

Visualisation 3D de données astronomiques dans un navigateur Web, plus aspect Réalité Virtuelle

Jérôme Desroziers

Encadrant universitaire: Denis Roegel

Maître de stage: André Schaaff

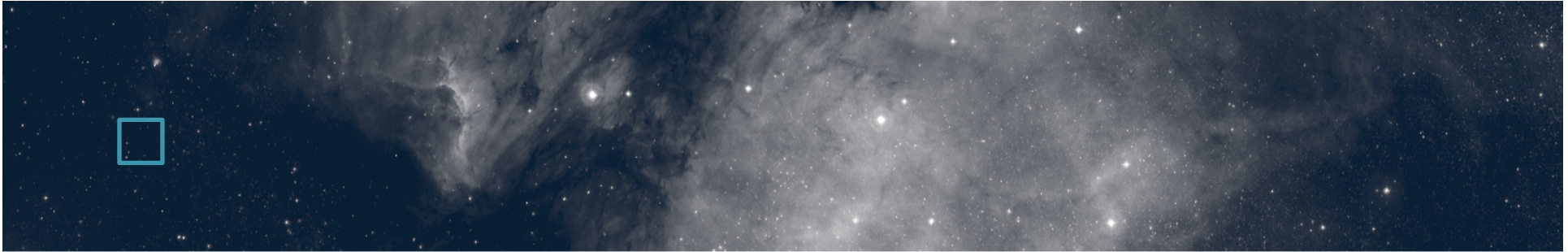




□ Sommaire

- Contexte
- Étude de l'existant
- Travail effectué
- Perspectives et conclusion



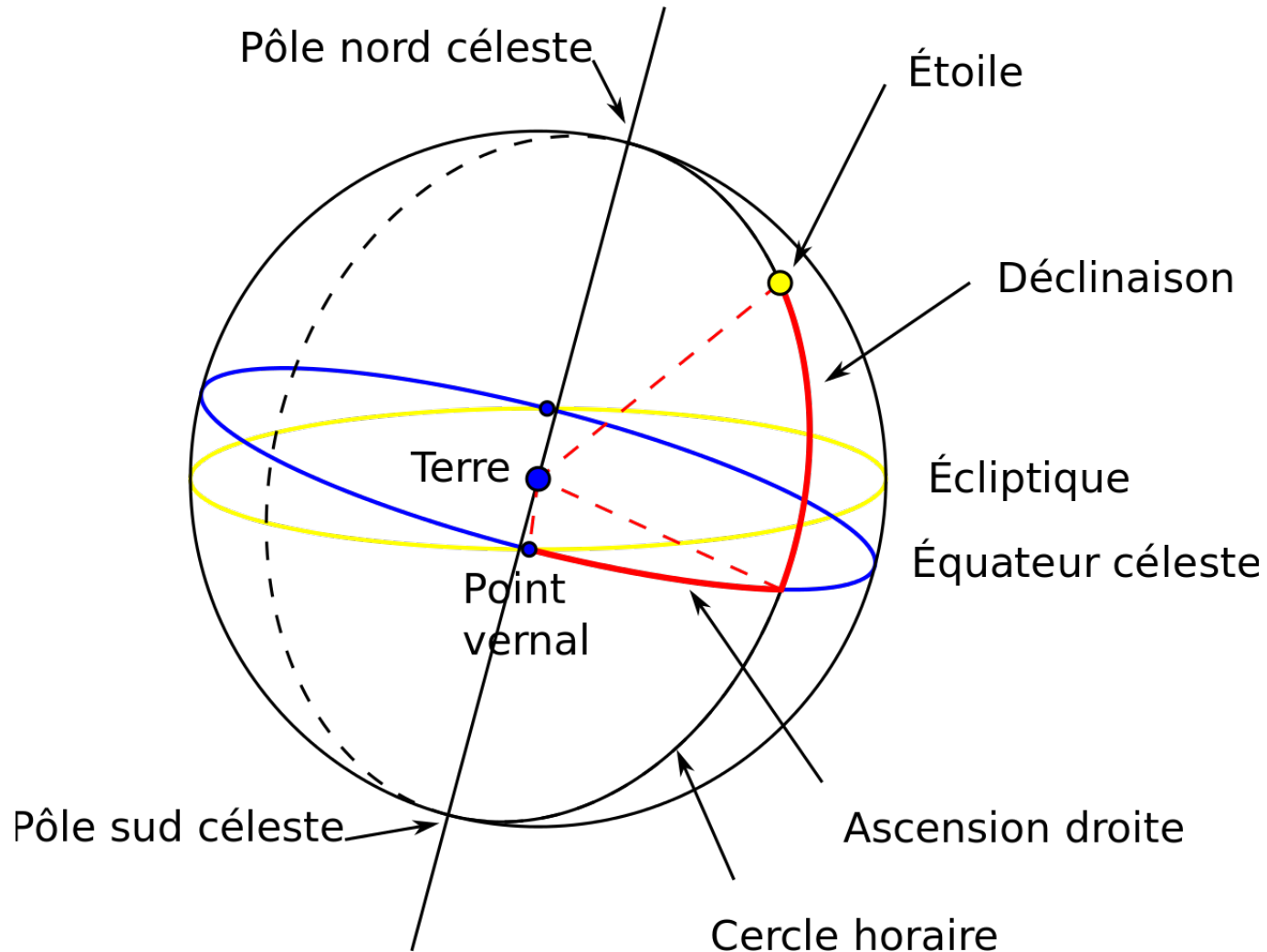


Contexte

- Détail sur le système de coordonnées
- Cadre
- Sujet de stage



□ Détail sur le système de coordonnées






□ L'observatoire astronomique de Strasbourg



- Dépend du CNRS
- Directeur: Hervé Wozniak
- 3 Équipes de recherche:
 - Galaxies
 - Hautes Énergies
 - CDS



□ Le Centre de données astronomiques de Strasbourg

-  **ALADIN**, atlas interactif du ciel, 85To d'images
-  **SINBAD**, plus de 8 millions d'objets du ciel
-  **VIZIER**, plus de 15000 catalogues issues de missions

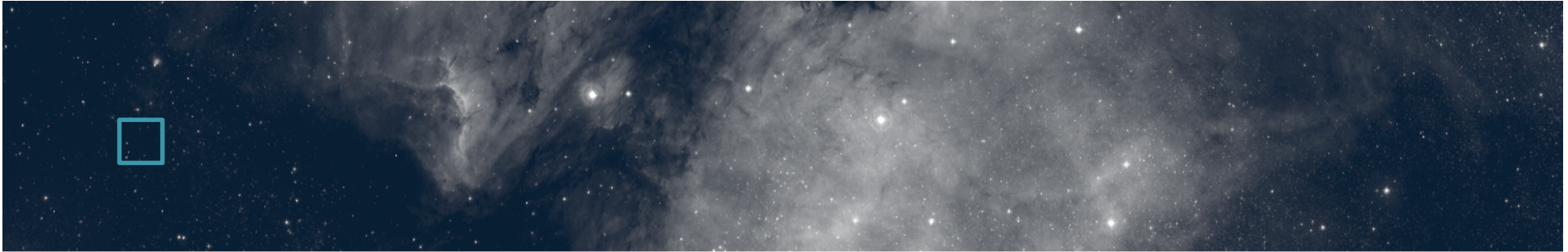
• Le CDS est un acteur majeur de l'IVOA
(International Virtual Observatory Alliance)



□ Sujet et objectifs du stage

- Poursuivre le développement (débuté en 2015) d'une application permettant de visualiser dans un navigateur des données issues de simulations numériques (formation d'une galaxie par exemple)
- Langage Javascript
- Ajout de fonctionnalités
- Correction et optimisation





Étude de l'existant

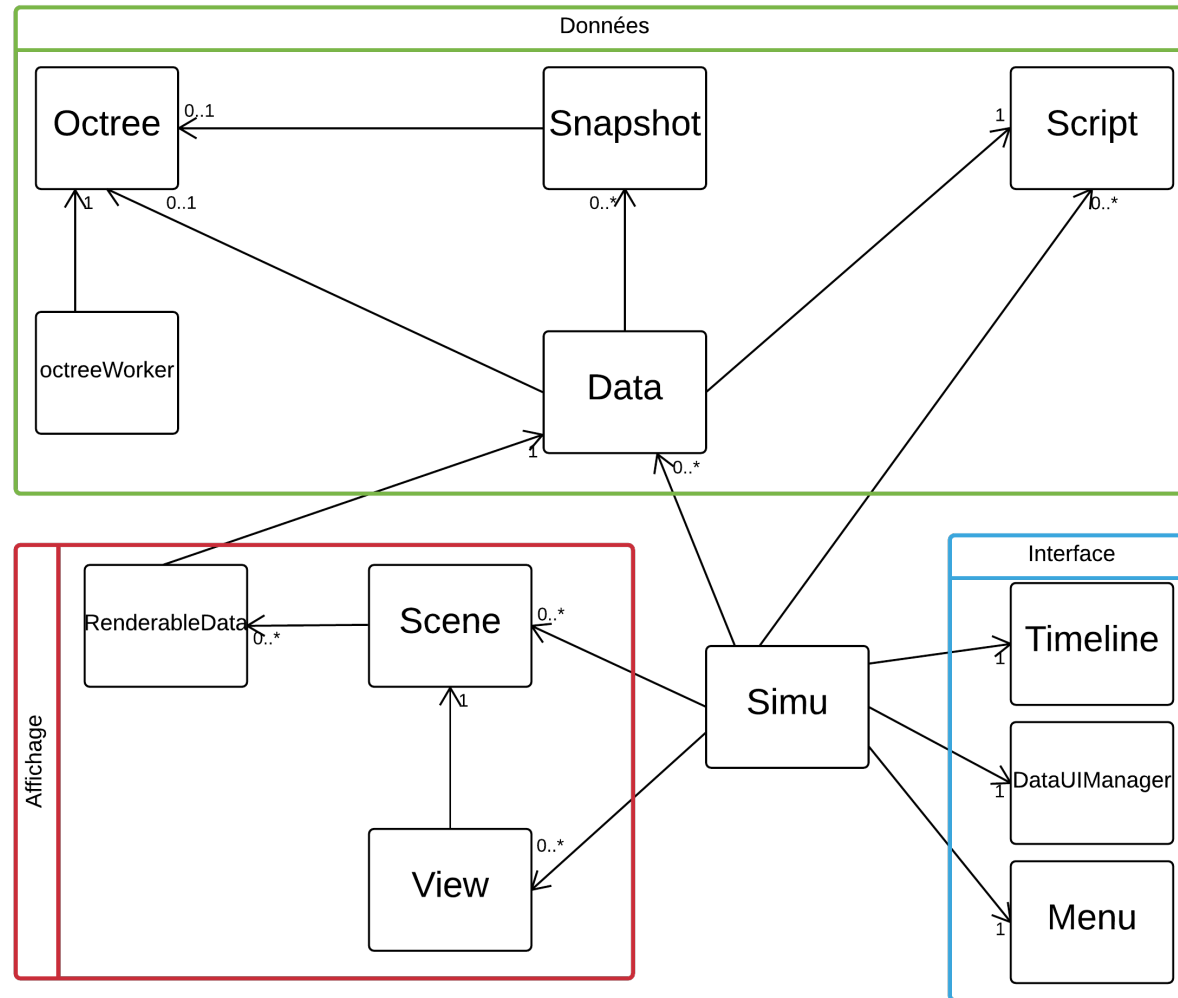
- Description de l'application
- Architecture de l'application
- Structure de données
- Aperçu de three.js



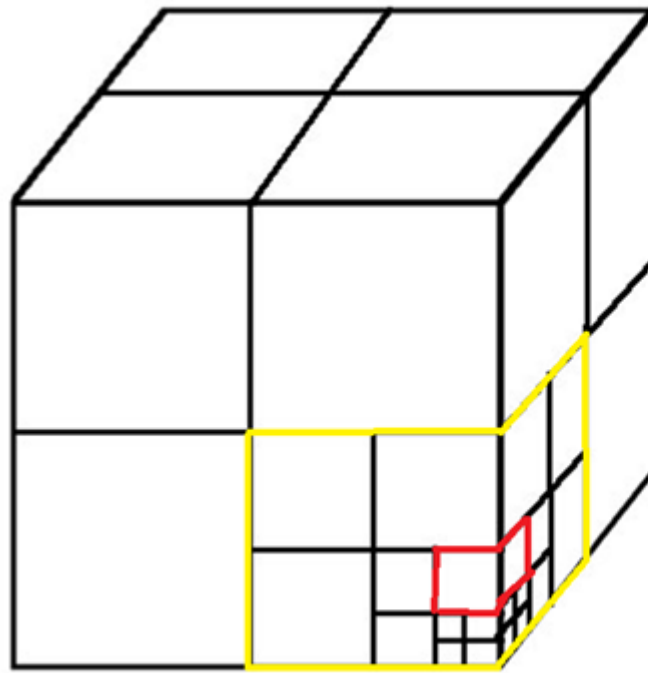
□ Description de l'application



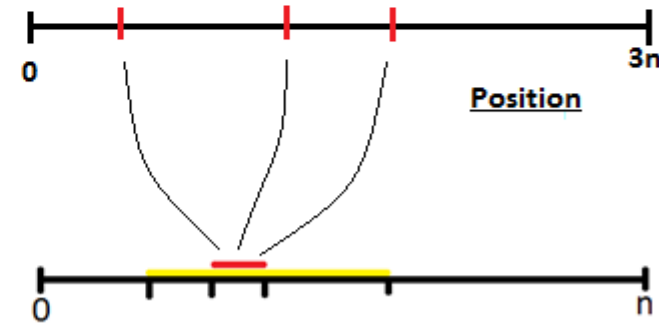
□ Architecture de l'application



□ Structure de données et principe de l'Octree



Octree



Index



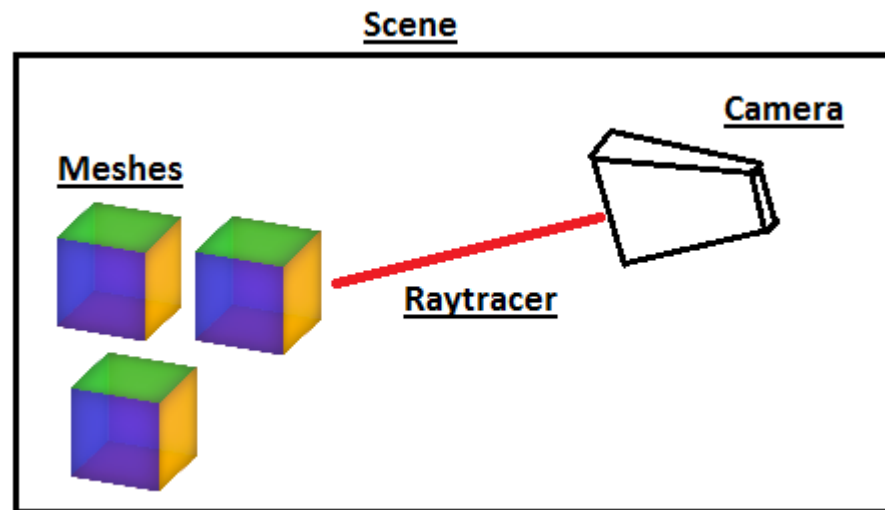
□ Librairie Javascript

- Three.js

- Librairie Javascript pour la modélisation et l'animation d'objets en 3 dimensions

- Basée sur la manipulation de 4 types d'objets

- Mesh
 - Raycaster
 - Scene
 - Camera

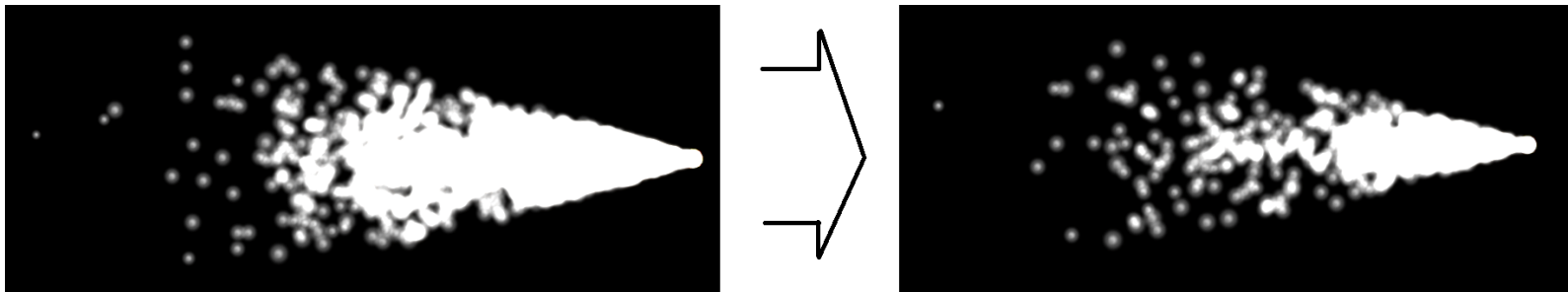


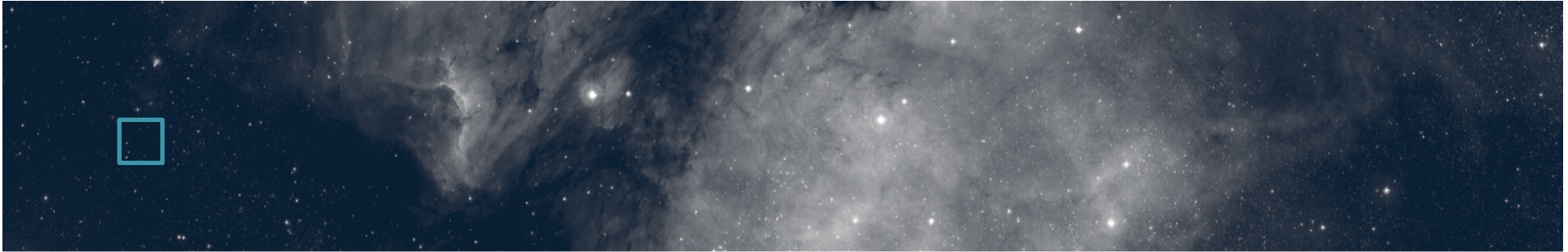
□ Librairie Javascript (2)

•Three.js

–Cas particulier de l'application: le pointCloud

- Mesh « fait main »
- Appels à la méthode drawCall
- Affichage réalisé à partir de couples indices / buffers



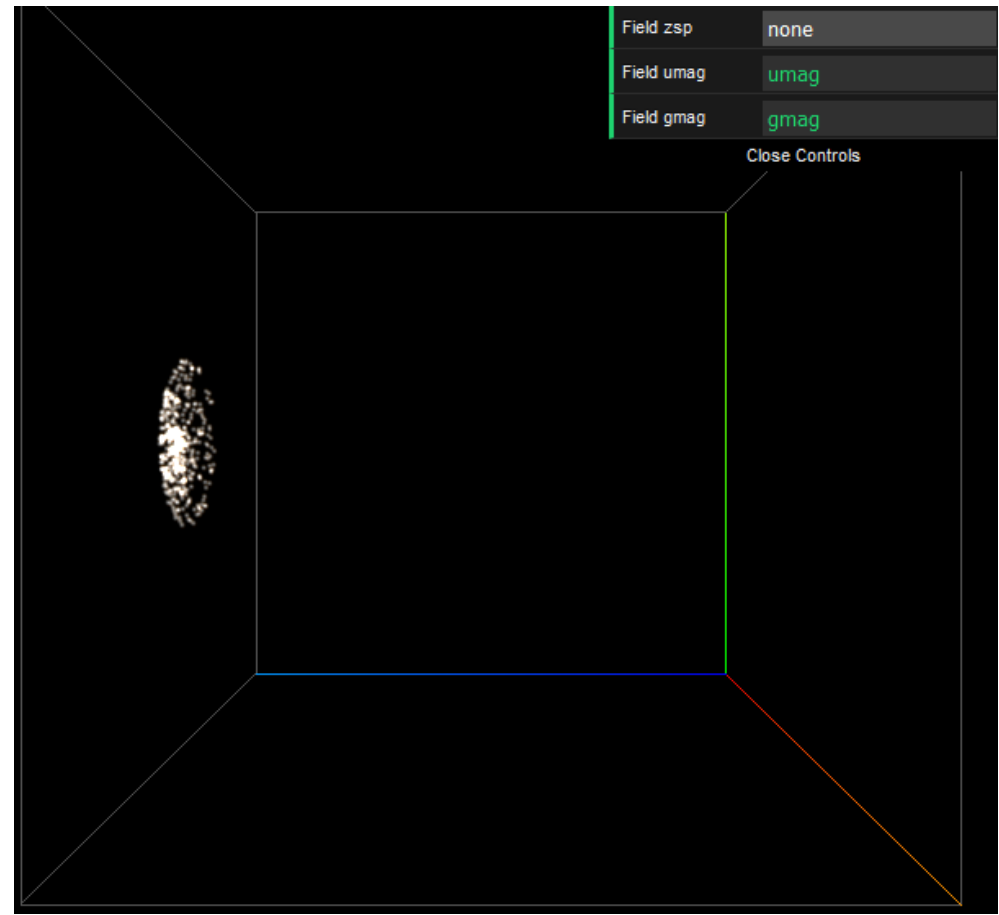
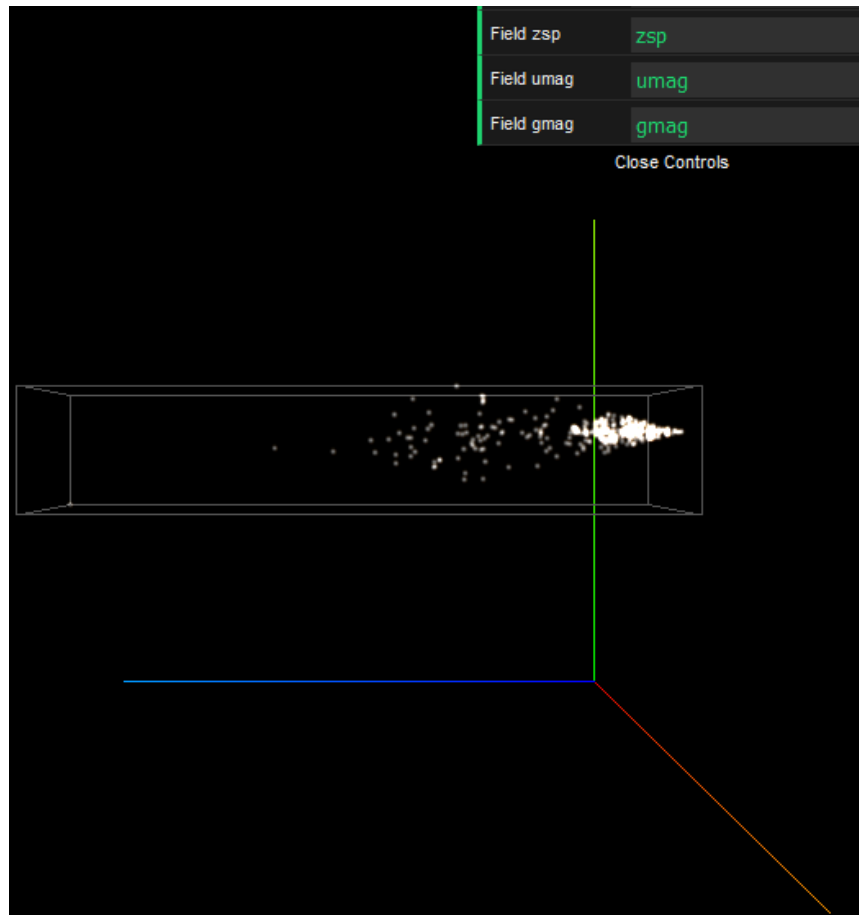


Travail effectué

- Chargement des données VizieR
- Homogénéisation des informations
- Homogénéisation spatiale
- Fonctionnalité de zoom
- De nombreux autres ajouts...



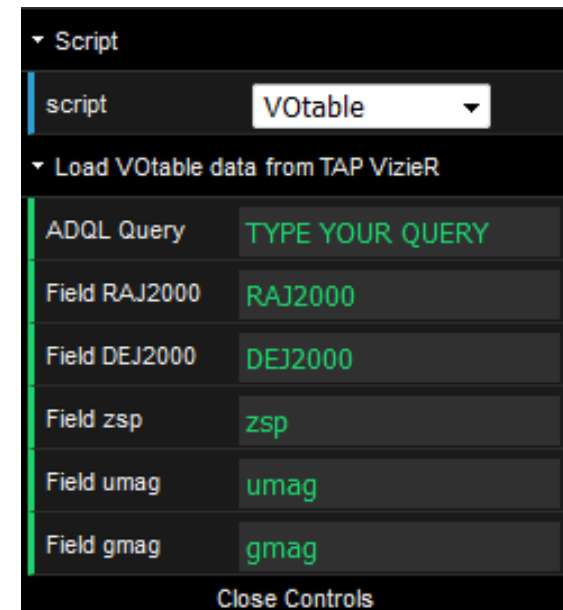
□ Visualisation de données de VizieR



□ Visualisation de données de VizieR – Accès aux données

- Objectif: permettre l'affichage et la manipulation de données provenant de relevés de mission (ex: SDSS7) par l'application

Requêtes effectuées en ADQL
(Astronomical Data Query Language,
recommandation IVOA)
sur VizieR



The screenshot shows the VizieR interface for loading data. It features a 'Script' dropdown menu set to 'VTable'. Below it, a section titled 'Load VTable data from TAP VizieR' contains several input fields for ADQL queries and field names. The fields are: 'ADQL Query' with the placeholder 'TYPE YOUR QUERY', 'Field RAJ2000' with 'RAJ2000', 'Field DEJ2000' with 'DEJ2000', 'Field zsp' with 'zsp', 'Field umag' with 'umag', and 'Field gmag' with 'gmag'. A 'Close Controls' button is located at the bottom right of the panel.

Field	Value
ADQL Query	TYPE YOUR QUERY
Field RAJ2000	RAJ2000
Field DEJ2000	DEJ2000
Field zsp	zsp
Field umag	umag
Field gmag	gmag

□ Visualisation de données de VizieR – Accès aux données (2)

- Au format VOTable (recommandation IVOA)
- Coordonnées sphériques
$$X = zsp * \cos(\text{RAJ2000}) * \cos(\text{DEJ2000})$$
$$Y = zsp * \sin(\text{RAJ2000}) * \cos(\text{DEJ2000})$$
$$Z = zsp * \sin(\text{DEJ2000})$$
- Données non centrée
 - Définir le centre des données et la longueur maximum sur x, y et z
 - Translater les données par rapport à l'origine et redimensionner
 - Translater par rapport au centre du cube de données



□ Visualisation de données de VizieR – Echanges avec VizieR

- Le chargement des données est basé sur l'utilisation de XMLHttpRequest

- i) `xhr.open("POST", URL, true);`

- `("http://tapvizier.u-strasbg.fr/TAPVizieR/tap/async")`

- ii) `xhr.setRequestHeader("Content-Type", TYPE);`
`("application/x-www-form-urlencoded")`

- iii) Enfin, on utilise

- `xhr.send("request=doQuery&format=votable&lang=adql&phase=RUN&runid=vQuery&query=" + request)`



□ Visualisation de données de VizieR – Echanges avec VizieR (2)

- Envoi de la requête (réponse: requête correcte ou non)
- Si correcte, échanges avec le serveur toutes les 2 secondes pour accéder au statut de la requête
- Requête complète: envoi de l'URL résultat par le serveur
- Application du script VOTable aux données reçues
 - N.B.: nécessite un parcours d'arbre XML, utilisation du parseur du CDS dans le cas des données VOTable

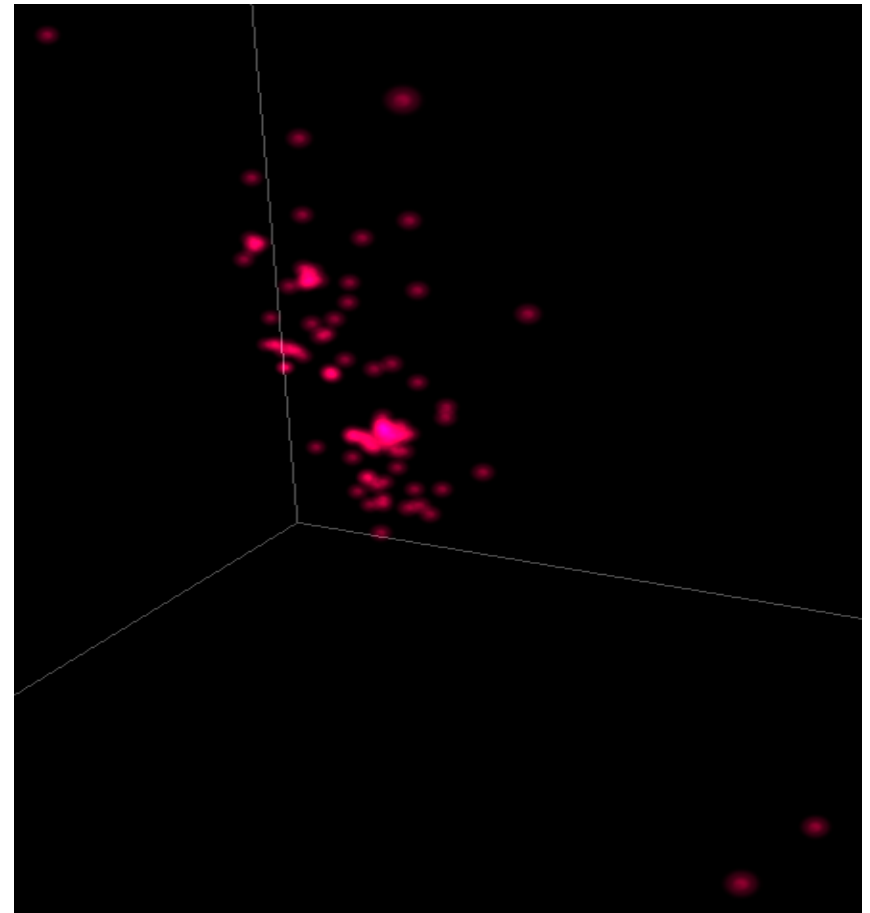
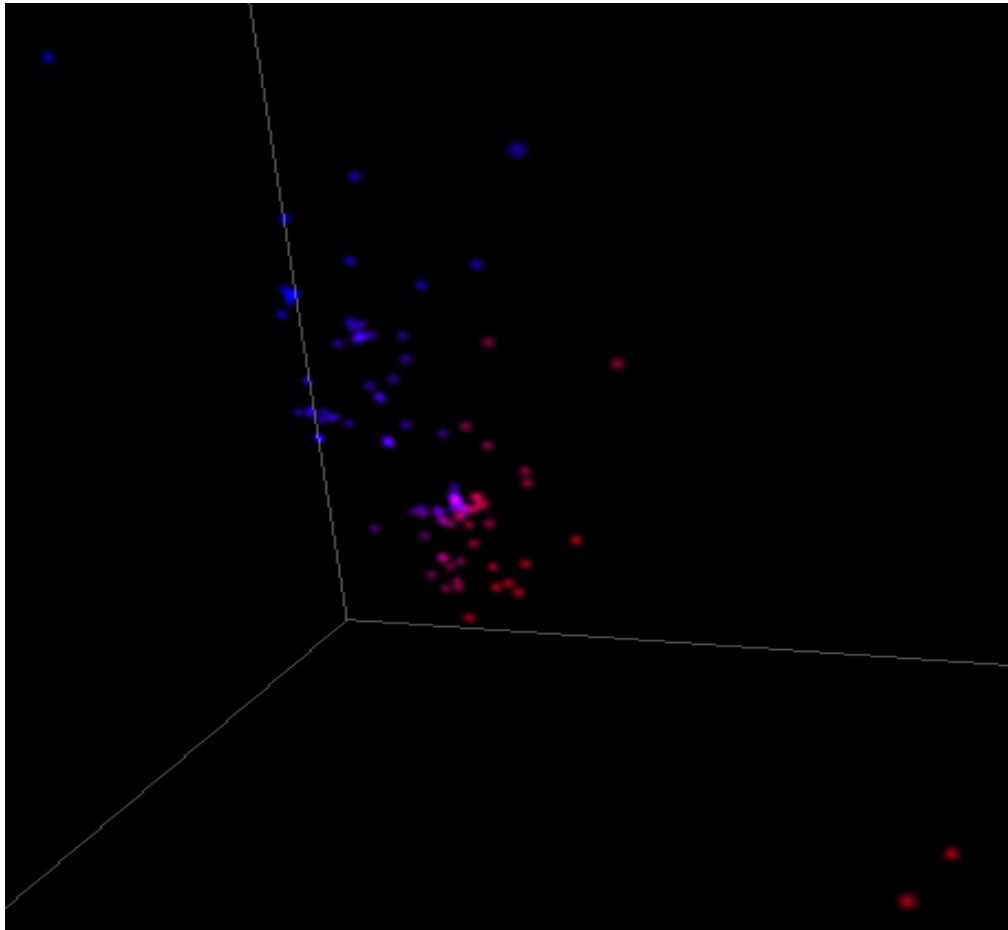


□ Homogénéisation - Informations

- Fonctionnalité de mise en valeur des éléments via un attribut donné
- Problème: pour les données VOTable, les bornes de chaque attribut sont locales à chaque jeu de données
- Solution mise en place:
 - Chaque vue possède son propre ensemble de bornes min/max pour chaque attribut qu'elle possède
 - Mise à jour des bornes et de l'affichage lors de changements dans les données
 - Fonctionnalité de « merge »



Homogénéisation – Informations (2)

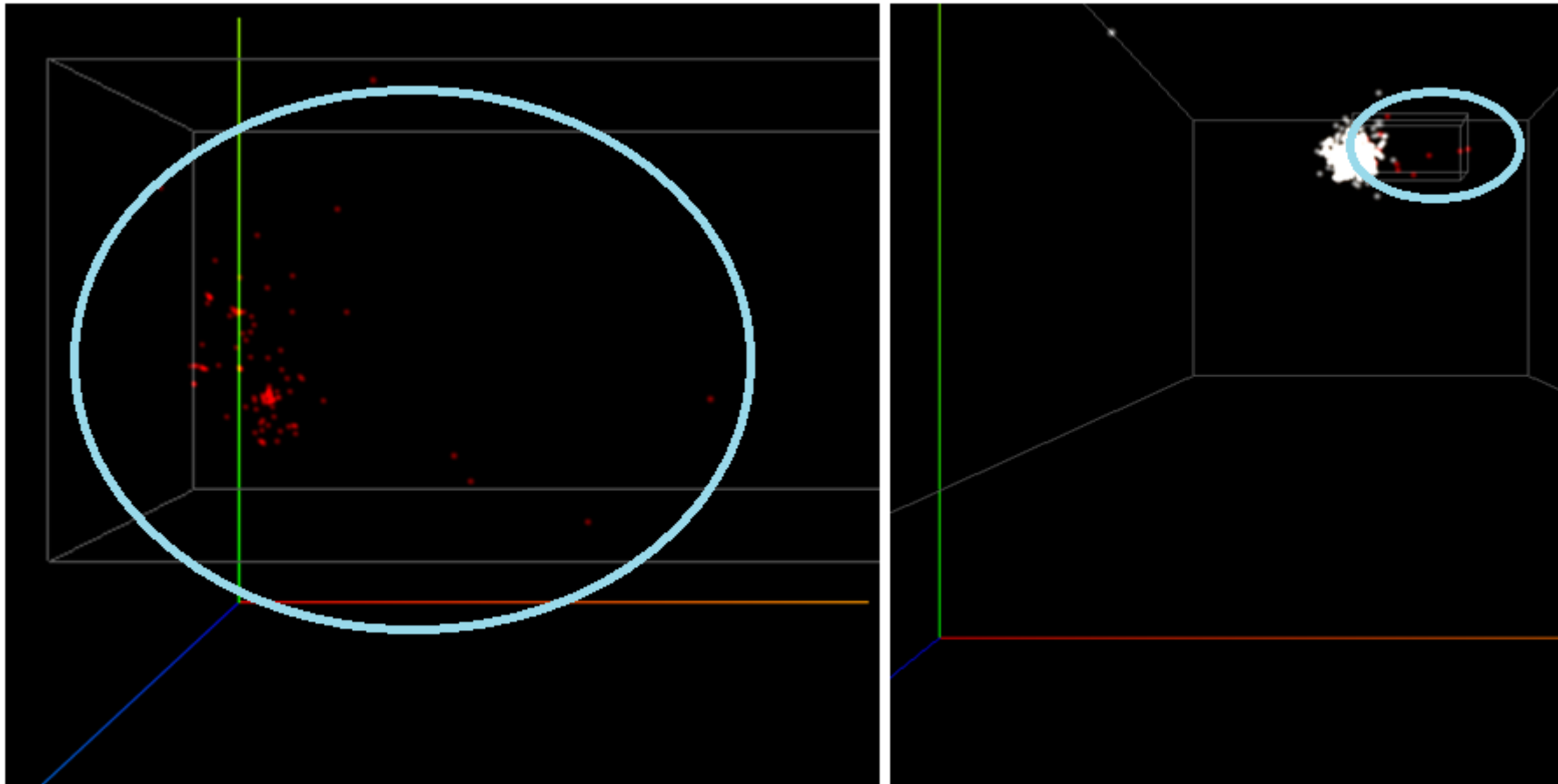


□ Homogénéisation spatiale

- Problème: chaque jeu de données VOTable est centré dans le cube de données; différents jeux ne sont pas proportionnels entre eux
- Solution mise en place
 - Chaque vue possède son propre ensemble de bornes positions min / max xyz
 - Mise à jour des bornes et de l'affichage lors de changements dans les données



□ Homogénéisation spatiale (2)

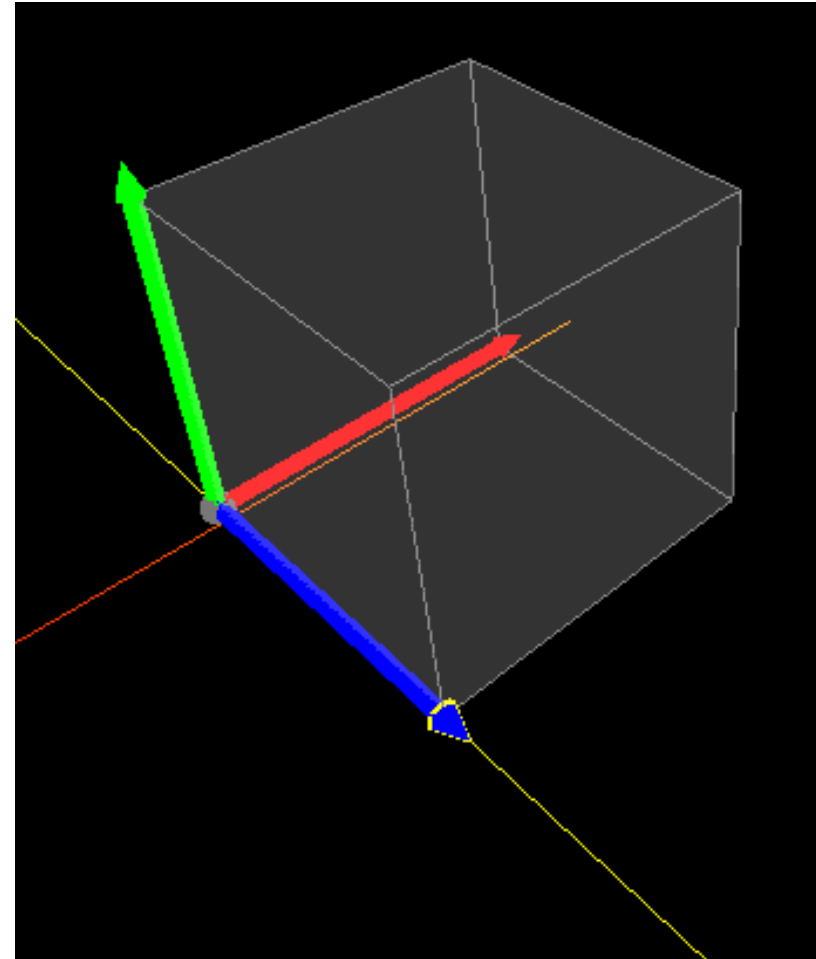
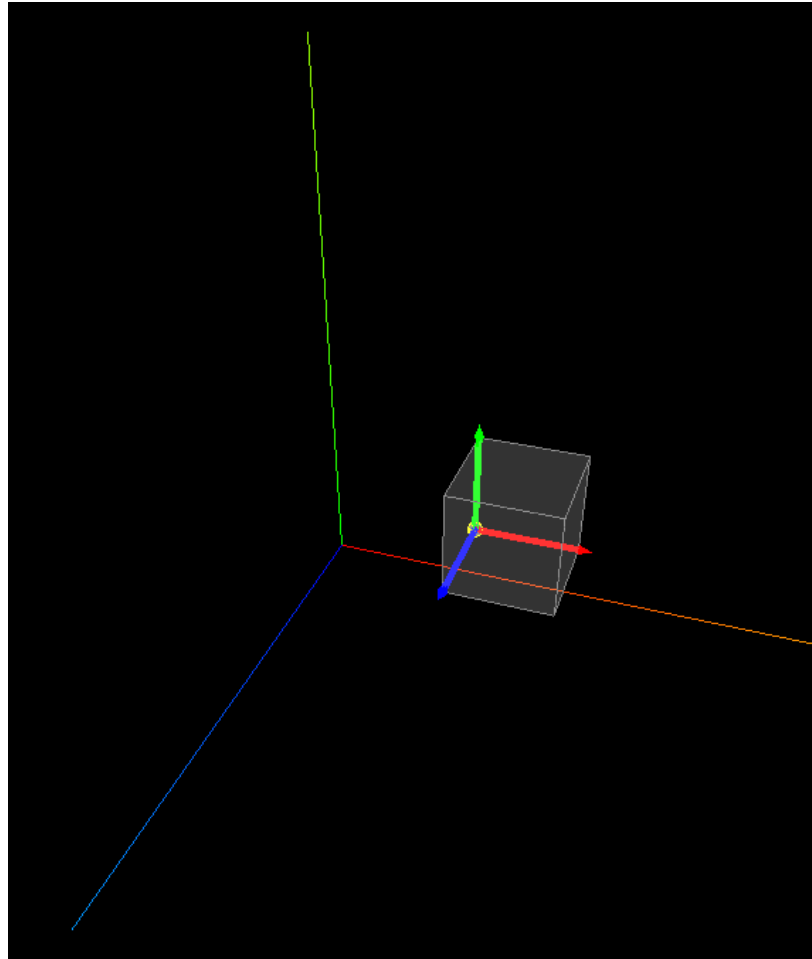


□ Fonction de Zoom

- Doit permettre à l'utilisateur d'effectuer une sélection sur une zone ciblée des données affichées
- La sélection doit prendre en compte tous les jeux de données présents dans la zone sélectionnée
- L'outil de zoom doit être utilisable sur n'importe quel jeu de données, 3D ou non, à $t=0$ ou non



□ Fonction de Zoom (2)

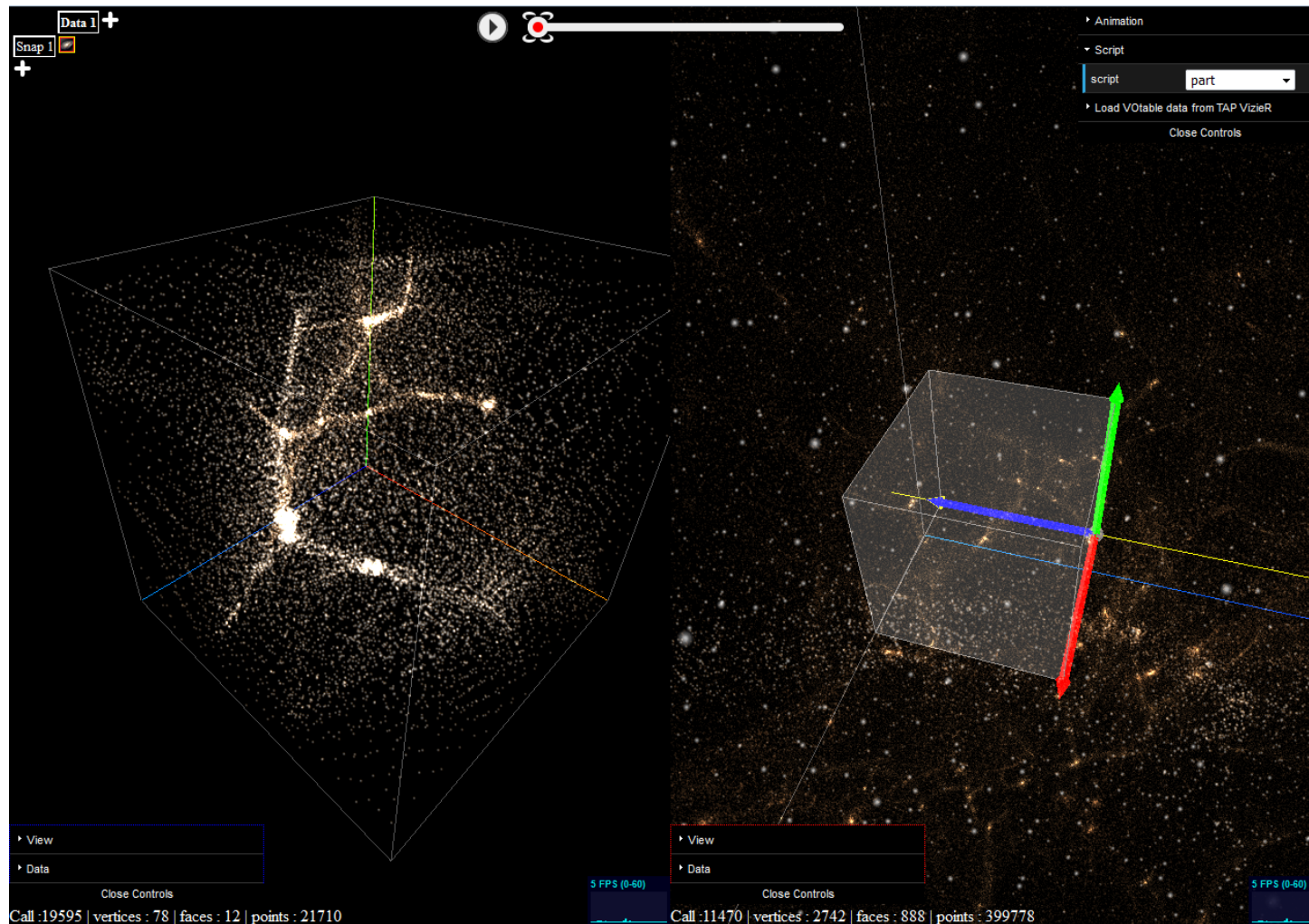


□ Fonction de Zoom (3)

- Basée sur le principe du mode multivues
 - La 2nd vue devient l'esclave de la première (maître) durant le zoom
 - Sauvegarde des paramètres de l'esclave
 - La vue maître crée les données du zoom (octree, buffers) et les stocke dans les objets Data correspondants
 - La vue esclave récupère ces données stockées
 - Certaines fonctionnalités sont désactivées (animation notamment)



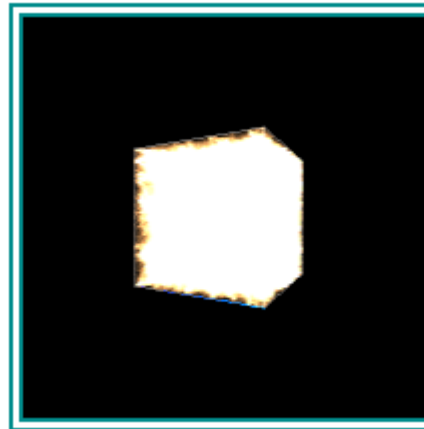
□ Fonction de Zoom, aperçu



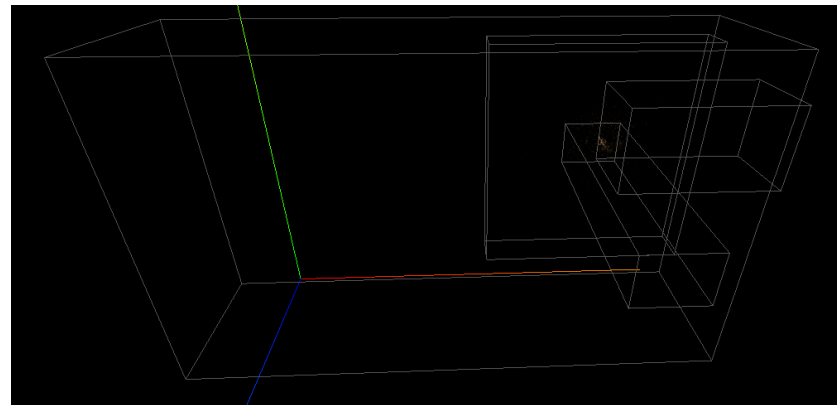
☐ Autres

- Ajouts:

- Mode widget



- Boîte englobante



□ Autres (2)

- Corrections et améliorations:
 - Problème du niveau de détail
 - Gestion de l'Octree lors de l'animation
 - Divers problèmes d'affichage (redimensionnement, chargement des données, position cyclique, BoundingBox...)
 - Amélioration de l'interface



□ Conclusion et perspectives

- Stage extrêmement intéressant
- De nombreuses évolutions possibles

- ... un outil ouvert

