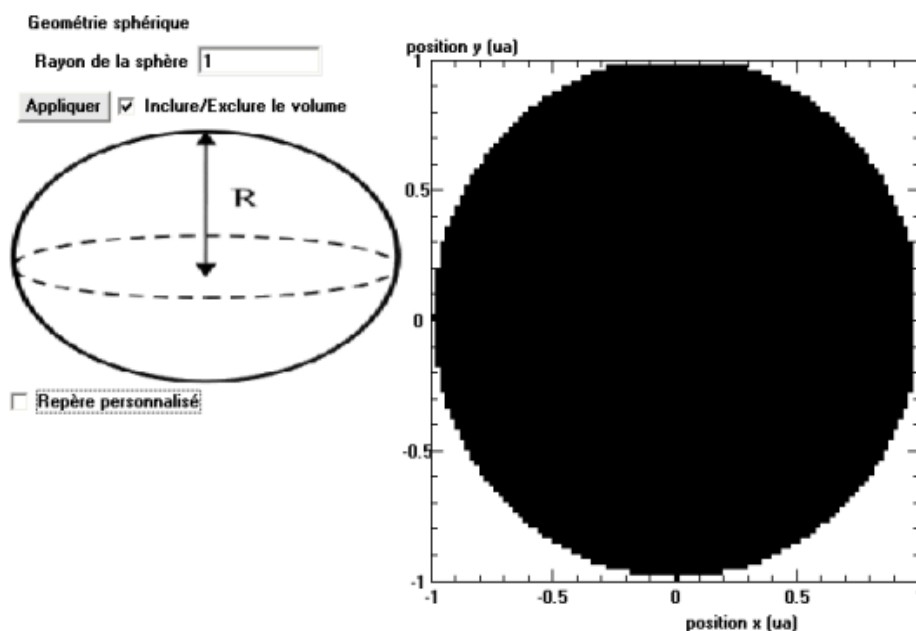


Figure 7 : Panneau de configuration de la géométrie sphérique (exemple).

C. Le générateur

Il n'y a qu'un panneau pour configurer l'ensemble des fonctionnalités du générateur. On peut y définir le cube à l'intérieur duquel les points aléatoires seront générés, ce sont les bornes min et max pour x, y, et z. On peut définir également le nombre de points d'échantillonnage en x, y, et z dans ce cube (Figure 8). (plus ces pas sont élevés, plus le temps de génération par point augmente.)

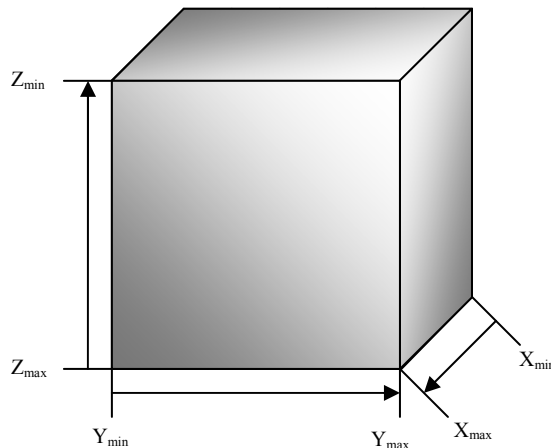


Figure 8 : Ce cube illustre ce qu'on entend par le nombre de points d'échantillonnage en x, y, et z. La densité est calculée à chaque intersection de plans.

Enfin, le pas utilisé pour les calculs d'intégrales nécessaires à l'initialisation du générateur peuvent également être définis (plus ces pas sont élevés, plus l'initialisation du générateur est longue...)

Les effets de ces réglages sur le temps de génération sont détaillés de façon plus approfondie dans le rapport technique. Les réglages initiaux permettent des calculs rapides avec des machines de puissance moyenne.

D. Les représentations

1. Le panneau de choix des représentations

Ce panneau (Figure 9) permet de sélectionner dans la liste de gauche les types de représentation à ajouter. À droite s'affichent les représentations en cours, qu'on peut supprimer en utilisant le bouton ad hoc.

Ce panneau permet également de choisir le nombre de points calculés par cycle de simulation. Ceci permet à l'utilisateur de choisir s'il préfère un rafraîchissement fréquent des graphes au détriment du temps de calcul, ou le contraire.

L'utilisateur peut créer autant de vues qu'il le souhaite pour une même simulation : bien qu'il ne puisse en voir qu'une à la fois, il peut à tout moment pendant la simulation passer d'une vue à l'autre, en la sélectionnant dans la liste des objets paramétrables, dans la partie gauche de l'applet.

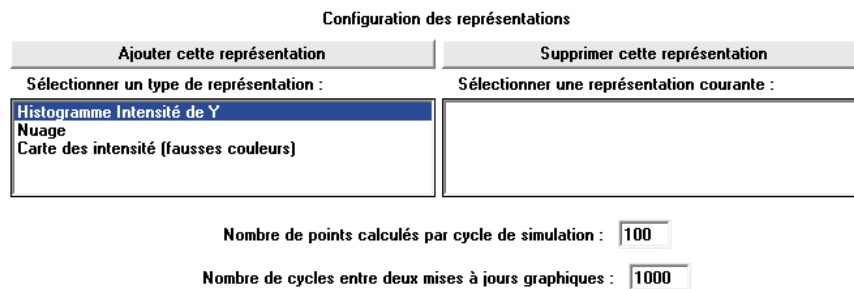


Figure 9 : Panneau de configuration des représentations.

2. Les panneaux des graphes

Ces panneaux (voir l'exemple Figure 10) fournissent en plus des graphes, des réglages spécifiques permettant à l'utilisateur de paramétrer plus finement ses vues. Il y a actuellement trois types de panneaux de graphes :

- ✓ les histogrammes, qui permettent de visualiser vitesses moyennes, écart-types des vitesses, intensité... cela à l'intérieur d'un parallélépipède défini par l'utilisateur, en fonction de la vitesse radiale ou de la position (un des 3 axes du cube de données).
- ✓ les cartes en fausses couleurs, permettent d'obtenir les mêmes renseignements, mais en fonction de la position dans un plan et non plus seulement le long d'un axe.
- ✓ le nuage de points, destiné à faciliter la visualisation du modèle : il fournit une vue en perspective de la distribution des points générés lors de la simulation, contrairement aux 2 types précédents qui montrent des projections.

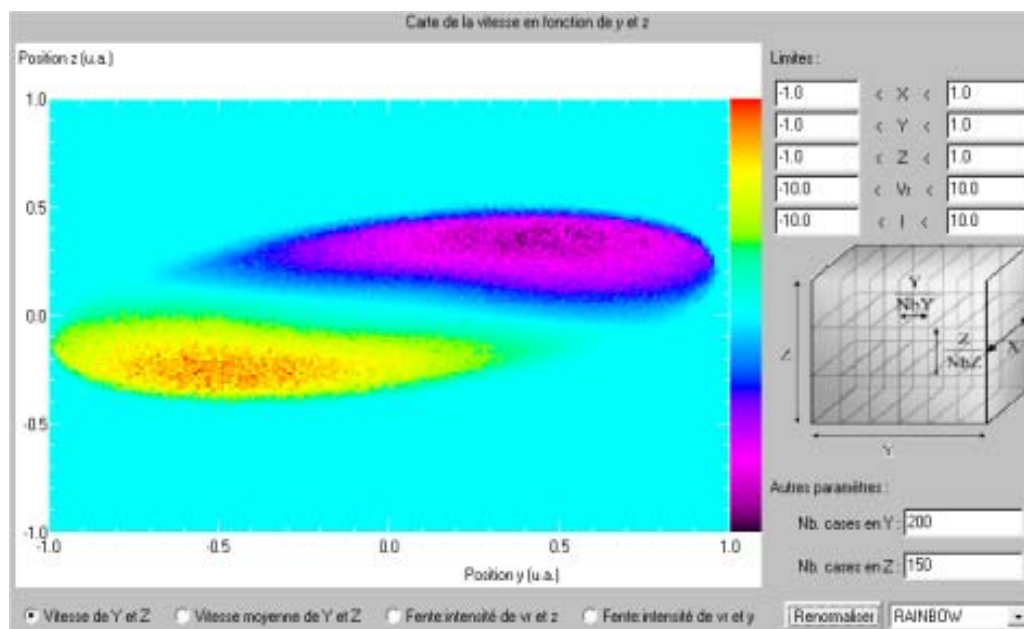


Figure 10 : Panneau « carte en fausses couleurs » (exemple)

E. Exemple concret

On observe un objet astronomique à partir du rayonnement qu'il émet. A l'aide de l'effet doppler on peut savoir si un objet se rapproche ou s'éloigne. On dresse ainsi une carte en fausse couleur des intensités dans un plan d'observation (on observe l'objet à l'infini).

Cette carte porte le nom dans notre applet de carte des intensités : vitesse de Y et Z.

On observe l'objet X suivant :

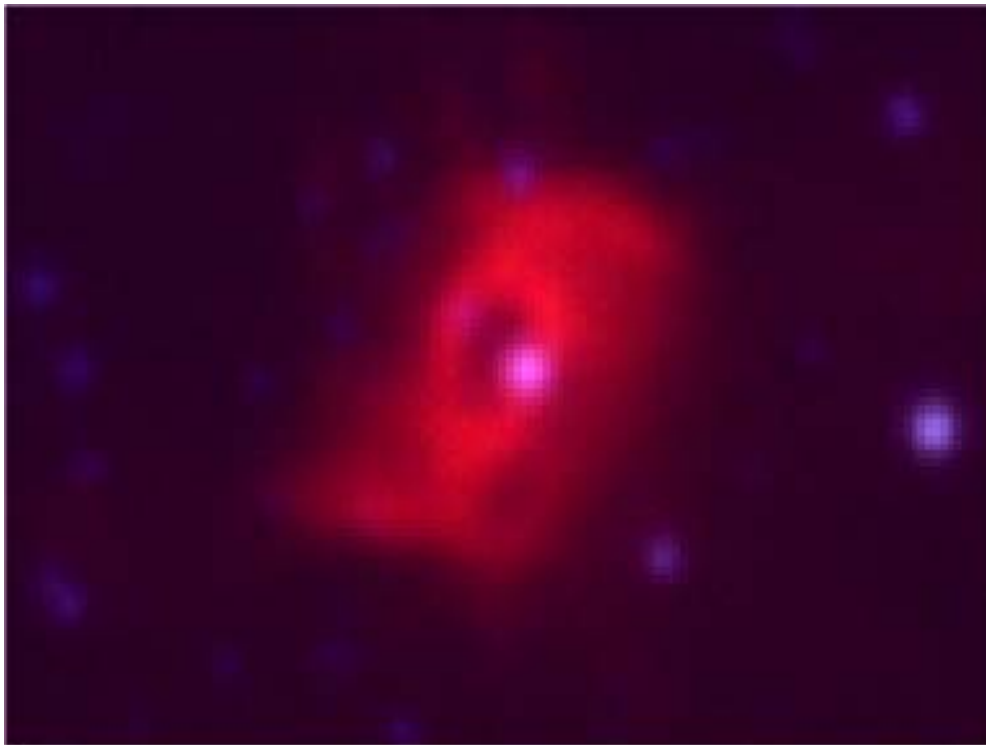
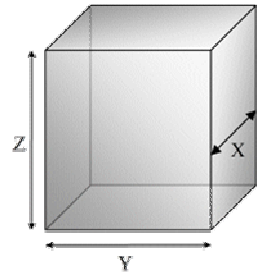


Figure 11 : Observation réelle d'un objet astronomique

On remarque une ellipsoïde évidée au centre (anneau étiré) ainsi que deux branches aux extrémités.

Le problème qui se pose est de comprendre la nature de l'objet observé à partir de cette image. En particulier on cherche un modèle permettant de le décrire. Ce modèle comporte une géométrie, une loi de densité, une loi de vitesse : par exemple une planète (sphère) de densité constante et en expansion radiale.

L'applet nous permet de trouver se modèle en comparant la carte d'intensité théorique de différents modèles possibles avec la carte observée : on en déduit le modèle le plus approprié. On effectue une recherche par tâtonnements.

1) On observe peut être une planète ?

On crée le modèle suivant. On observe la carte des intensités à l'aide de l'applet :

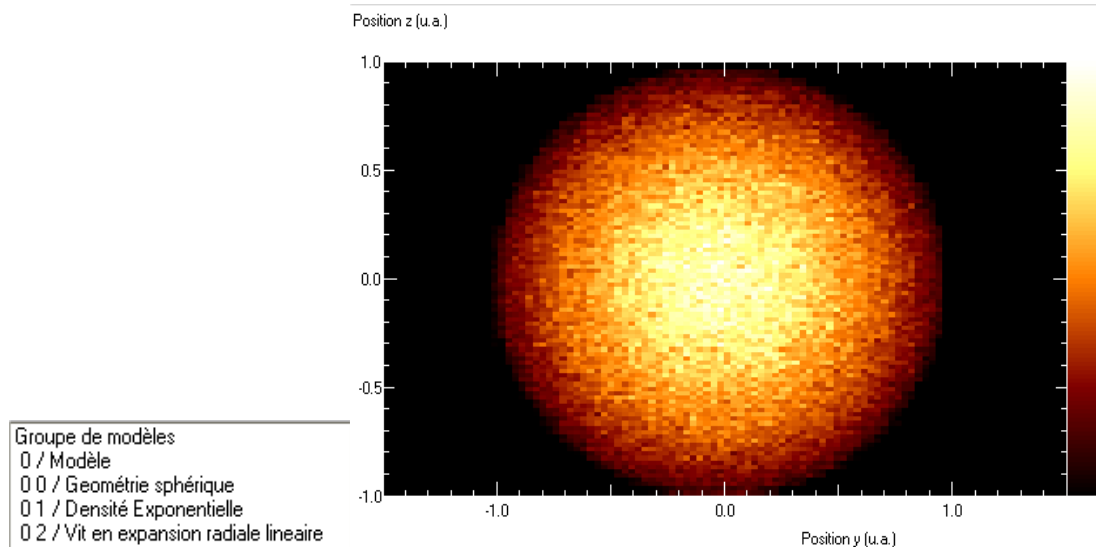


Figure 12 : Carte d'intensité d'une sphère de densité exponentielle

Cette carte correspond bien à une sphère de densité exponentielle : Il y a beaucoup de particules au centre ce qui correspond à une couleur claire et de moins en moins (exponentielle décroissante) au fur et à mesure que l'on s'éloigne du centre.

On voit d'emblée que ce modèle ne correspond pas pour l'objet X : on a une ellipsoïde mais pas d'anneau ni de branches. L'objet X est plus compliqué.

2) hypothèse n°2 coquille en expansion

On crée le modèle :

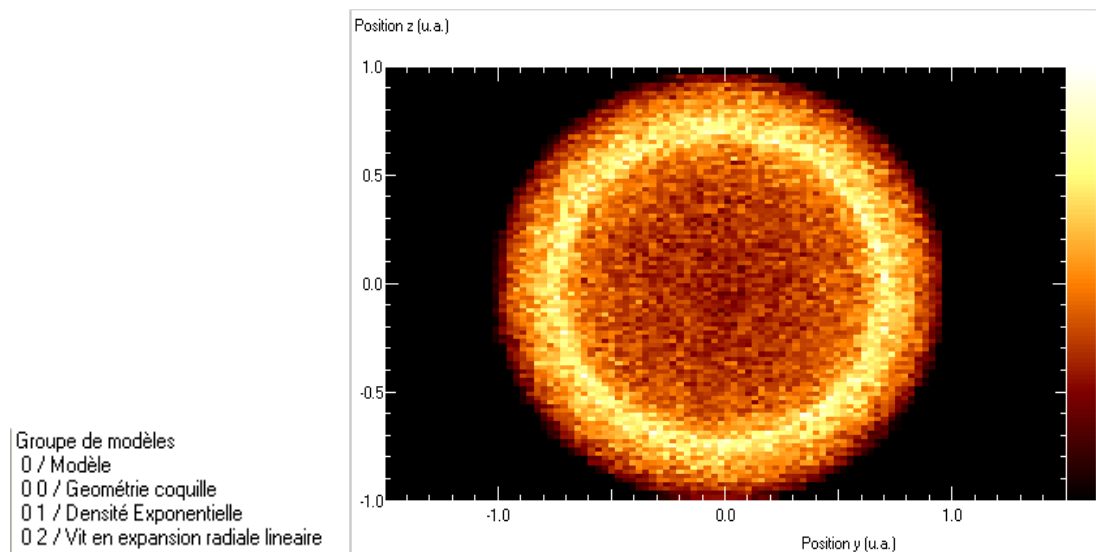


Figure 13 : Carte d'intensité d'une coquille en expansion

On se rapproche un peu de l'objet X on a l'ellipsoïde et un anneau. Mais on n'a toujours pas les deux branches à l'extérieur.

3) Hypothèse 3 : deux coquilles en expansions

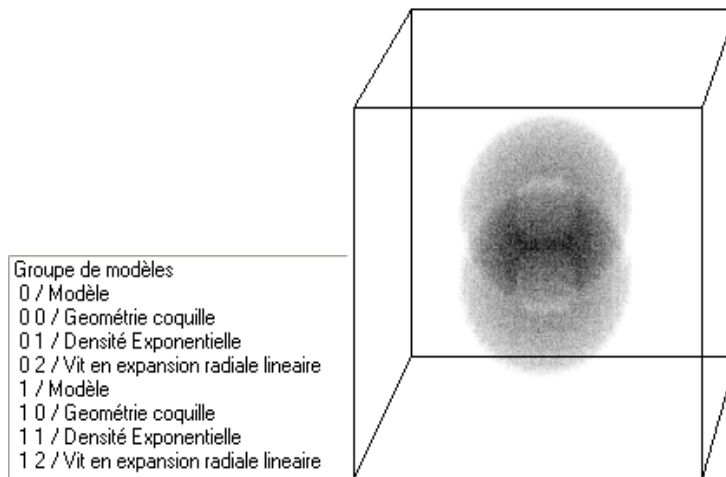


Figure 14 : Vue en perspective d'un modèle à 2 coquilles

Le cube permet de visualiser en projection le nuage de point ainsi créé : ici on vérifie que les deux coquilles se chevauchent au centre.

On obtient la carte des intensités :

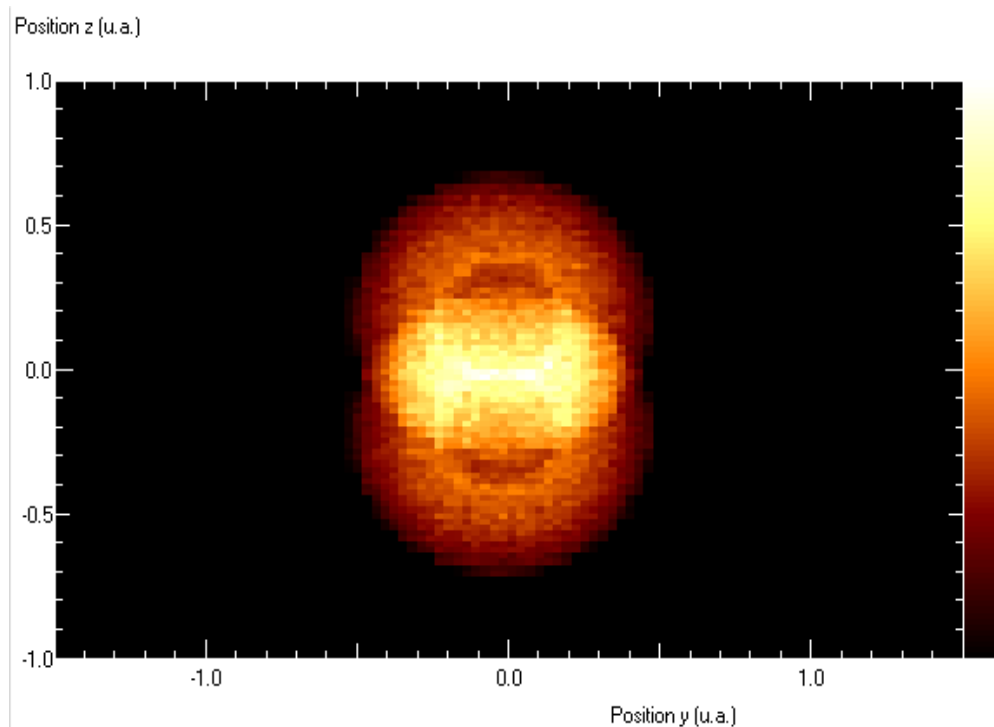


Figure 15 : Carte d'intensité du modèle à 2 coquilles

On s'éloigne de l'objet X. On va remplacer la coquille par une parabolöide.

4) hypothèse 4 deux parabolöides en expansions :

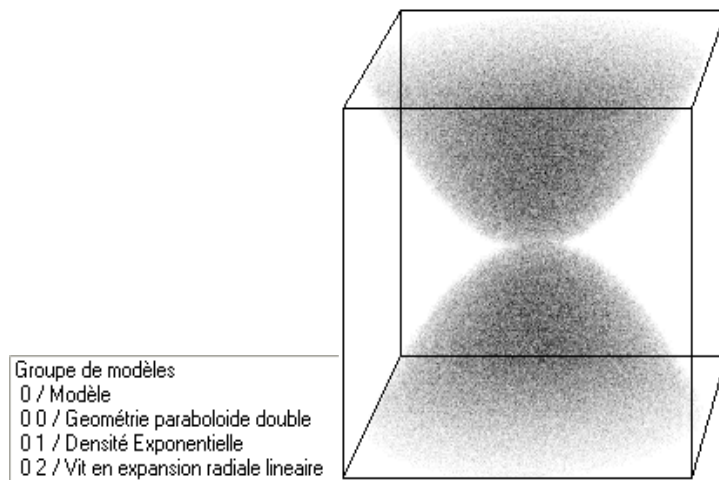


Figure 16 : Vue en perspective du modèle paraboloïde double

Cette géométrie correspond à un diabolo.

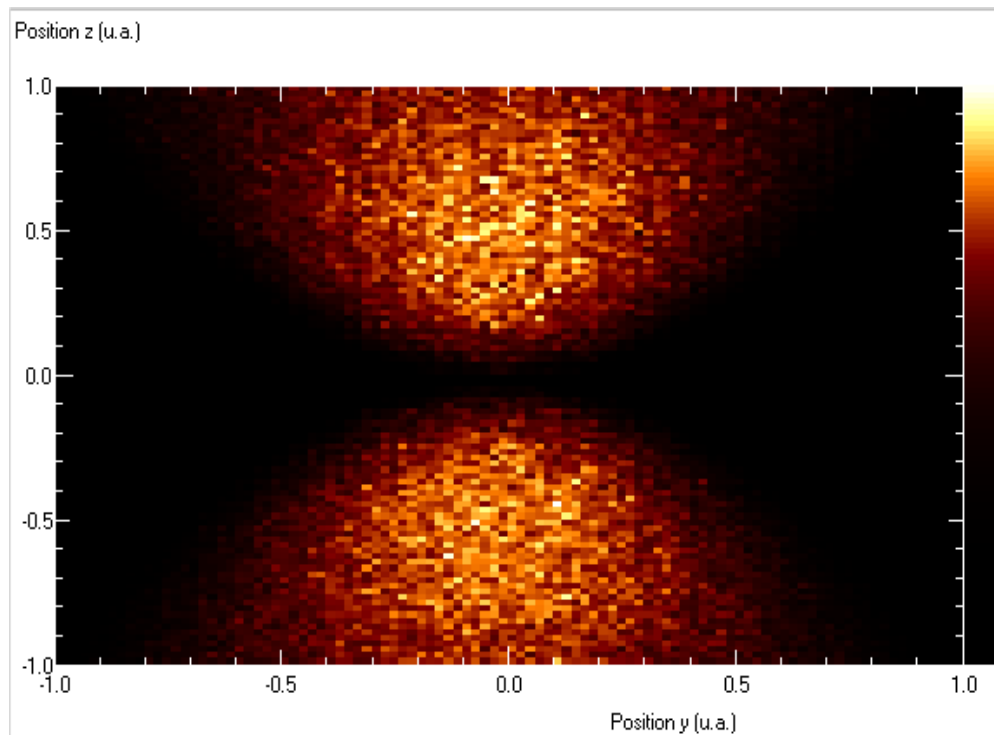


Figure 17 : Carte d'intensité du modèle paraboloïde double

On a observé à l'aide de l'anneau que l'objet était plutôt creux au centre donc on va modifier le modèle pour avoir des points uniquement en bordure.

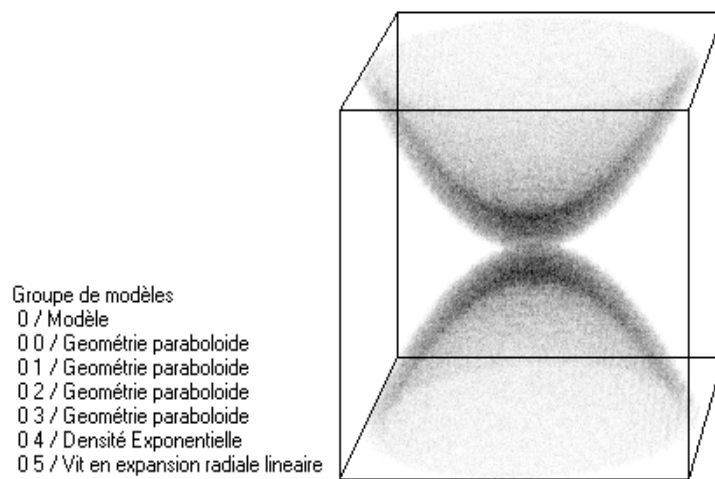


Figure 18 : Vue en perspective d'un modèle paraboloïde évidé

On a utilisé 4 paraboloïdes 2 orientés vers le haut et deux orientés vers le bas. Les 4 ont la même densité et deux d'entre eux sont exclus (leur densité est comptée négativement) ce qui permet d'éviter le diabolos : il n'y a que des points sur l'enveloppe externe.

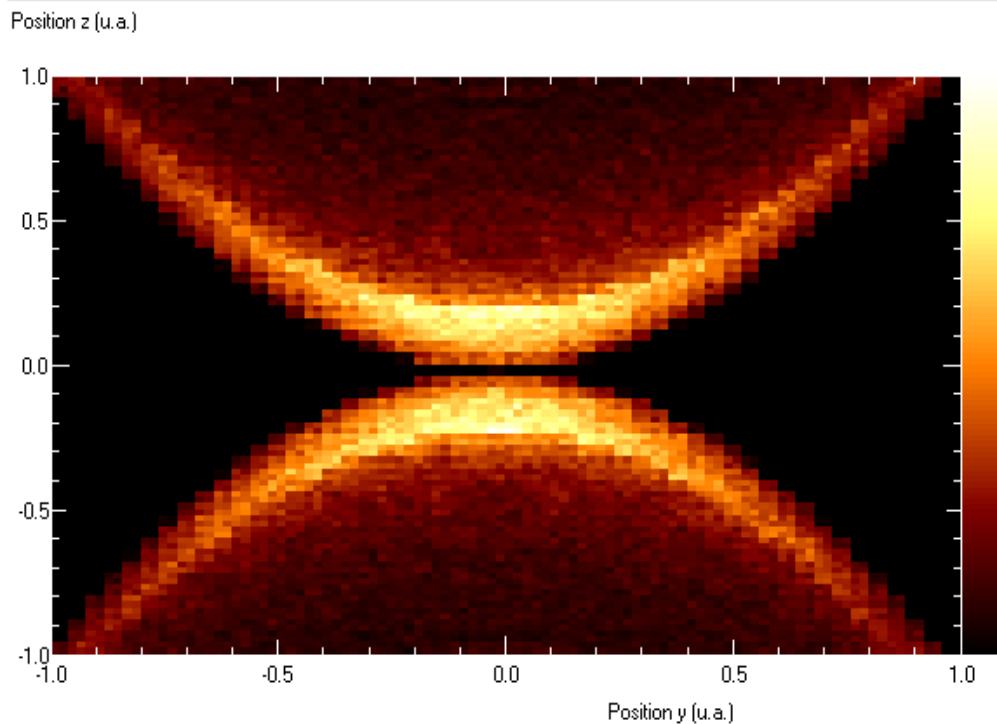


Figure 19 : Carte d'intensité du modèle paraboloïde évidé

On approche du résultat. On incline le diabolos : rotation de 0.8 rads par rapport à l'axe y.

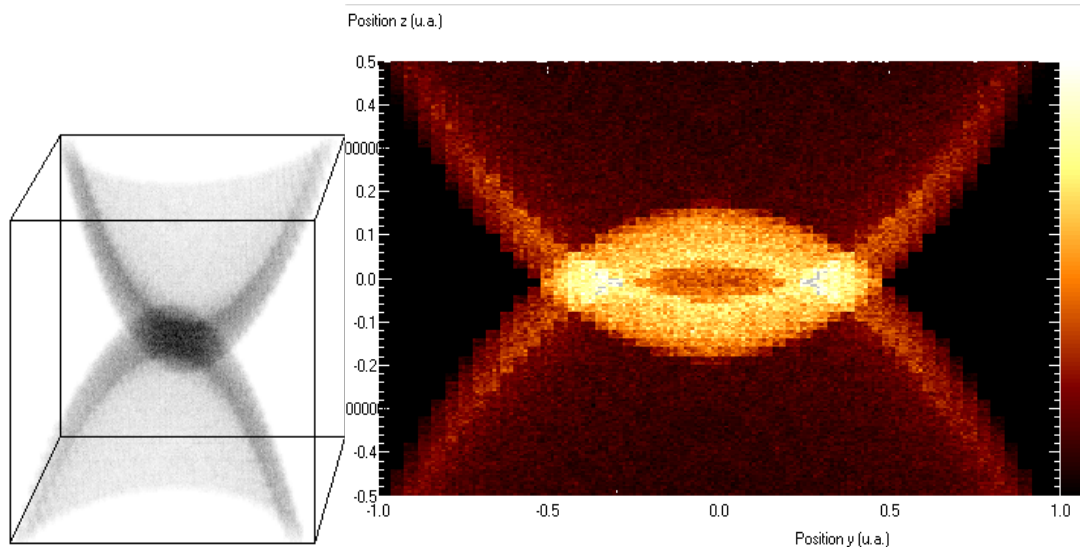


Figure 20 : Vue en perspective et carte d'intensité du paraboloïde évidé, avec un angle d'observation différent.

On a maintenant une image intéressante : en effet on observe les 2 branches externes et l'ellipsoïde évidée au centre plus lumineuse. L'objet X est bien un diabolos de densité exponentielle.

Remarque : nous avons résumé la démarche puisque l'on a pris directement une densité exponentielle et une vitesse en expansion radiale linéaire. Il faudrait maintenant effectuer la même démarche avec les cartes de spectroscopie par une fente pour vérifier que la loi de vitesse choisie est correcte, et éventuellement tester les effets d'un changement.

VI. Site Web

Il a été décidé vers la fin décembre de mettre en place un site Web pour présenter l'Applet et le projet dans sa globalité. En effet, les Applet Java sont des applications faites pour être diffusées sur le Web au travers de pages au format html et donc la création d'un site nous est apparue logique.

Ainsi avec un site Web nous espérons pouvoir toucher un plus grand nombre d'utilisateurs (étudiants, astronomes, amateurs d'astrophysique) car il sera ainsi accessible depuis n'importe quel ordinateur relié à Internet et ce dans la monde entier.

Cette applet était au départ destinée à être incluse dans les pages personnelles de M. Köppen, nous avons ensuite cherché à étendre sa diffusion par l'intermédiaire d'un site hébergé par Lycos.

Le site Web est donc visible aux deux adresses suivantes :

<http://membres.lycos.fr/enspsastroprojet>

Site consacré au projet.

<http://astro.u-strasbg.fr/~koppen/cisobom/Simulation.html>

Sur les pages Web de M. Köppen.

Il se compose des rubriques suivantes :

- ✓ Une page d'accueil à laquelle on accède en premier.

- ✓ Un sommaire avec la présentation générale du site. Cette page comporte les liens vers les différentes pages du site par l'intermédiaire d'une barre de navigation située en bas de chaque page.
- ✓ Une page entièrement consacrée à l'Applet.
- ✓ Une page d'aide en ligne sur les principales fonctions de l'Applet.
- ✓ Une page présentant l'équipe et les rôles de chacun.
- ✓ Une page avec des liens utiles pour en savoir plus sur les différents aspects du projet.

VII. Conclusion

Ce travail qui s'est étalé sur 7 mois nous amène à différentes conclusions :

Tout d'abord concernant le projet en lui-même ; nous avons conçu une applet pédagogique, interactive et conviviale pouvant être utilisée par des étudiants, dans le but de se familiariser avec le monde de l'astrophysique, ou par des chercheurs, dans le but de faciliter leurs travaux de recherche. De plus, cette applet est accessible à tous car nous l'avons implantée au sein d'un site. Le cahier des charges a donc bien été validé et notre client, le Pr. KÖPPEN nous a dit être satisfait du résultat. Grâce à son implémentation, ce logiciel peut aussi jouir d'améliorations ultérieures afin de construire des modèles plus complets ou de mettre en place d'autres représentations (cf. dossier industriel pour plus de renseignements).

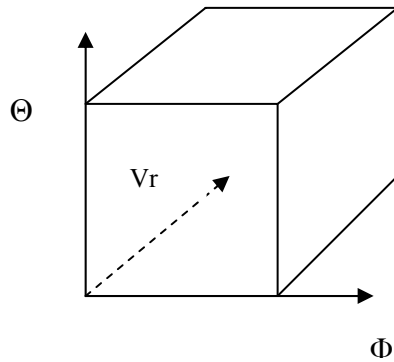
En second lieu, l'équipe et chacun de ses membres ont pu acquérir des compétences. Au niveau du savoir et du savoir-faire, chacun de nous a appris à utiliser le langage Java qui jusqu'alors nous était inconnu. Au niveau de la méthodologie, nous avons appris à analyser notre travail, en particulier en testant à des moments stratégiques nos morceaux de code mais surtout en étant toujours à l'écoute de notre client. Nous avons aussi appris qu'un projet ne pouvait se développer convenablement que si il y avait un minimum de communication entre les membres de l'équipe, afin de pouvoir juger de l'évolution du projet, avec le client, afin qu'il nous renseigne sur ses exigences, et avec d'autres personnes qui seraient susceptibles de nous aider (par exemple nous avons fait appel à Mr.LEROY).

Cette première approche d'une démarche projet a été une expérience concluante pour chacun de nous au niveau humain et au niveau de la réalisation du projet : c'est la première fois dans notre cursus que notre travail débouche sur un produit « utile », apprécié du client.

VIII. Annexes

Annexe A : Glossaire

Cube de données (d'observations astronomiques) :



Les positions et la vitesse radiale d'un point nous donne le cube de données.

Effet Doppler: variation de hauteur du son perçu lorsque la source sonore se déplace par rapport à l'observateur (que l'on peut transposer à la lumière).

Applet: programme écrit en Java implantable dans une page de la Toile.

Simulation Monte Carlo: à partir d'une génération de nombres aléatoires ou pseudo aléatoires, la méthode Monte Carlo consiste à représenter un phénomène physique par une succession d'événements tirés au sort à partir de lois de probabilités correspondantes.

Annexe B : Bibliographie

Pour ce projet, la documentation se présente sous plusieurs formes. Une partie des documents utilisés sont des codes Java fournis par le client. Ils constituent une bibliothèque graphique permettant de réaliser par exemple des cartes en fausses couleurs. Ces différents programmes Java se retrouvent de façon plus détaillée dans le rapport industriel.

Il y a ensuite plusieurs références d'ouvrages qui nous ont servi et que nous présentons ci-dessous.

Cahier des charges complet fourni par le client donnant des explications précises sur le travail attendu ainsi que des exemples de représentations de données astronomiques.

Mémoire de DEA Analyse et Traitement de Données sur les Milieux Astrophysiques : « Cartographie de nuages moléculaires géants dans la galaxie spirale M81. » par Eric Dupuy (2000)

Branché Java par Hoff et al. Aux Editions International Thomson Publishing Company, 1996.

Java in a Nutshell par Flanagan aux Edition O'Reilly.

Programmer en Java aux Editions Anuff, 1997.

Java par Neuwmann et al. Aux Editions Le MacMillan, 1996.

Numerical Recipes, par Press et al. Aux Edition Cambridge University Press, 1992.

Annexe C : Contenu du rapport industriel

Le rapport industriel se présentera sous forme informatique. Dans ce document, nous détaillerons de manière précise comment est construite l'applet, nous inclurons les morceaux de code et nous indiquerons de quelle manière le logiciel peut être complétée par la suite. Les différentes parties constituant le rapport industriel seront donc les suivantes :

- ✓ Une présentation rapide (sous forme de diagramme).
- ✓ Une description détaillée de chacun des modules : générateur, modélisation, représentation, interface dans laquelle nous inclurons les morceaux de code.
- ✓ Une description du site internet où est implantée l'applet.
- ✓ Une partie expliquant comment compléter l'applet : pour ajouter des lois, des boutons, etc.

Annexe D : Transparents de R6