

## Reformulation synthétique du cahier des charges du projet de 2<sup>ème</sup> année ENSPS n°6

### ***Situation actuelle et enjeux***

Les observations astronomiques d'objets à émission (galaxies, nuages gazeux...) ne permettent pas de déterminer de façon immédiate la forme et la cinématique en 3 dimensions de ces objets. En effet, on observe que des images projetées (2D), ainsi que la vitesse radiale (1D grâce à l'effet Doppler), c'est-à-dire un cube de données.

Dans ces conditions, il serait utile d'avoir 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.

### ***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.

En annexe sont fournies quelques exemples de modèles, de propriétés associées, et de représentations.

### ***Interlocuteurs***

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 du projet

LE LOIREC Cindy – responsable de la communication

BRESSON Sophie – responsable de la documentation et du budget

SERCKI Marc-Olivier – coordination technique

Cette équipe est encadrée au sein de l'ENSPS par le Pr. GOLTZENE Alfred. Elle est évaluée 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.

### ***Fonctionnalités du produit, exigences particulières***

- ✓ **Applet Java v.1.05** : par applet, on désigne un programme écrit en Java (JDK version 1.05 imposée), et implantable dans une page de la Toile.
- ✓ **Evolutif** : l'architecture de ce programme doit être suffisamment adaptable pour permettre d'ajouter ultérieurement de nouveaux modèles, ainsi que de nouvelles représentations des observations simulées, 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 (il faut donc mettre à jour la représentation en cours de simulation, et permettre un arrêt par l'utilisateur).

### ***Evaluation (fictive) des coûts***

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

Le matériel et logiciel nécessaires au projet sont les suivants :

- un ordinateur par personne (4 \* 1500€ soit 6000€) : déjà disponible.
- le Java Software Development Kit v.1.05 dit 'distribuable', dont la licence autorise gratuitement l'utilisation normale.
- Documentation diverse, sur divers supports, dont le coût non encore fixé, ne saurait dépasser 500€, s'il n'était disponible à l'Université ou à l'Ecole.

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

Le budget matériel s'élèverait donc à 9000€.

### ***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 spécifiques 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 + : implémentation de propriétés supplémentaires au modèle, de représentations supplémentaires. Tests et corrections.

D'après ces estimations, il serait possible de fournir un prototype fonctionnel dans le courant de mars, et une version définitive en mai 2003.

## Annexe

### **Modèles géométriques possibles (liste non exhaustive) :**

- sphère
- ellipsoïde
- jets
- disques
- modèles asymétriques

### **Propriétés de ces modèles**

- rotation (à implémenter)
  - o solide
  - o constante
  - o arbitraire...
- expansion (à implémenter)
  - o radiale linéaire
  - o radiale arbitraire
  - o sans symétrie sphérique...
- distribution de densité (à implémenter)
  - o constante
  - o exponentielle
  - o gaussien
- mouvements turbulents (éventuellement, selon les performances de calcul)
- effets d'une résolution spatiale et spectrale finies (éventuellement, selon les performances de calcul)

### **Représentations**

- histogrammes
- cartes en fausses couleurs
  - o de l'intensité totale (intégrée sur la vitesse radiale  $v_r$ ) en fonction de la position dans la projection ( $dz$  et  $\eta$ )
  - o intensité à  $v_r$  fixé en fonction de  $dz$  et  $\eta$
  - o moyenne de  $v_r$ , écart-type de  $v_r$  en fonction de  $dz$  et  $\eta$
  - o intensité à  $\eta$  fixé en fonction de  $dz$  et  $v_r$
  - o ...tout ce qui paraîtra pertinent...

Il s'agira d'implémenter un sous-ensemble des différents modèles, propriétés et représentations ci-dessus, mais celui-ci n'a pas besoin d'être exhaustif. En effet, l'applet doit en priorité être conçue pour permettre des ajouts aisés par la suite.