

Soutenance

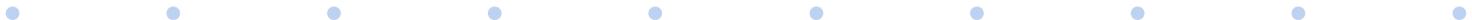
Développement d'Alipad

Version simplifiée pour Android du mode Allsky d'Aladin, l'Atlas 3D du CDS

Dominique Colnet



André Schaaff
Pierre Fernique



Plan de la soutenance

- Contexte
- Développement de prototypes
- Développement de l'application
- Conclusion et perspectives



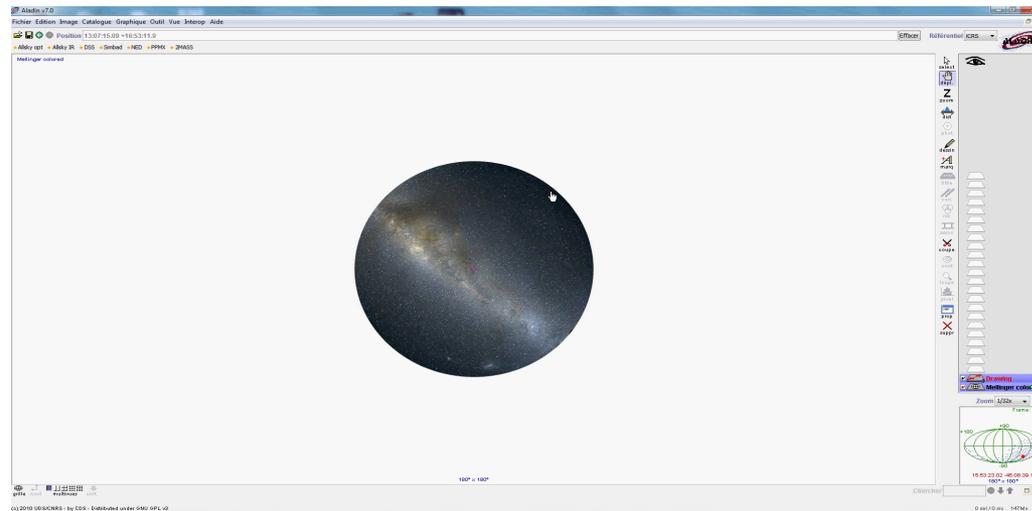
Contexte

- Présentation d'Aladin
- Concepts utilisés par Aladin



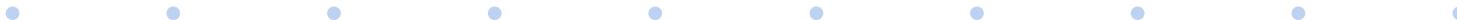
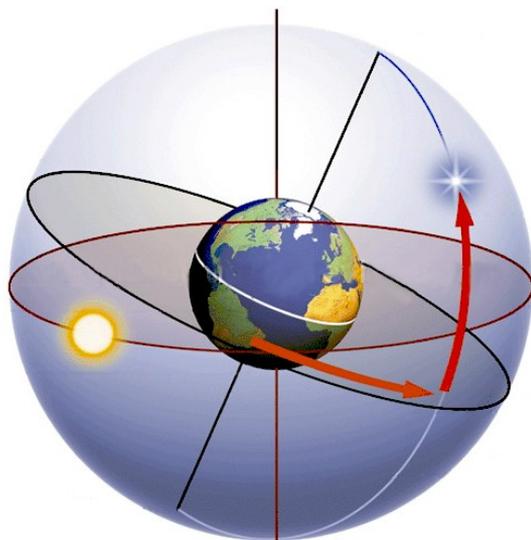
Aladin

- Carte des images et des données sur les étoiles
- Outils de calcul et d'analyse



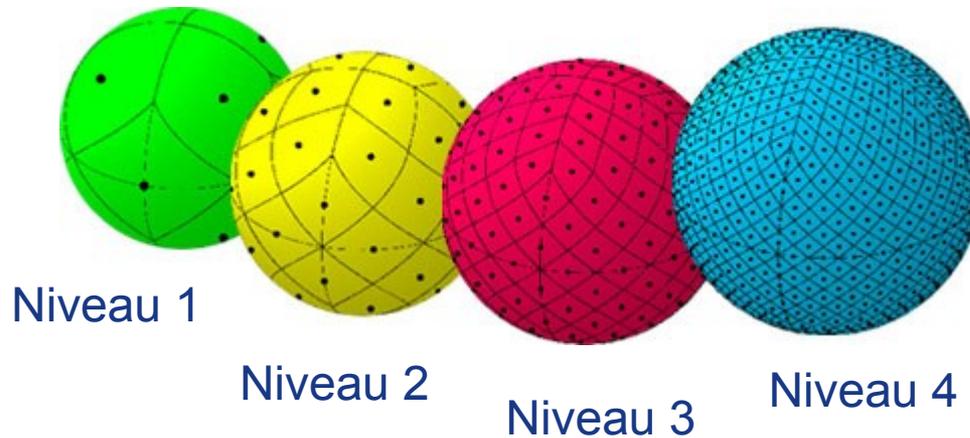
Représentation du ciel

- Coordonnées célestes
- Sphère céleste



Sphère Healpix

- Subdivision adaptée au "mapping" d'images
- Peu d'endroits de distorsion

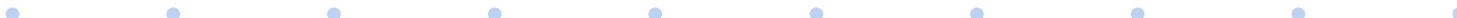
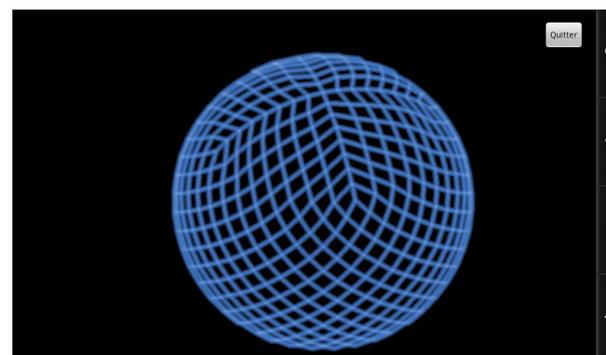
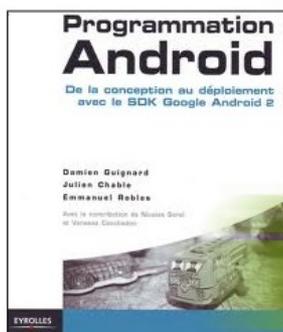
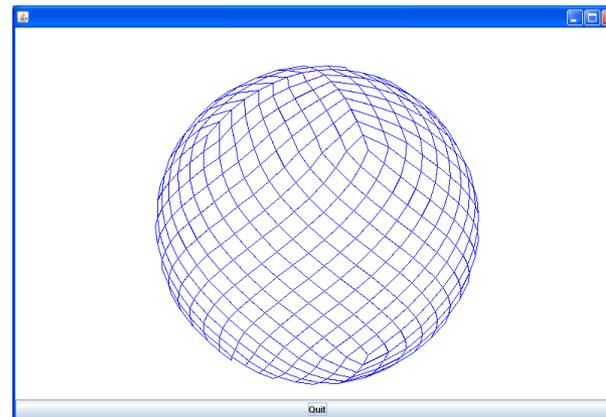


Développement de prototypes

- De Swing à Android
- Utilisation d'OpenGL-Es
- Application de comparaison

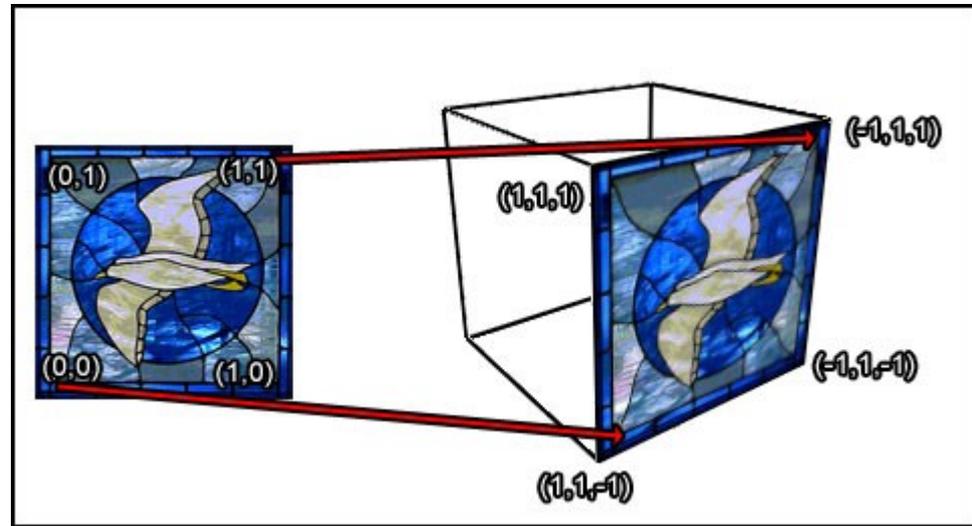


De Swing à Android



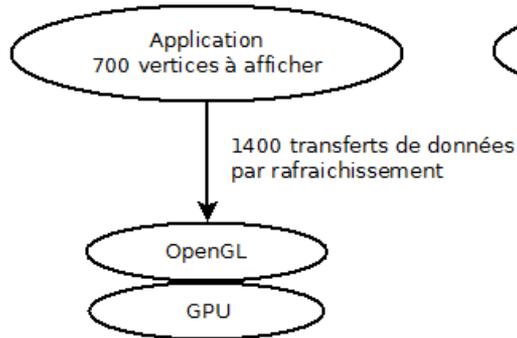
OpenGL-Es

- Vertex(ices)
- Faces
- Textures

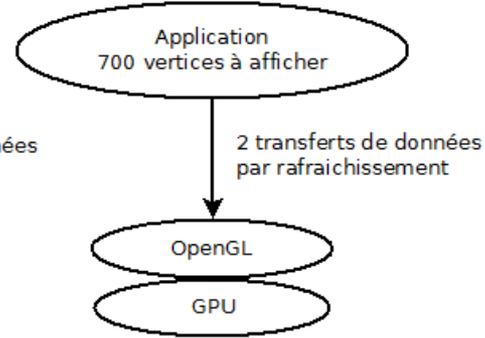


OpenGL-Es

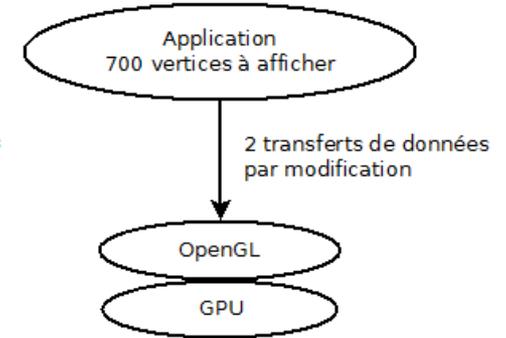
Basique (OpenGL-Es 1)



VertexArray (OpenGL-Es 1)



VertexBufferObject (OpenGL-Es 2)



Le modèle VertexArray a été choisi pour les raisons suivantes :

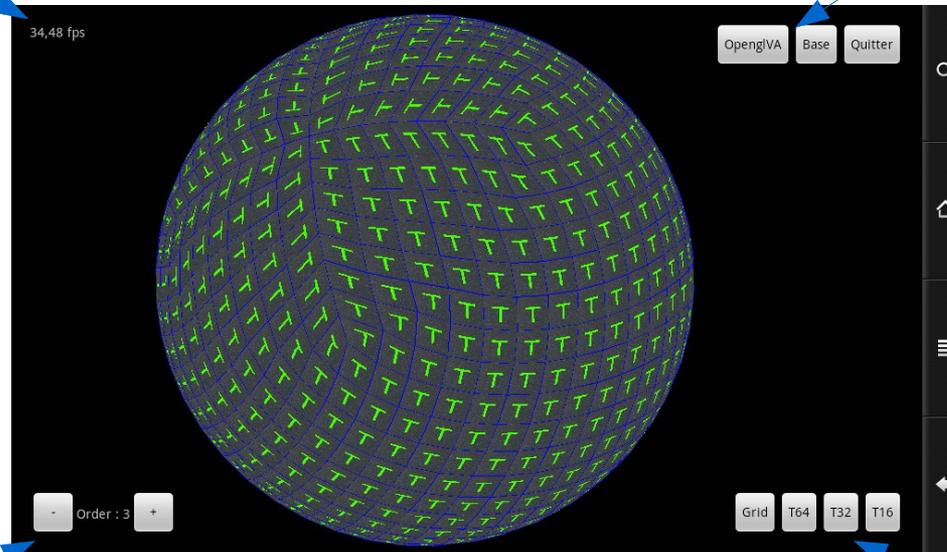
- Couche d'abstraction en moins par rapport aux VertexBufferObject (VBO)
- Compatible OpenGL-Es 1



Application de comparaison entre OpenGL-Es et la version originale

Performances (frames par seconde)

Choix du mode d'affichage



Choix du niveau de subdivision

Choix de la définition de texture



Application de comparaison

Résultats

| Niveau Subdivision | Version Normale (en fps) | Version OpenGL-Es (en fps) |
|--------------------|-----------------------------|-------------------------------|
| 1 | 9 | 350 |
| 2 | 2,2 | 125 |
| 3 | 0,8 | 30 |
| 4 | 0,2 | 9,5 |

Objectif : au moins 25 fps pour le niveau 3
afin d'obtenir un rendu fluide pour l'œil humain,
ce qui est obtenu avec OpenGL-Es (30 fps)

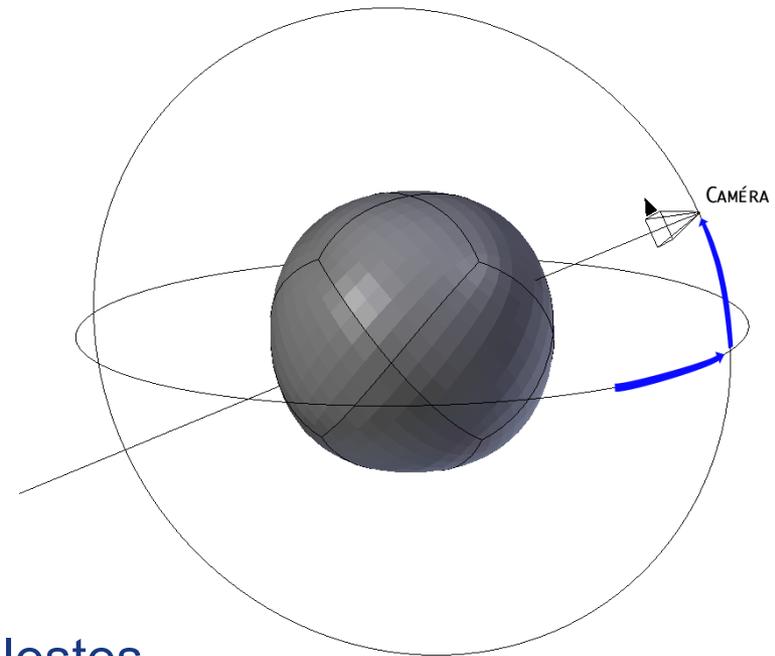


Développement de l'application

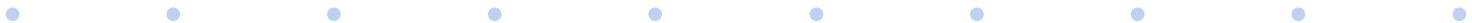
- Zoom et positionnement de la caméra
- Restriction du maillage
- Chargement des textures
- Interface graphique



Zoom et position de la caméra

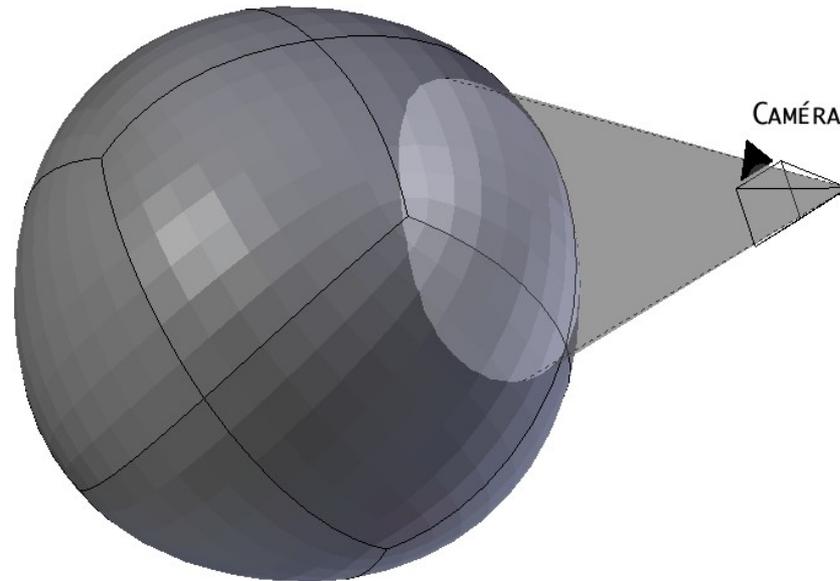


- Zoom
- Position en coordonnées célestes
- Vitesse de déplacement proportionnelle au placement courant

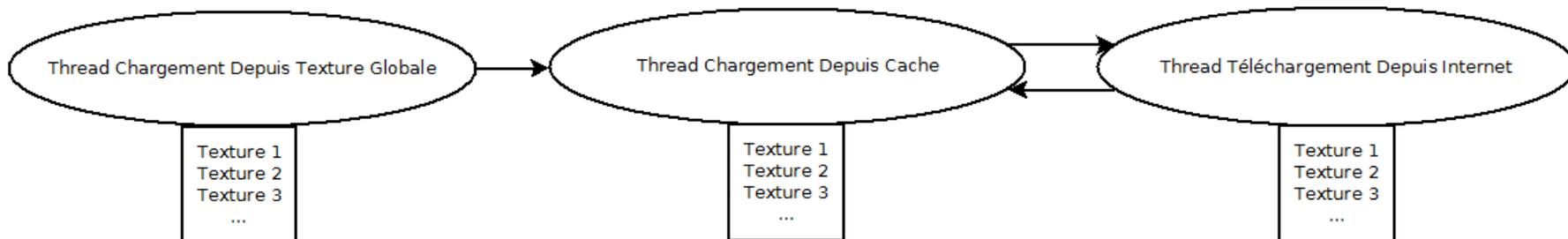


Restriction des faces

- Zone déterminée d'après la position de la caméra
- Structuration des données adaptée au modèle Vertex Array d'OpenGL

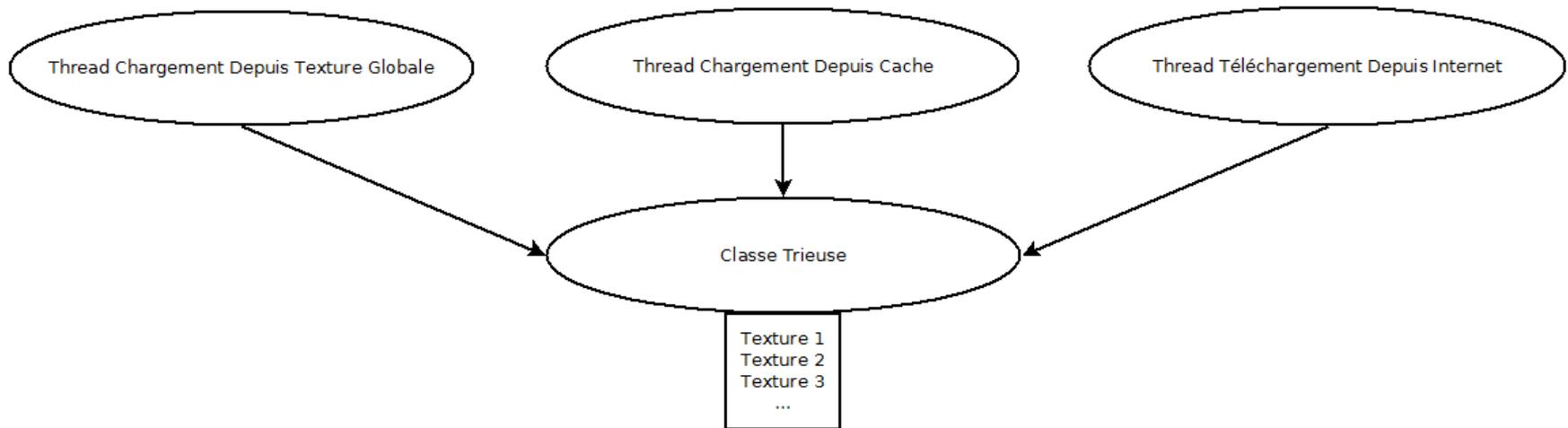


Chargement des textures modèle actuel



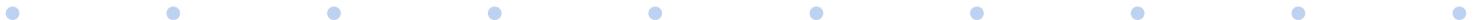
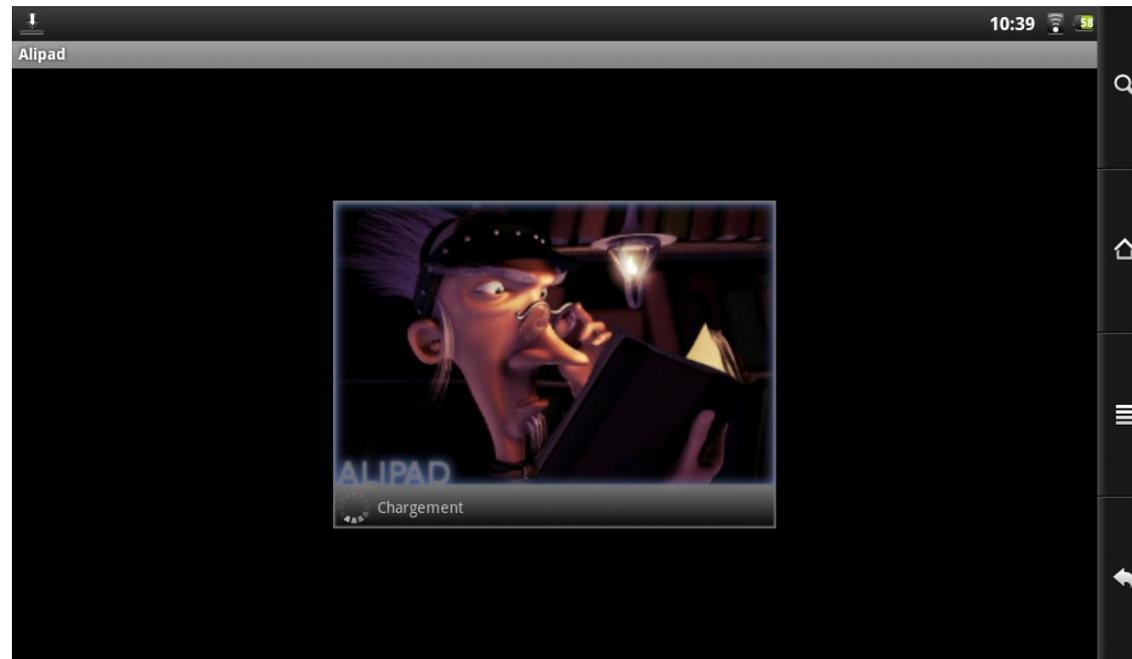
Chargement des textures

modèle envisagé



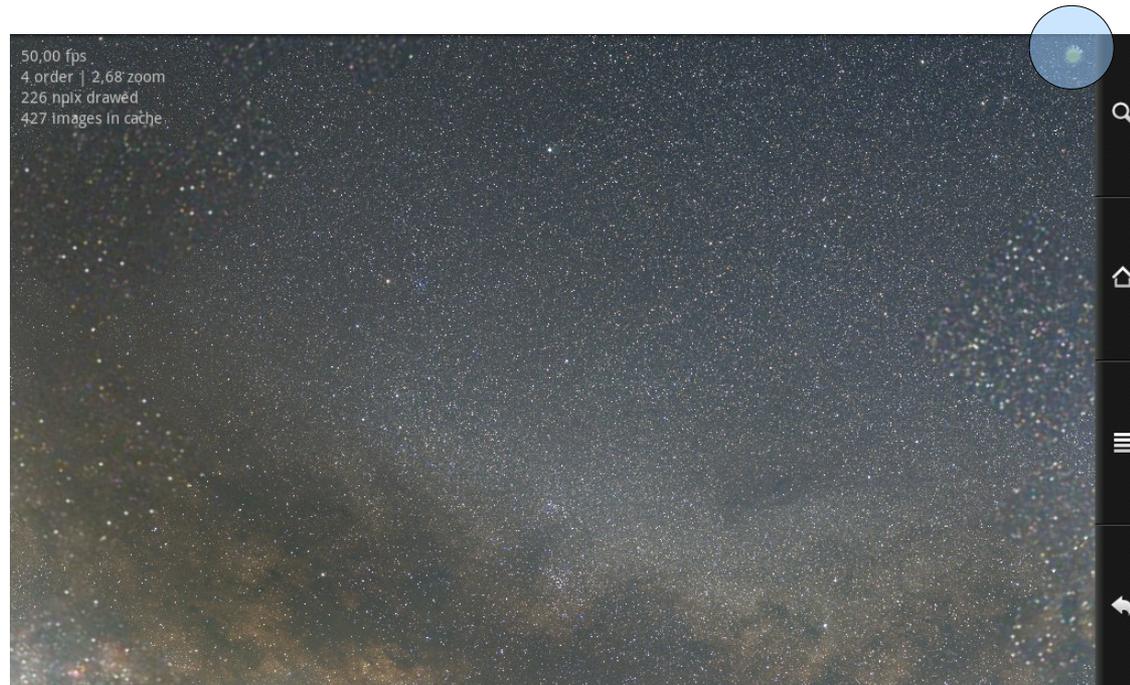
Interface graphique

Une "SplashScreen" pour précharger certains buffers



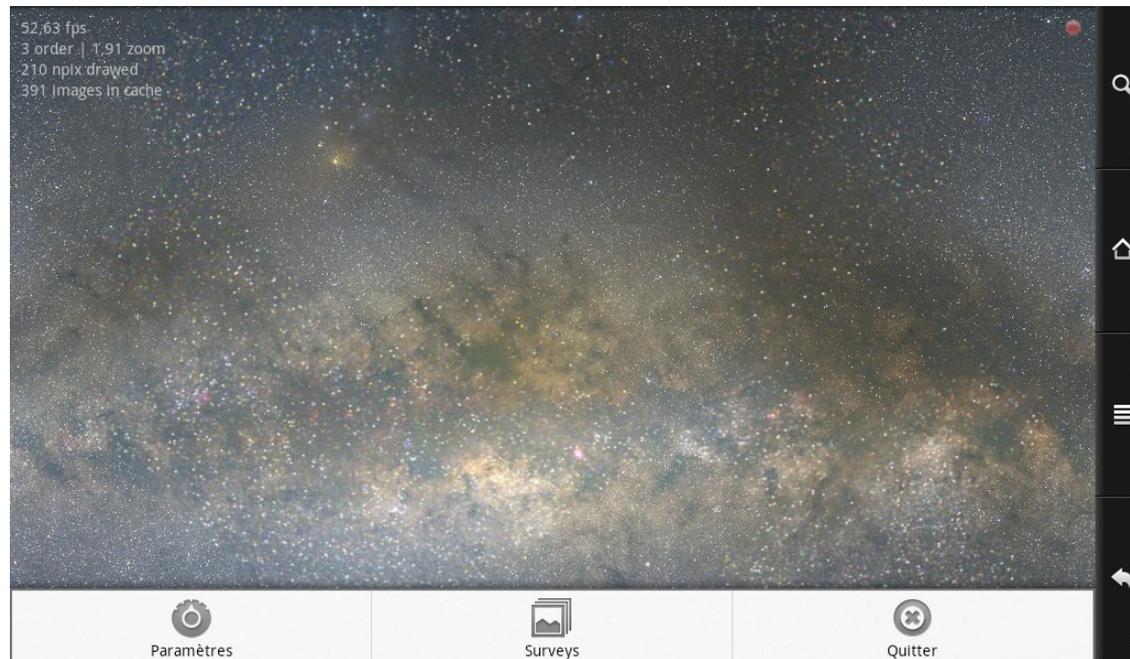
Interface graphique

Un indicateur de connexion et de chargement



Interface graphique

Paramétrage : Capacité du cache, Sélection du survey



Perspectives pour l'application

- Corriger les bugs liés aux Threads
- Affichage de données sur un objet céleste sélectionné
- Contrôle à l'aide des capteurs (boussole, gps et accéléromètre)



Conclusion

- Réponses en terme de faisabilité
- Application presque publiable
- Stage :
 - un sujet passionnant
 - excellent environnement de travail



