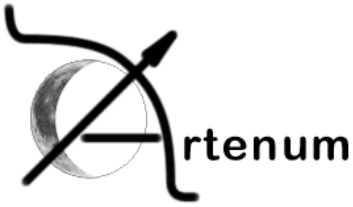


Cassandra 2.0
Tutorial V1.0

Sébastien Jourdain, Fatiha Zeghir

2005/06/01

Abstract

Cassandra is a generic VTK data viewer written in Java which provides native multi-platform support. Cassandra is the result of internal R&D of Artenum company around scientific data visualisation and its know-how in VTK [1][2]. The use of Cassandra is dedicated to 3D visualisation in pre/post processing. Cassandra has already been used for many application in scientific computing and space environment analysis.

Products and services

More products and services are available at Artenum company. Please visit on our Web site or contacts us.

Project manager : cassandra@artenum.com / jourdain@artenum.com

Cassandra's Web page

http://www.artenum.com/cassandra

Cassandra's development area

http://artenum.libresource.net/projects/cassandra

Artenum VTK and Java development page

http://www.artenum.com/en/services/vtk.php
http://www.artenum.com/en/services/java.php

Contact

Artenum SARL,
Technopôle Paris CyberVillage,
101-103 Bd Mac Donald
75019, Paris, France
Phone: +33 (0)1 44 89 45 15
Fax: +33 (0)1 44 89 45 17
E-mail: contact@artenum.com
http://www.artenum.com

Table of contents

I. Introduction

Cassandra is a generic VTK data viewer written in Java which provides native multi-platform support. Cassandra is the result of internal R&D of Artenum company around scientific data visualisation and its know-how in VTK [1][2]. The use of Cassandra is dedicated to 3D visualisation in pre/post processing. Cassandra has already been used for many application in scientific computing and space environment analysis.

Cassandra already provides several processing filters such as cutting functions, iso-levels and data threshold. Therefore, those filters can be easily extended by external plug-ins. Plug-ins can be any kind of data processing such as import/export module or VTK pipeline definition. Cassandra is actually a VTK framework providing the common features on generic VTK objects and simplifies the creation of external plug-ins. The existing processing filters are external plug-ins automatically loaded at the runtime. The plug-in loading can be done any time, even after the application starts. This feature allows a wide flexibility in the definition of filters and provides a highly customisable generic VTK viewer.

II. Key features

1) Pipeline viewer and editor

Cassandra is composed of a graphical pipeline editor that simplifies the understanding and the manipulation of the data processing and visualisation. This graphical representation of the VTK pipeline provides a rich interaction between the users and the data to process.

2) Dynamic plug-ins system

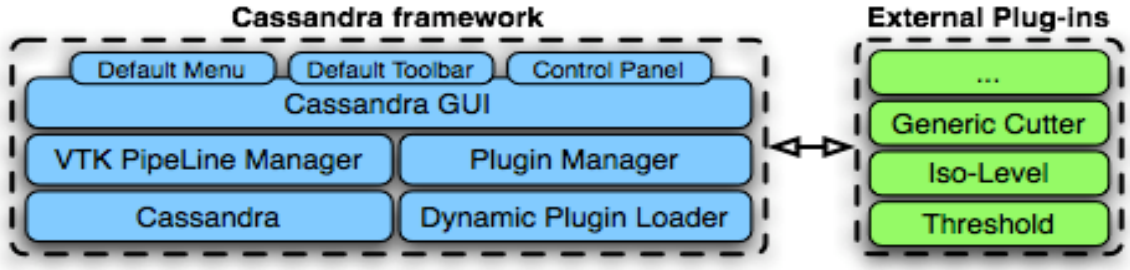
Cassandra works on top of a dynamic class loader, which allows dynamic class integration such as plug-ins with dedicated filter processing.

3) Dynamic object manipulation

Cassandra integrates an advanced Python console for the hot manipulation of Java or Python objects. This console can be used to load Python script in order to extend the plug-in approach with the script approach. It has become possible to write a script in order to generate animation and export them as a set of images or movies.

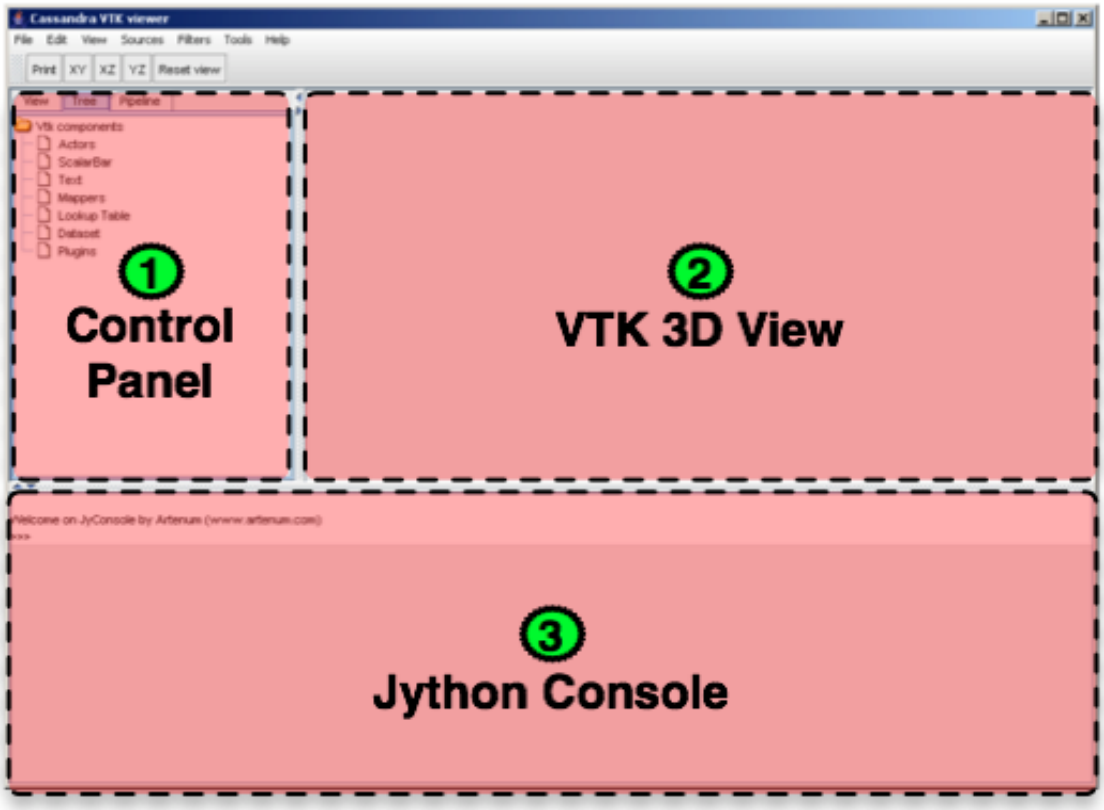
4) Cassandra architecture

Cassandra provides a powerful framework for manipulating VTK data. It is based on a dynamic class loader for loading external plug-ins at runtime and a powerful VTK pipeline manager that allows a graphical representation and manipulation of the VTK pipeline.



5) Cassandra GUI

The Cassandra Graphical interface is divided in three main areas which have different usage. The first one is used to manipulate and change the input data. The second one is the 3D view which is used to manipulate the 3D object by rotating and scaling it. The last one is the console which allows hot manipulation of Java and Python objects. Each graphical section can be hidden or reduced in order to enlarge or view another component.



III. Cassandra usage

1) Launching Cassandra

Cassandra can be launched in two different ways on each system. Each system has its own launching script which will automatically set the global variable for the local VTK binaries. If you already own a VTK installation, Cassandra can be directly launched from its JAR file by double clicking on it or by typing "**java –jar run.jar**".

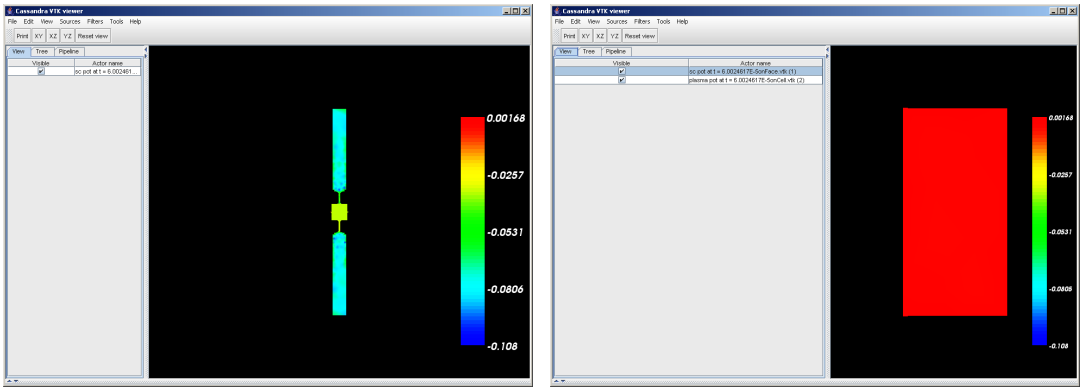
Trouble shooting:

On some Windows system, the "run.bat" does not seem to work, so you will have to set manually the VTK system variable (LD_LIBRARY_PATH, PATH) and launch Cassandra directly by double clicking on it.

2) Loading a VTK file

Go to the “File” menu then “Open”, and select your input VTK file. By default, you will only see the file that ends with the VTK extension. You can repeat this operation in order to load several data in the same view.

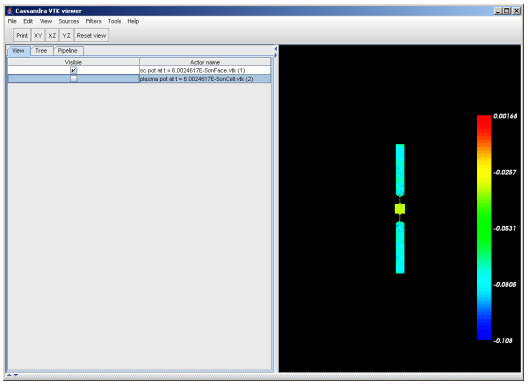
In our example, we will load two files, the first one will be a satellite and the second one its surrounded plasma environment with the electrical potential as the data visualisation.



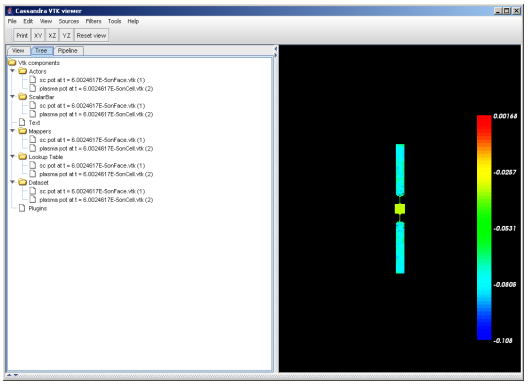
Once the files are loaded, you should be able to see your data inside the 3D VTK panel on the right side of the Cassandra GUI. You can rotate, scale and move it with the different mouse movements and buttons.

On the left side you will find several tools for manipulating your data.

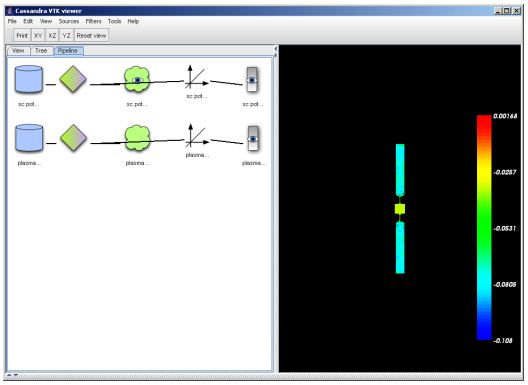
- In the View tab you are allowed to show or hide quickly 3D objects.



- In the Tree tab you can reach the structured tree of objects where you can access their dedicated contextual menu.



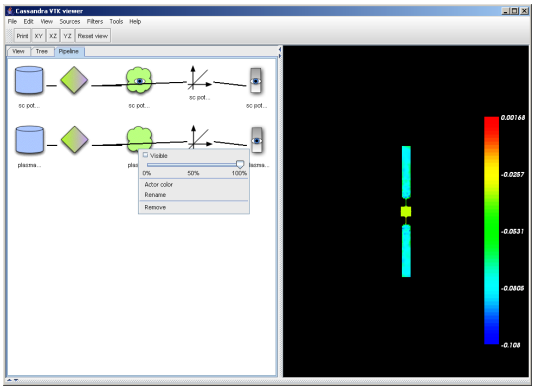
- In the Pipeline tab you can see the data processing with the links between filters, data and objects with direct access to their contextual menu for their specific properties in order to manipulate them…

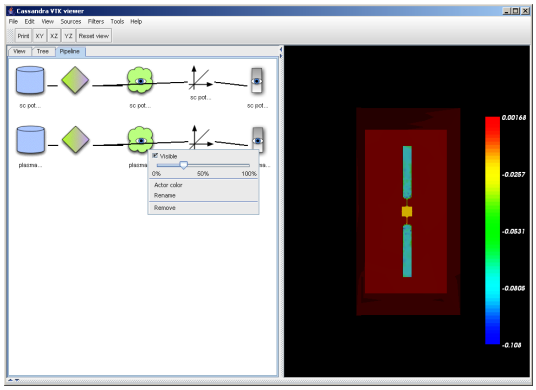


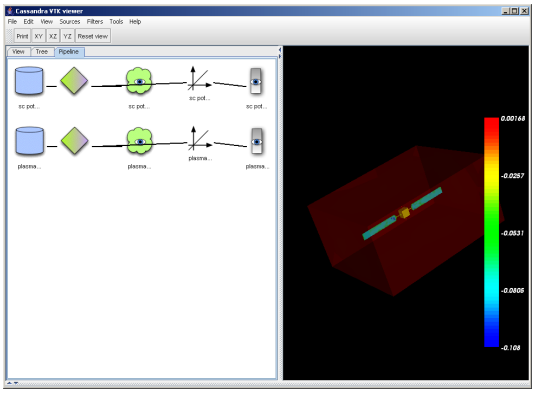
Most of the time we will switch to the Pipeline view for a better understanding of what we see, and what we do.

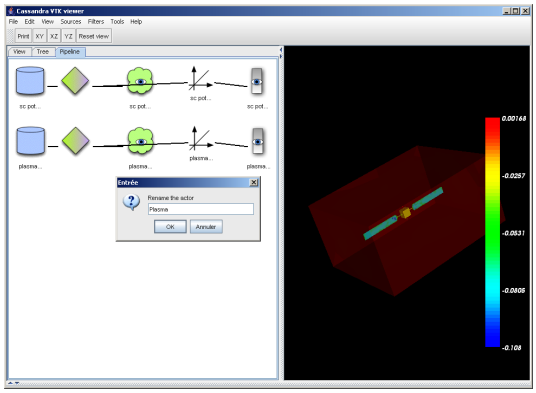
Actor manipulation:

- Making it visible or not.
- Changing opacity.
- Assigning a colour when you unselect the view of scalar data on the mapper.
- Rename the object.
- Remove the object.







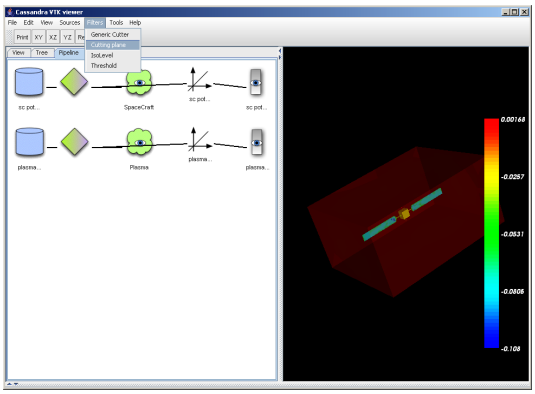


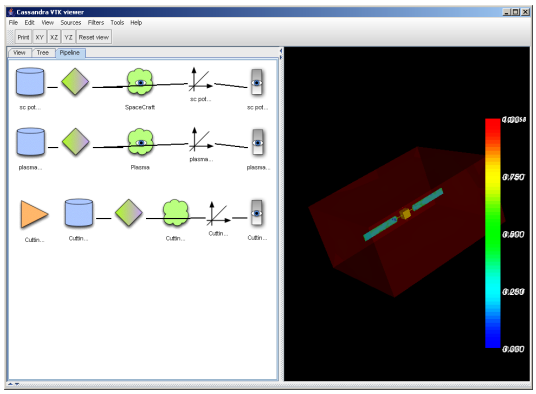
3) Apply a simple filter

Go to the “Filters” menu and select the one you want to add to your VTK pipeline. In our example, we will start with a cutting plane, in order to look at the plasma potential around the satellite along a plane.

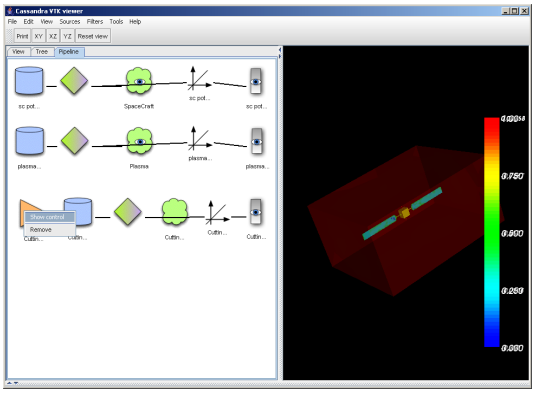
Step to follow

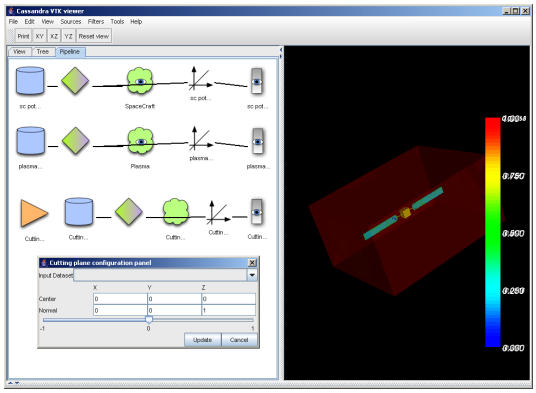
a) Add the filter to the pipeline by selecting it in the Filters menu.

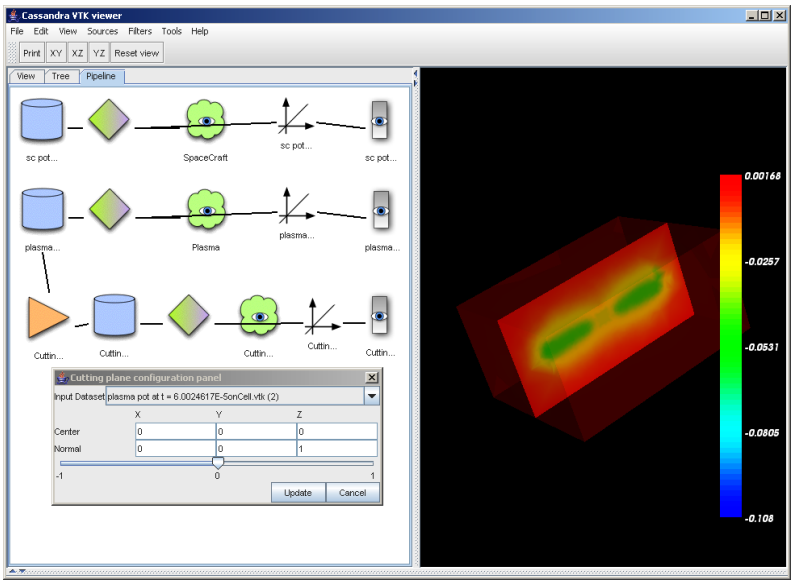




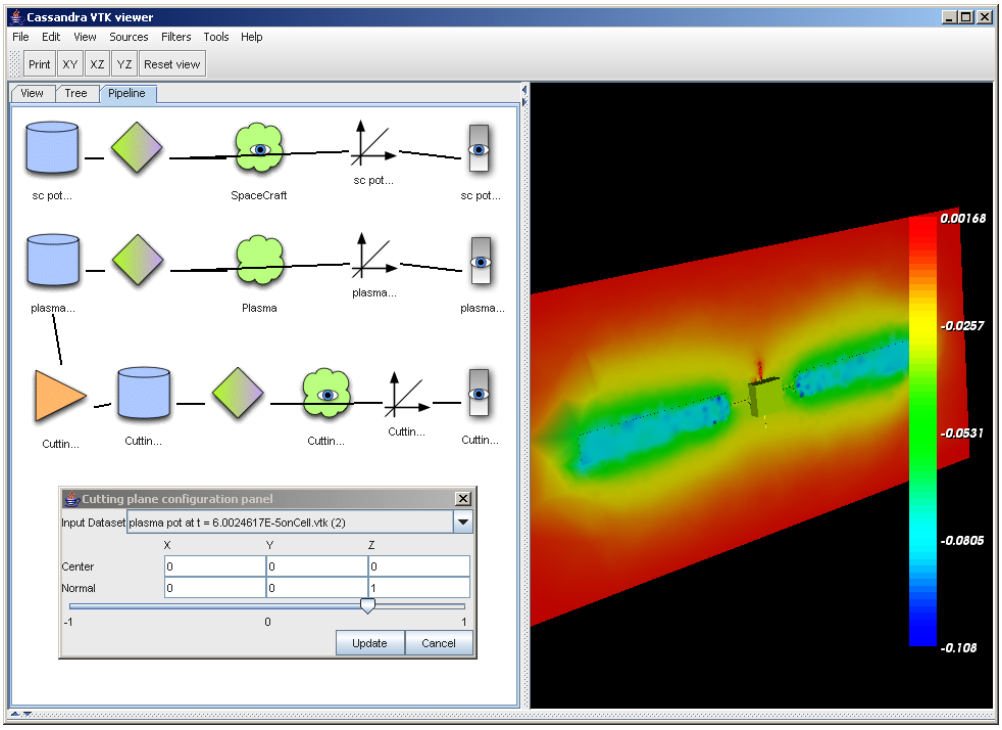
b) Go to the control panel of the filter in order to set the data to process and specify the property of our filter. Since you update your control input, the pipeline will be updated automatically and you will be able to see the flow of the data processing.







c) In the cutting plane filter, you set a point inside the plane and the normal of that plane. The slide bar is used to move slightly around the specified point easily. It enables you to move the plane back and forward around the normal vector as a unit.

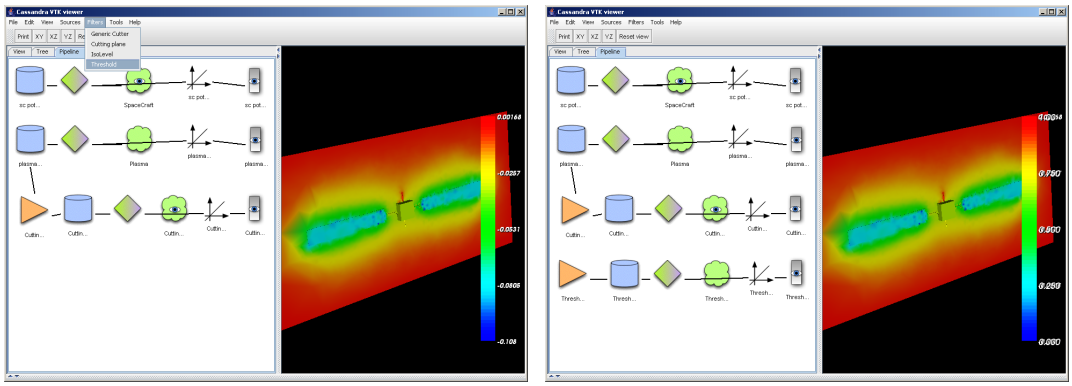


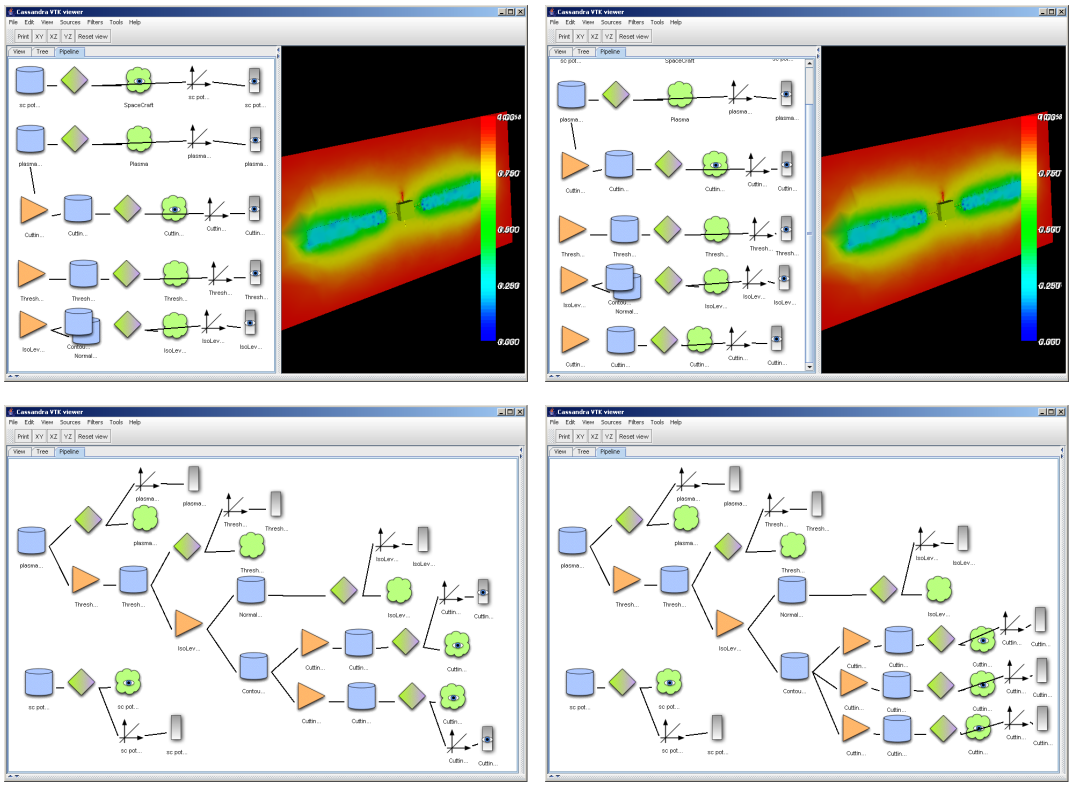
4) Composition of filters

Go to the “Filters” menu and select the one(s) you want to add to your VTK pipeline. In our example, we will add an iso-level on which we will apply several cutting planes in order to see the field lines. On top of that, we will cut the range of the data with a threshold filter.

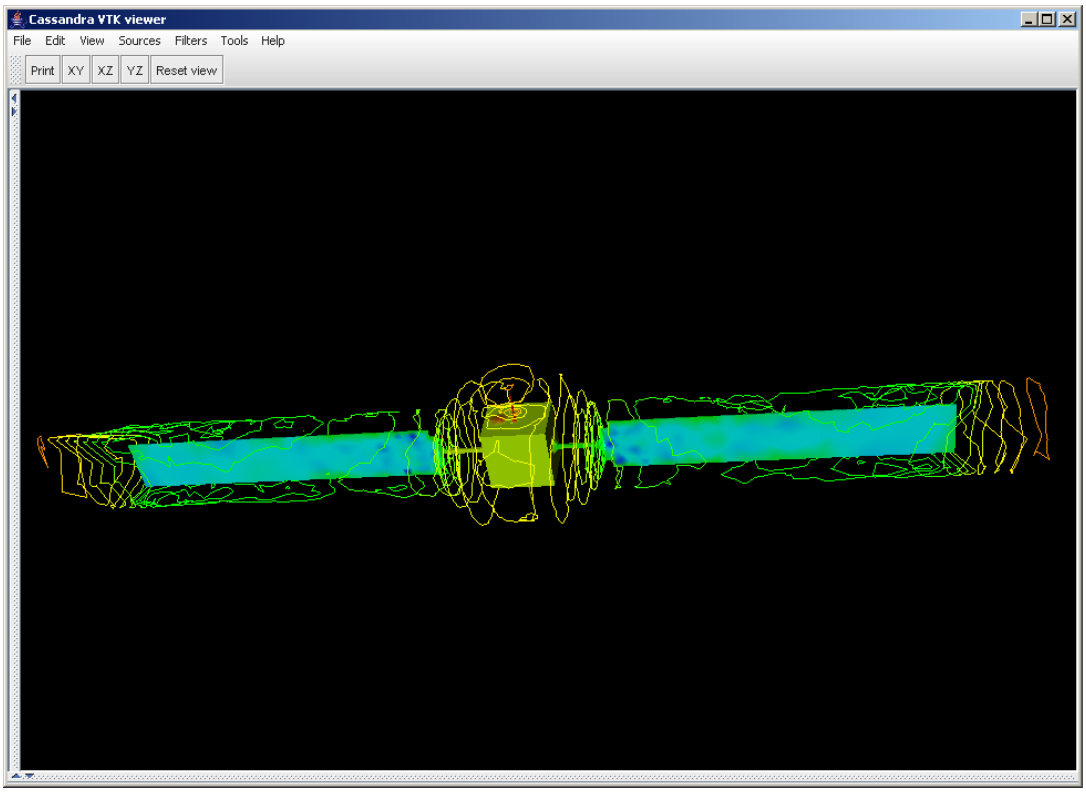
Steps to follow

a) Add the filter to the pipeline by selecting it in the Filters menu. (1x Threshold – 1x Iso-Level – 2x Cutter)





b) Go to the control panel of the filter in order to set the data to process and specify the property of our filter. Since you update your control input, the pipeline will be automatically updated and you will be able to see the flow of the data processing.

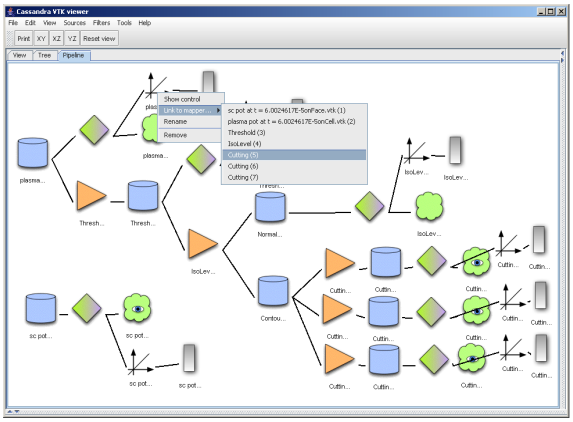


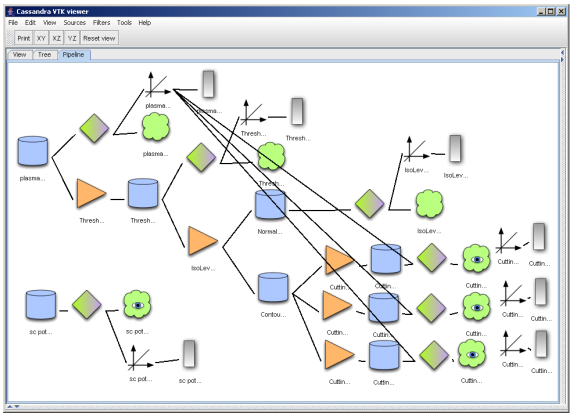
5) Managing several scales and lookup tables

As you must have noticed, everytime a file is loaded or a filter added, a dedicated lookup table and a scalar bar appear.

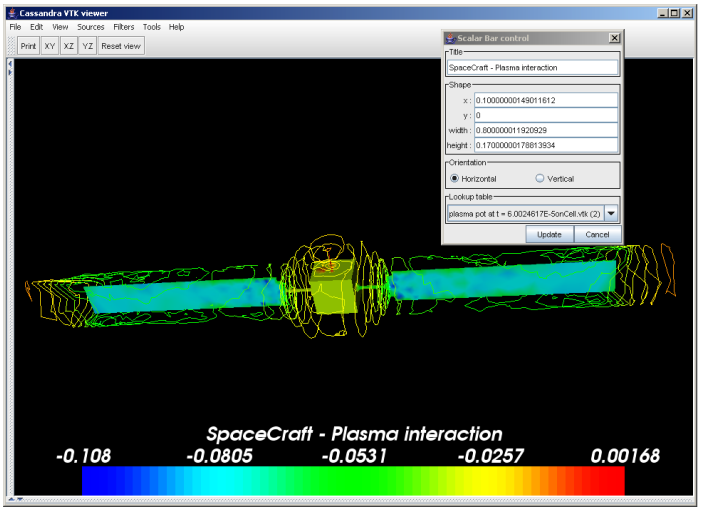
Note:

When you want to compare the same type of data with a unique lookup table and scalar bar, you will have to link that lookup table to the dedicated mappers with the contextual menu of the wanted lookup table.



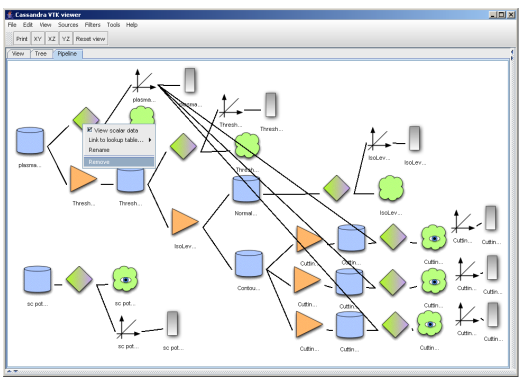
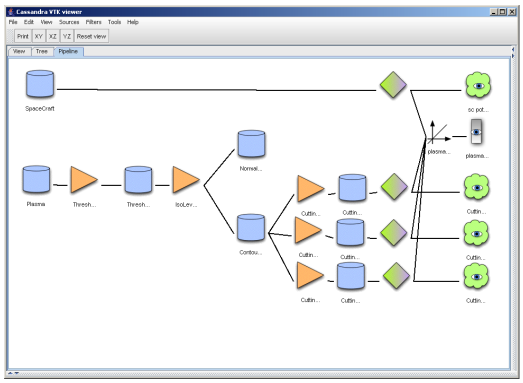


To change the location and the scalar bar information, show its control panel from its contextual menu.



6) Removing components

To simplify the understanding of the pipeline, you can remove parts of the unnecessary items.

7) Saving generated view

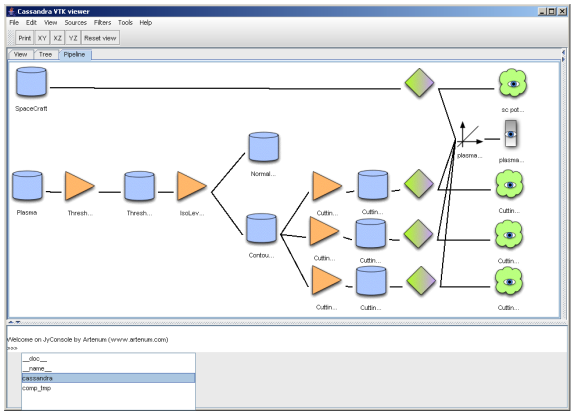
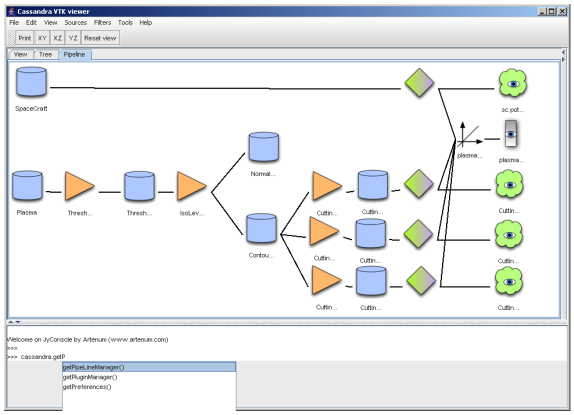
The 3D view can be saved by the “Save as image” button on the toolbar or directly in the file menu by the “Save image” item. This will use the Tiff image format.

8) Using the Python console

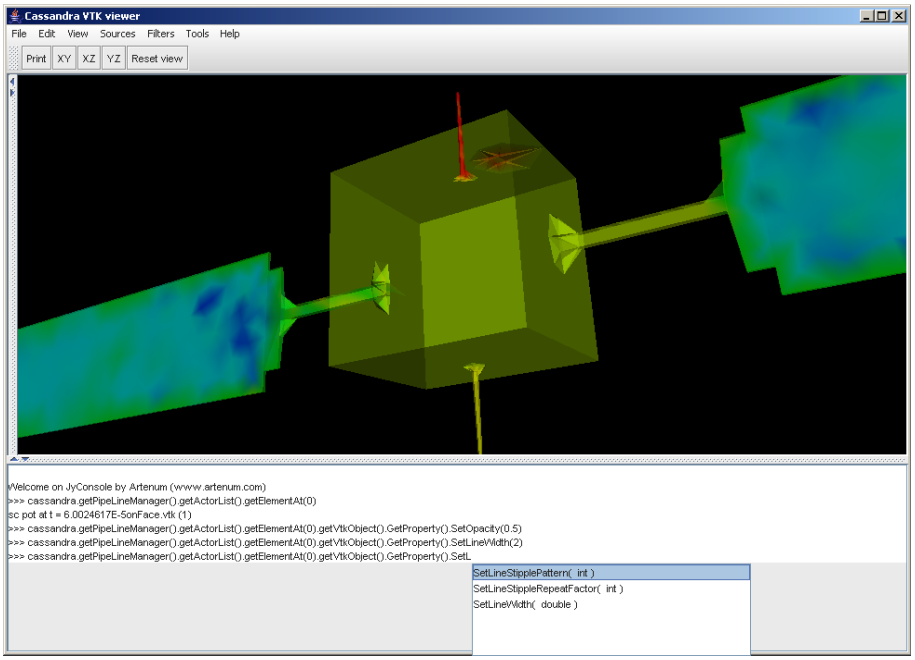
The Jython console integrated into the Cassandra application enables the user to call directly the method on the Java object. Moreover, the completion enables the user to see which operation can be done on that object, in a guided action approach.

9) Completion

Here are some examples: *(To access the completion just press CTRL + SPACE.)*

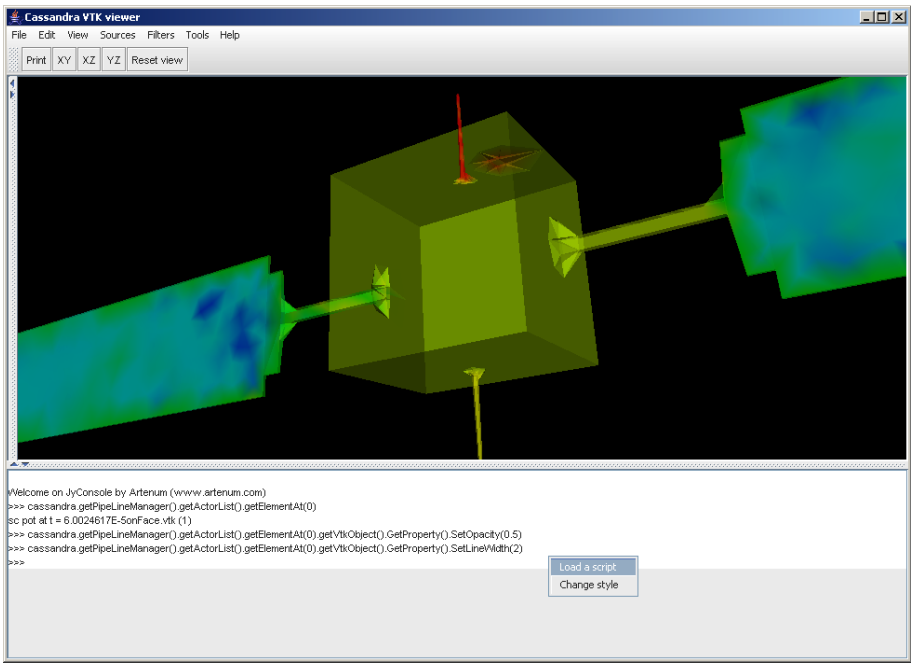



10)Accessing and changing the data



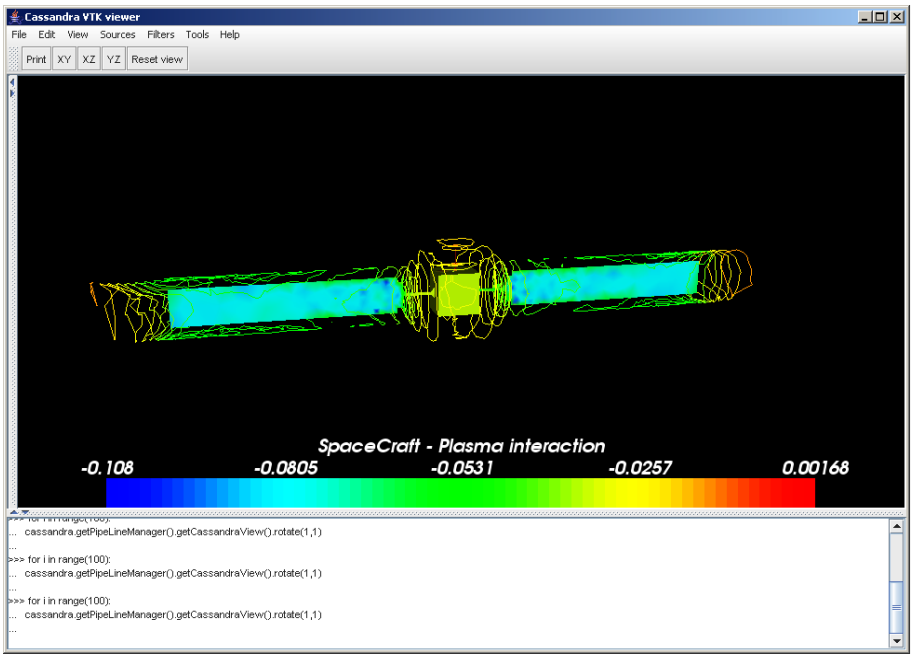
11)Loading an external script

To load a Python script written in an external file, go to the contextual menu of the console by right clicking the console window, then choose to load a script. A file browser will appear in order to simplify your file selection.



12)Making animation

The simplest way to do an animation so far has been to write a simple Python script that will change some input data. You can directly write it in the console or, for more complex animation, you can write an external file that you will be able to load later if you want to reproduce your animation. An example is available in the default Script directory of Cassandra. One file is used to load and set the data, and a second one to manipulate and change the view parameters in order to generate a set of images.



References

[1] Artenum's Web Site, http://www.artenum.com, 2001-2005

[2] Julien Forest, Jean-Francois Roussel, Alain Hilgers, Benoit Thiebault, Sebastien Jourdain, *SPIS-UI, a New Integrated Modelling Environment for Space Applications,* 9th Spacecraft Charging Technology Conference, Tsukuba, Japan, April 2005.