

Memòria: Eina per a la visualització de molècules

Assignatura: TFC-Gràfics per computador. 05 119
Nom: Victor Manuel Esteban Sanchez. <vesteban@uoc.edu>
Data: 10-01-2005

Assignatura: TFC-Gràfics per computador. 05 119

Nom: Victor Manuel Esteban Sanchez. <vesteban@uoc.edu>

Data: 10-01-2005

Memòria: Eina per a la visualització de molècules

Índex

1. Introducció.
2. Objectius i requisits.
3. Requeriments de Software.
4. Pla de Treball.
5. Anàlisi.
6. Captures de pantalles de l'aplicació amb diferents tipus de molècules.
7. Implementacions futures o millores.
8. Identificació del procés de desenvolupament del TFC.
9. Manual d'instal·lació.
10. Descripció de la aplicació, manual d'us.
11. Bibliografia.

1. Introducció

La visualització de molècules es una tasca molt important en moltes àrees com la química, la farmàcia, etc.

Les molècules estan compostes d'àtoms i lligams. En el format de visualització *ball-and-stick* els àtoms es representen com a esferes i els lligams com a cilindres.

Aquest tfc consisteix en construir una eina que permet visualitzar molècules en aquest format de visualització. L'eina haurà de permetre navegar al voltant de la molècula i utilitzar tècniques de selecció que permetin identificar àtoms, calcular distàncies i angles de torsió. A més, s'haurà de poder definir un eix i sobre ell generar una rotació i gravar-la en una pel·lícula (en el format que sigui més senzill).

2. Objectius del treball i requisits.

L'objectiu final és una aplicació que permeti inspeccionar molècules. Per a assolir-lo, caldrà estudiar sobre els següents termes:

- Format de fitxer d'entrada. (On es guarden els símbols i les posicions dels àtoms. També tindrem un altre arxiu d'entrada que conté la informació dels radis covalents dels àtoms i les distàncies que les separen. Necessària per que l'aplicació pugui crear els enllaços.)
- Format de sortida de la pel·lícula.
- Tècniques d'interacció i usabilitat.
- La visualització es realitza sempre en projecció ortogonal.

3. Requeriments Software.

Sistema operatiu Linux/Windows
Java (JDK)
OpenGL
Eina per a la creació d'interfícies d'usuari.

4. Pla de Treball

4.1 Operacions

Inicial ment tenim els apartats.

1. documentació i definició de requeriments.
2. implementació de la aplicació.
3. elaboració de la memòria.
4. elaboració de la presentació virtual.

Relació d'operacions.

- a) preparació sobre la documentació per conèixer la geometria necessària sobre molècules.
- b) preparació de documentació necessària sobre opengl i java.

- c) identificació de les necessitats de software i llibreries/ gràfiques.
- d) preparació de documentació necessària sobre la captura de imatge de la càmera i del format de pel·lícula en la que s'emmagatzemarà el gir de la molècula.

4.2. Implementació de l'aplicació.

a) Lectura de la informació de la molècula.

- a.1) definició i disseny del format de fitxer d'entrada.
- a.2) definició i disseny del TAD que contindrà la definició de la molècula.
- a.3) definició de la part del interface d'usuari relatiu a la lectura de la molècula.

b) Modelatge de la molècula i representació visual.

b.1) definició i disseny de les classes i objectes de forma que permetin treballar amb una escena que continguin una taula d'objectes 3D.

b.2) implementació del TAD que contindrà la representació virtual.

b.3) dintre de la classe que defineixi un sòlid, creació d'un constructor d'objecte esfera i un constructor d'un paral·lelepípede que serveixi com a nexa d'unió entre els diferents àtoms.

b.4) Creació d'una classe que defineixi les línies de cota que permetin fer la definició gràfica de les distàncies i angles de torsió.

c) Creació de l'interfície d'usuari per a permetre realitzar la funcionalitat de moure la molècula dintre dels tres eixos, la identificació i la selecció de àtoms.

- c.1 definició i disseny de la interfície d'usuari.
- c.2 incorporació al TAD de definició de la molècula la capacitat de contenir la selecció feta del àtoms que es desitja calcular distàncies i angles de torsió.
- c.3 implementació de la interfície d'usuari.

d) Funcionalitats de càlcul de distàncies, angles de torsió i posicionament de l'eix per poder realitzar una rotació.

d.1 disseny i implementació de les classes per realitzar el càlcul indicat.

e) definició del model d'il·luminació per a la representació de l'escena. Posicionament dels focus d'il·luminació en relació al posicionament de la molècula.

f) definició i disseny del format de la pel·lícula. Implementació de la captura de les imatges i de la generació del arxiu de sortida corresponent a aquest pel·lícula.

4.2 Elaboració de la memòria

Presentació del treball, identificació dels objectius proposats, aspectes formals. Incorporació de la definició i disseny de la aplicació i documentació de les funcionalitats. Identificació de la bibliografia emprada i de les contribucions personals.

4.3 Elaboració presentació virtual.

Presentació en forma de transparències virtual de una síntesis de la aplicació.

5.0 Anàlisis

La denominació de la aplicació es **vmol**, que està constituïda per 3 paquets, el paquet central que conté tota la informació de la molècula, **vmol**, un paquet destinat a fer l'anàlisi xml del arxiu d'entrada, **vmol.xml**, i un paquet amb tota la informació gràfica per poder fer la visualització, **vmol.gl**

La representació de la molècula es realitzarà emprant una esfera aproximada per polígons plans. La unió entre àtoms per a representar les connexions es realitzarà emprant un cilindre aproximat igualment per polígons plans.

5.1 Guions

Guió de l'usuari

El usuari inicia la aplicació. La aplicació comprova si té un arxiu de configuració, si el té, el llegeix i mira com es diu el darrer arxiu de molècula llegit. Si existeix llegeix la molècula i la mostra. En cas contrari espera a rebre instruccions del usuari.

Si no hi ha cap molècula llegida la aplicació solament haurà de mostrar la opció de seleccionar arxiu de molècula.

Una vegada llegida i visualitzada l'usuari pot fer:

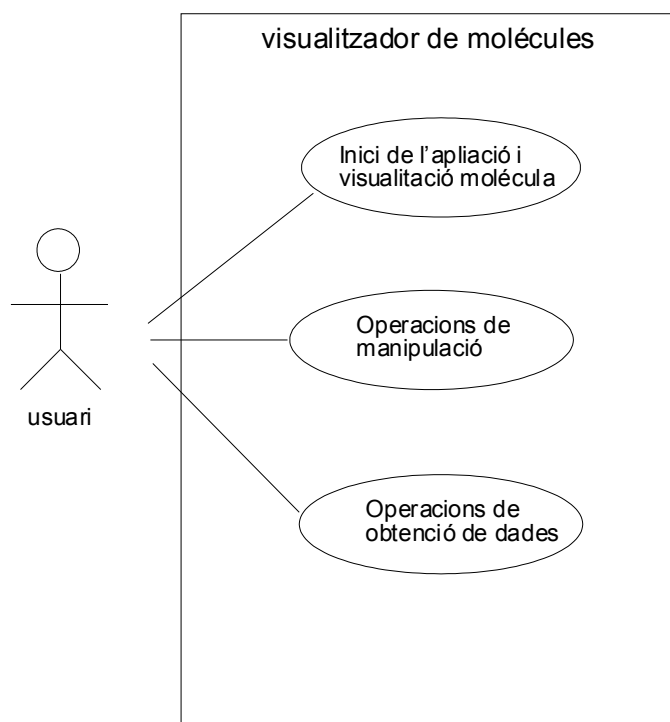
- zoom per apropar allunyar la molècula
- fer un gir transversal o longitudinal de la molècula.
- Seleccionar àtoms individualment
- mostrar la barra d'utilitats

Una vegada la barra d'utilitats es visible l'usuari té la capacitat de fer les següents accions:

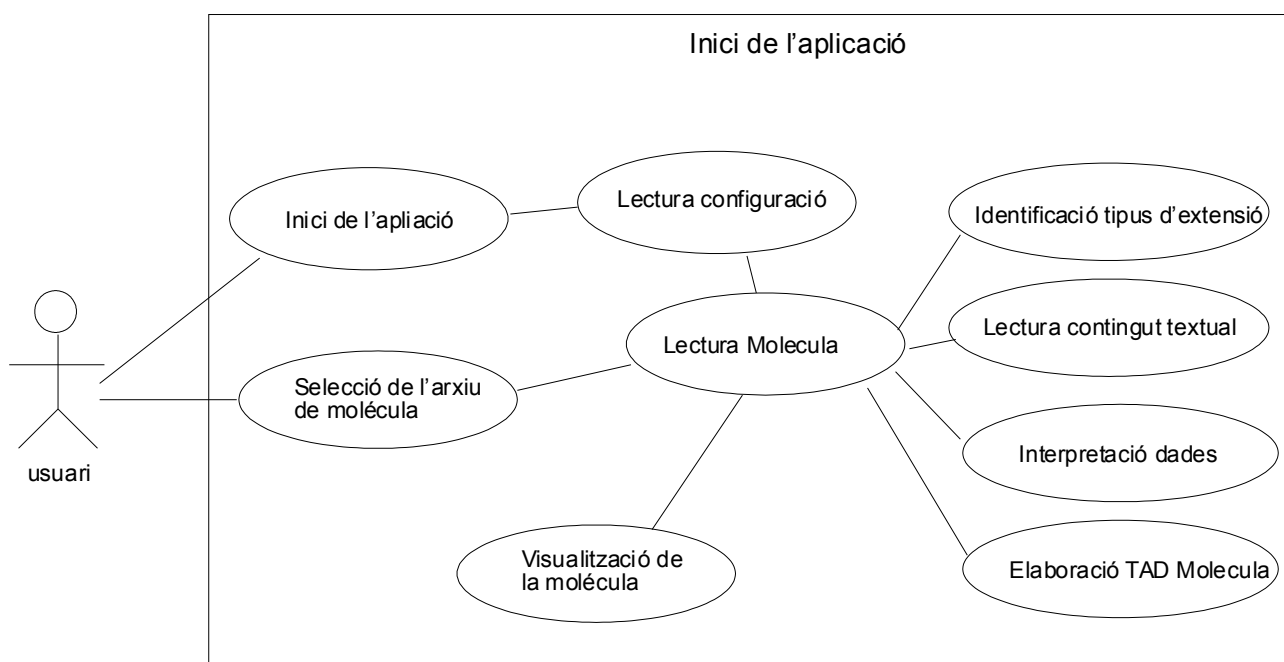
- seleccionar en la barra d'utilitats la opció de visualitzar la definició d'un eix de rotació.
- seleccionar l'acció d'iniciar la rotació, en el cas de que no hagi definit prèviament un eix de rotació, l'eix escollit per defecte es el vector 0,1,0
- seleccionar l'acció de exportar la rotació.
- seleccionar l'acció de mostrar distància, prèviament l'usuari ha d'haver seleccionat 2 àtoms. A la mateixa barra es mostrarà el valor numèric resultat de l'execució.
- Seleccionar l'acció de mostrar angle, prèviament l'usuari ha d'haver seleccionat 3 àtoms. A la mateixa barra es mostrarà el valor numèric resultat de l'execució.

5.2 Casos de ús

Cas d'us **visualitzador de molècules.**

**Precondicions:**

- es necessari disposar d'arxius de molècules en format .xyz ó .xml per poder realitzar la visualització dels mateixos.
- es necessari tenir instal·lades les eines de software indicades a l'apartat requeriments.

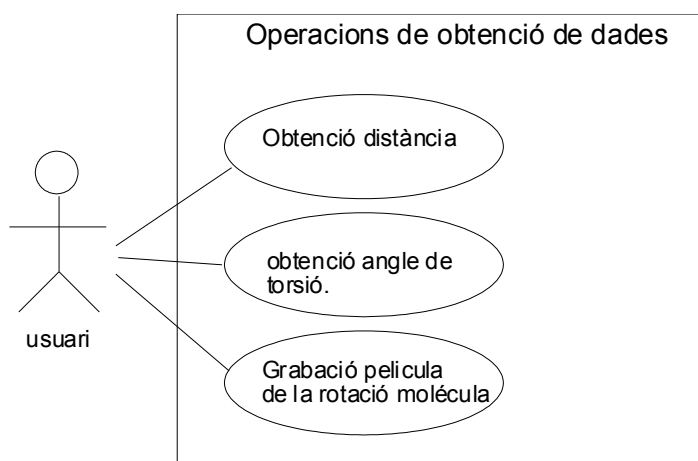
Cas d'us inici de l'aplicació

Precondicions

- La lectura de configuració es fa prèviament sobre l'arxiu de nom vmol.conf.tmp, si no existeix no llegeix cap molècula, si existeix, llegeix el seu contingut i inicia el procediment de visualització.
- Es necessari tenir permisos d'escriptura sobre la carpeta on es troba l'aplicació.

Post condicions

- La molècula es mostra amb una càmera i un punt de observador adequats per poder observar completament la molècula.

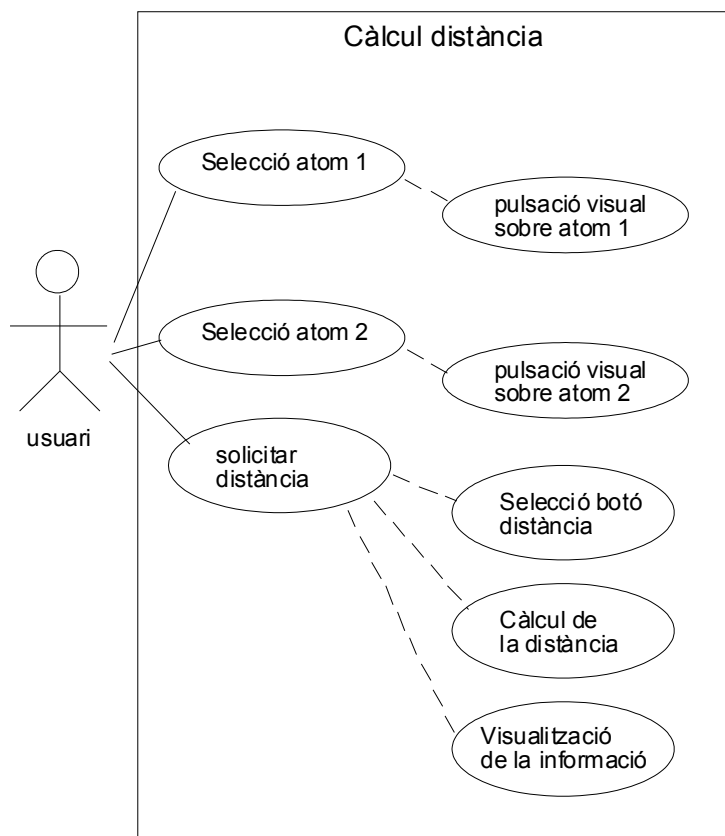
Cas d'us operacions d'obtenció de dades**Precondicions**

- Hi hem de tenir una molècula visualitzada.
- No es podem seleccionar més àtoms que els necessaris i suficients per donar l'operació demanada.

Post condicions

- Hi hem de tenir identificats visualment els àtoms seleccionats. Els àtoms seleccionats apareixeran de color blanc.

Cas d'us operacions de obtenció de la distància entre 2 àtoms.



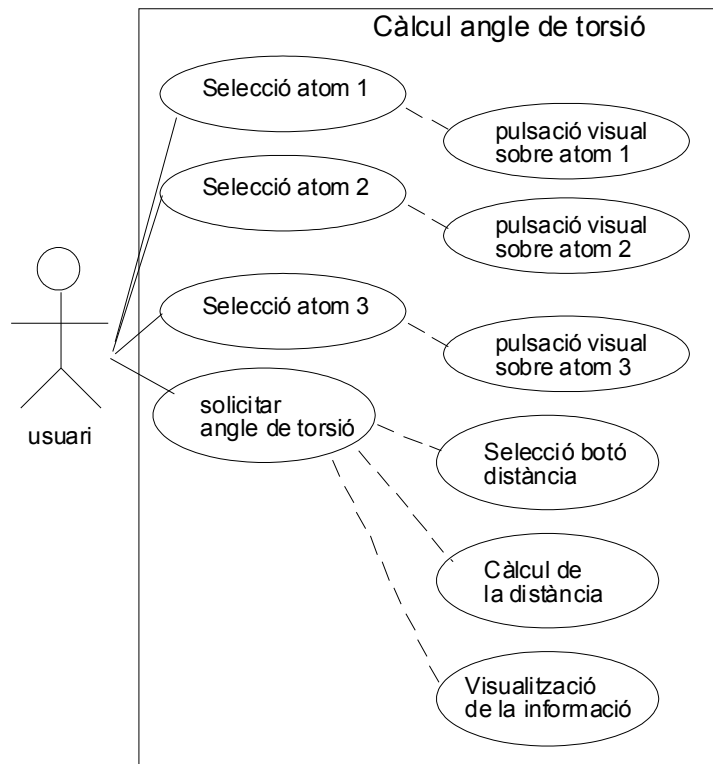
Precondicions

- Hem de disposar d'una molècula visualitzada per poder fer les operacions de manipulació.
- Es necessari que l'àtom a seleccionar estigui visible dins l'àrea de visualització. La selecció es realitzarà emprant el mouse i clicant sobre la superfície de l'àtom.

Post condicions.

- La següent visualització està d'acord amb l'operació sol.licitada.

cas d'us **obtenció angle de torsió**



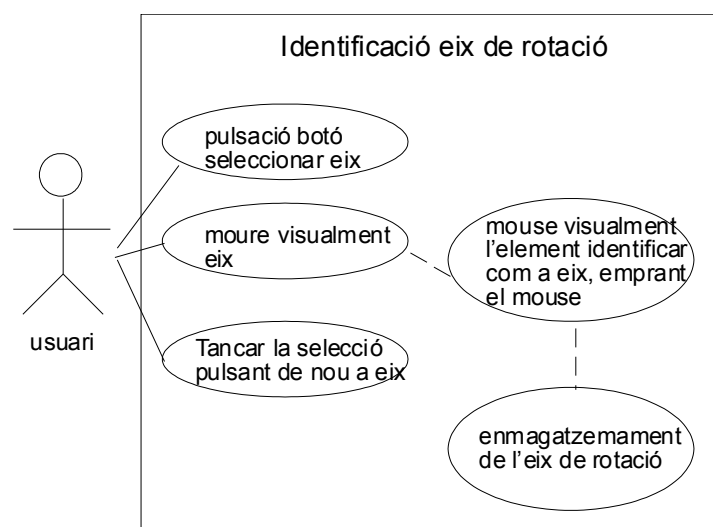
Precondicions

- Hem de disposar d'una molècula visualitzada per poder fer les operacions de manipulació.
- Es necessari que l'àtom a seleccionar estigui visible dins l'àrea de visualització. La selecció es realitzarà emprant el mouse i clicant sobre la superfície de l'àtom.

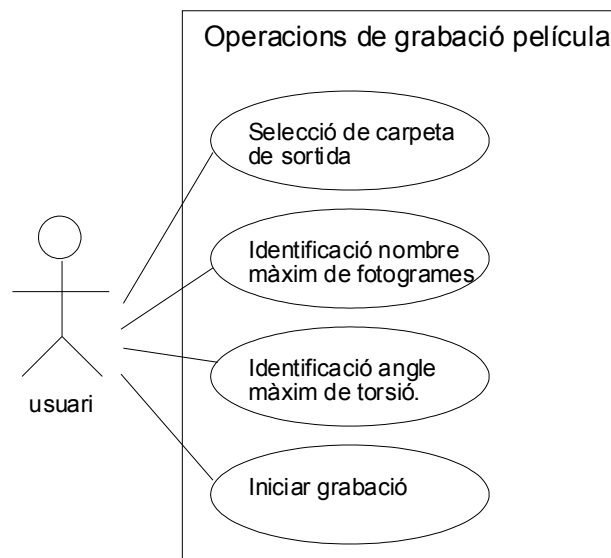
Post condicions.

- La següent visualització està d'acord amb l'operació sol.licitada.

Cas d'us **identificació eix de rotació**



Cas d'us **gravació pel.lícula**



Precondicions

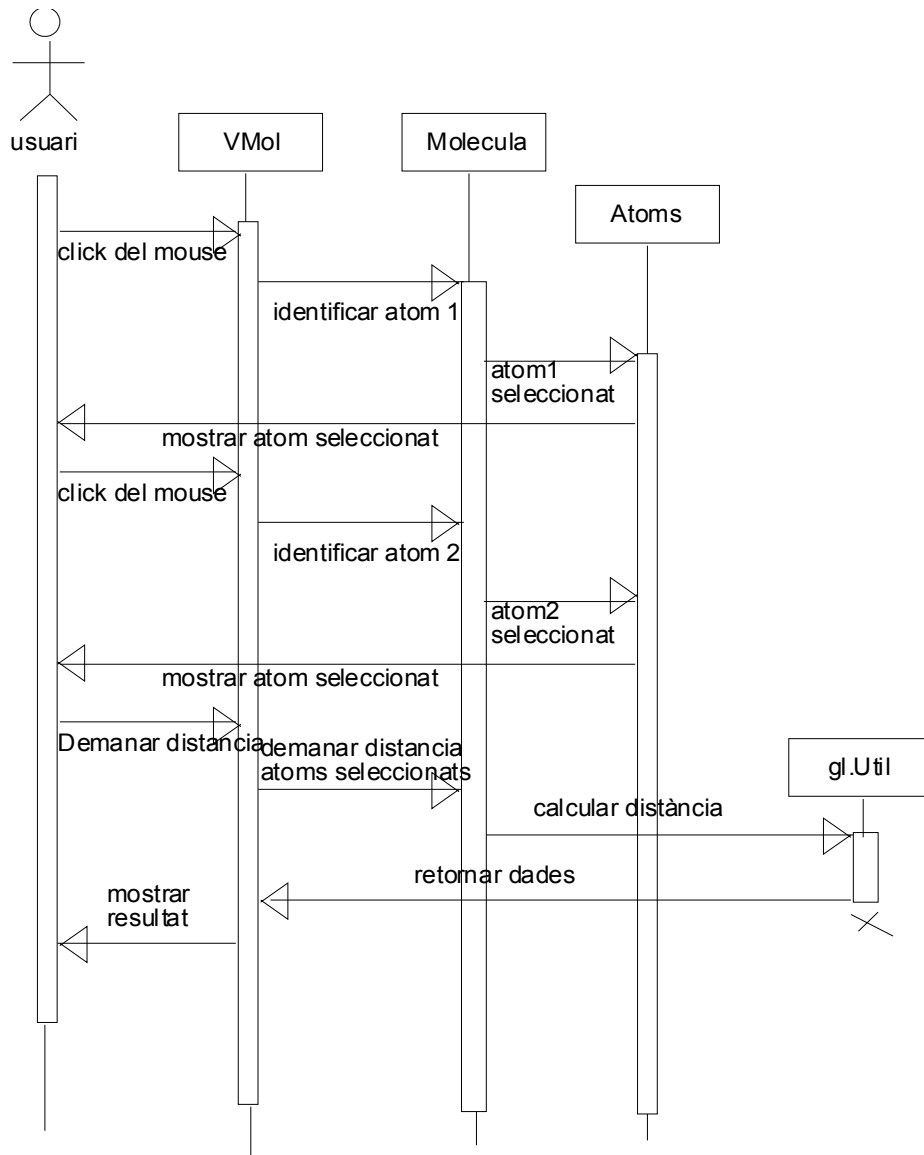
- Es necessari tenir visualitzada la molècula.
- Es necessari tenir identificat l'eix de rotació

Post condicions

- Hem de tenir a la carpeta seleccionada la seqüència d'imatges que corresponguin a la rotació de la molècula.

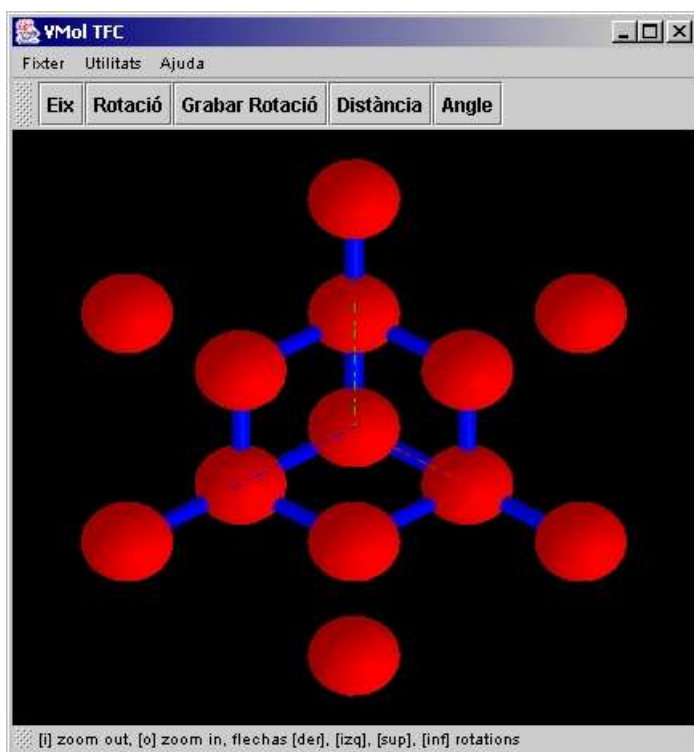
5.3 Diagrama de seqüències

Diagrama de seqüències del cas d'us obtenció de dades de la distància que separa dos àtoms.



5.4 Interfície d'usuari

1. Interfície principal, on es mostren les eines de selecció i navegació.



A la següent imatge es poden apreciar les opcions de menú disponibles.



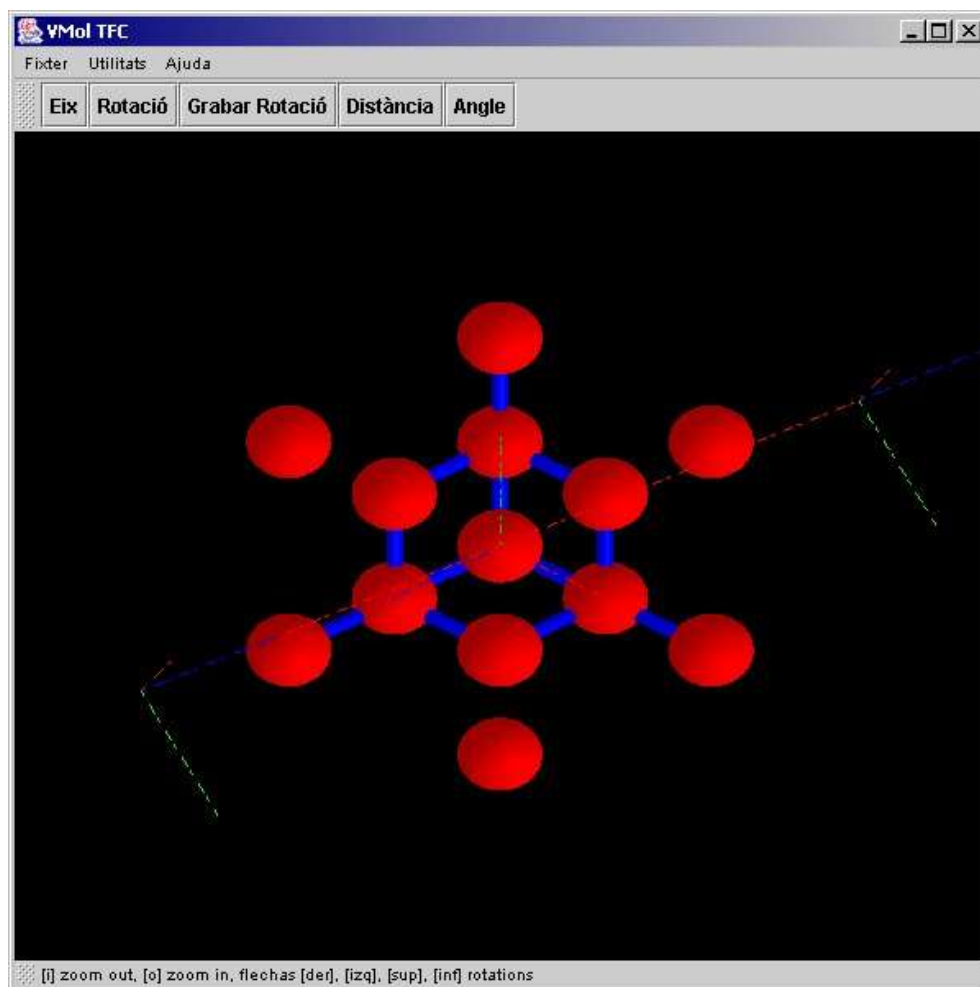
A l'opció Fitxer podem seleccionar la molècula a visualitzar. La barra d'utilitats ens mostra les operacions que podem realitzar, seleccionar un eix de rotació, rotar la molècula entorn aquest eix, gravar la rotació a un fitxer, identificar la distància entre 2 àtoms o identificar l'angle de torsió entre 3 àtoms.

Es possible tancar o mostrar la barra d'utilitats per poder tenir més espai de visualització de la molècula. Amb l'opció utilitats.

A la part dreta apareixen, dins la barra d'utilitats, apareix la informació sobre les dades que demanen de la molècula.

També es possible redimensionar a pantalla completa l'aplicació.

2. Interfície de selecció d'eix de rotació



La identificació es realitza visual e interactivament mitjançant el mouse. L'eix dibuixat es mou indicant l'eix de rotació de la molècula.

Quan es selecciona el botó eix es mostra una representació d'un eix de rotació, el qual el podem moure interactivament amb l'ajuda del mouse. A mesura que desplaçem el mouse dins l'àrea de visualització de la molècula, l'eix rotarà entorn els eixos xyz.

3. Interfície de exportació de pel·lícula

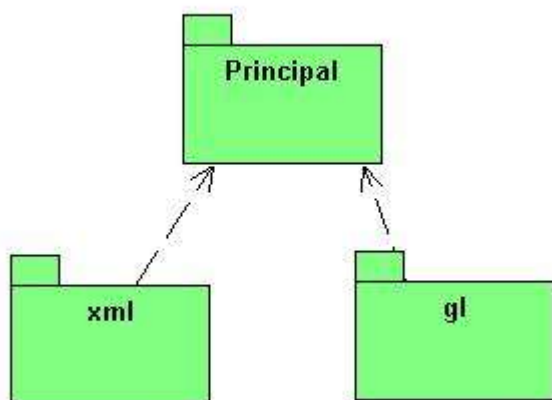


La primera operació necessària es identificar la carpeta on s'emmagatzemaran les imatges,

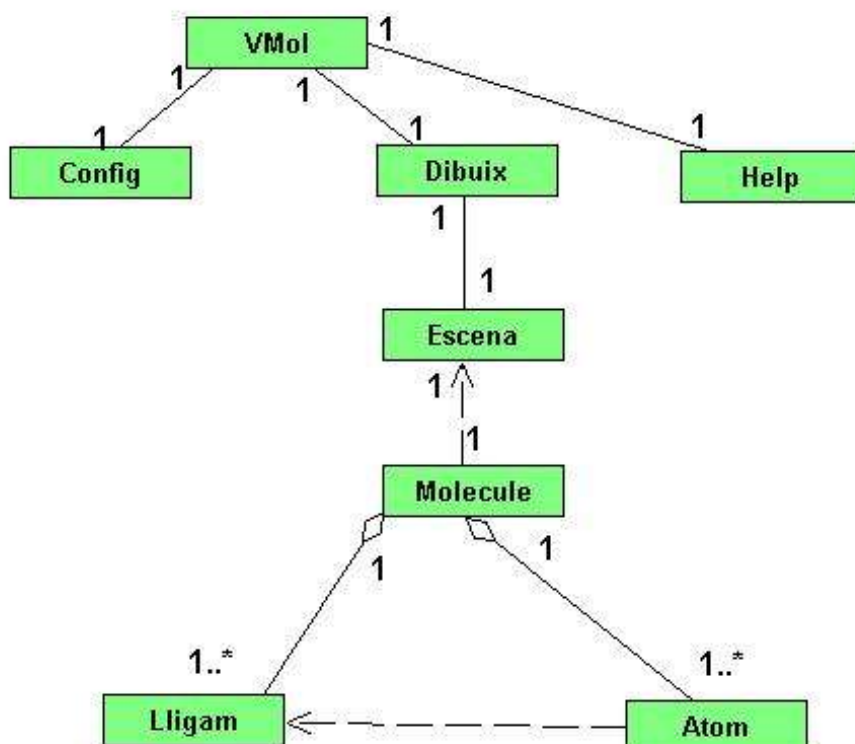
després modificar les valors de nombre de fotogrames i/o angle de rotació a capturar. Es polsa el botó iniciar i comença la gravació de la rotació.

5.5 classes i estructura de dades

La representació del diagrama estàtic de la aplicació es aquest. Fa servir 2 paquets interns a la aplicació, el gl amb eines gràfiques i xml per poder processar el format d'arxiu basat en xml.



5.5.1 El diagrama estàtic de la aplicació principal

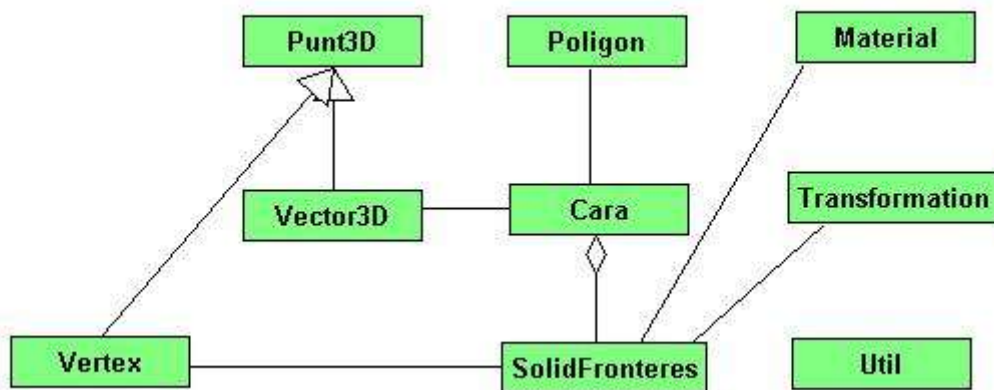


Vmol es la classe principal i fa servir les classes **Config**, que te la responsabilitat de gestionar la configuració i llegir les dades, **Dibaux** on es mostra visualment la molècula i **Help** que mostra un quadre de diàleg amb informació sobre la navegació.

A partir de **Dibuix** es defineixen **l'escena** que conté la informació i els mètodes necessaris per a fer la visualització dins de Dibuix. Entre totes aquestes classes la cardinalitat es sempre 1 a 1.

Després, dins de molècules tenim contenides els àtoms i els lligams. La classe Molecule conforma la definició (TAD) de la molècula.

5.5.2 Diagrama estàtic del paquet gl



A mode de llibreria gràfica he disgregat aquest element de l'aplicació principal en el paquet **gl** per que contenen funcionalitat genèrica o comú a qualsevol aplicació de visualització de contingut tridimensional. El paquet gl té com a objectiu emmagatzemar la informació geomètrica de l'escena.

Punt3D, defineix un punt dins l'espai vectorial x,y,z .

Vector3D, defineix un vector dins l'espai.

Vertex, es un punt que correspon a un vèrtex d'un objecte sòlid.

Polígon, defineix un polígon dins un pla.

Cara, defineix un polígon que correspon a una cara d'un objecte sòlid.

Material, defineix un determinat material segons els conceptes de OpenGL.

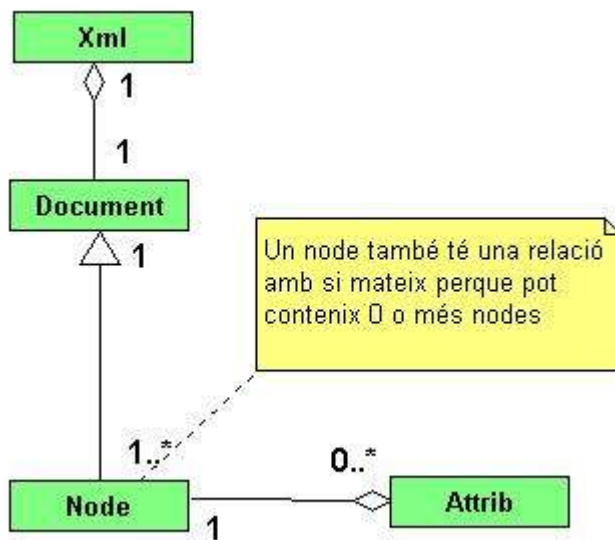
Transformation, defineix totes les operacions de translació i rotacions necessàries que es poden aplicar als elements.

SolidFronteres, defineix un objecte sòlid.

Util, conté totes les operacions comuns de càlcul vectorial necessàries:

- càlcul del mòdul de un vector
- multiplicació escalar de 2 vectors.
- multiplicació vectorial de 2 vectors.
- Càlcul del vector unitari
- distància entre 2 punts
- angle entre 2 vectors

5.5.3 Diagrama estàtic del paquet xml



Xml, conforma tota la definició (TAD) d'un objecte o document XML. El document XML conté una descripció de la seva codificació i el contingut (classe Document)

Document, classe que conté o de la que parteix tot l'arbre del document XML. Un document pot contenir 1 o més nodes.

Node, classe que conté un element d'un objecte xml, cada node pot contenir a sí mateix també un arbre de nodes o ser un node final, sense cap més node. Cada node conté una sèrie de atributs, parells de clau,valor.

Attrib, classe que conté la definició d'un atribut que correspongui a un determinat node, parell clau,valor.

5.5.4 Descripció de les classes del paquet principal

Vmol es la classe principal de la aplicació.

Atributs: desk,status,chooser son elements visuals que fan servir més d'un dels mètodes de la classe. config es una instància de la classe Config que conté les operacions necessàries per recollir la informació dels arxius de molècula. dib es una instància de Dibuir que realitza les operacions de visualització, eix es una instància de la classe Eix que conté la informació de l'eix de rotació escollit per l'usuari. is_eix_definit ens informa si l'usuari ja ha escollit un eix. molecule es una instància de la classe Molecule amb la informació de la molècula.

Mètodes: Al constructor Vmol() es declara tota la part visual de l'aplicació. Amb el mètode createMenuBar declarem i creem el menú d'opcions. setCenter() posiciona l'aplicació al centre de la pantalla. LoadFileOrig() mostra un quadre de diàleg de selecció del fitxer de molècula i actualitza l'atribut de classe forigen amb la informació del fitxer a llegir. readConf() Llegeix l'ultima selecció de arxius feta, per a millorar l'ús de l'aplicació. WriteConf() Escriu l'ultima selecció d'arxius feta. showMsg() mostra un quadre de diàleg amb un missatge. dibuixar() Fa la lectura de l'arxiu seleccionat i implementa l'acció de dibuixar la molècula visualment. Els tipus d'arxius reconeguts son .xyz y .xml

VMol
-desk:JPanel -status:JLabel -forigen:File -font:Font -config:Config -chooser:JFileChooser -dib:Dibuir -eix:Eix -is_eix_definit:boolean -molecule:Molecule -jinfo:JLabel
+VMol() +init() -createMenuBar() -setCenter() -loadFileOrig() -readConf() -writeConf() -showMsg(title:string, msg:string) -showInfo(title:string, msg:string) -dibuixar()

La classe **Config** fa les operacions de llegir les dades necessàries dels arxius de molècula i de identificació dels àtoms

La configuració de les dades dels àtoms està codificat dins l'aplicació, a la classe config, però també es possible dins l'arquitectura desenvolupada la possibilitat de llegir-la d'un arxiu extern (mètode readConfig()). Aquest arxiu està en format XML donat també dins l'especificació de formats d'arxiu. En aquest cas es llegiria la configuració i es posaria dins l'atribut xml_config. Una vegada processat s'emmagatzema al TAD àtoms.

Atributs: configfile, moleculefile, contenen els noms dels arxius de informació sobre àtoms i de informació de la molècula respectivament. content_scr_configfile i content_src_moleculefile, els continguts dels arxius. xml_config i xml_molècula el contingut però coma objecte XML. molecule es el TAD que contindrà la definició de la molècula. atoms conté les dades de configuració dels àtoms, tipus, pes, radi, color, etc.

Mètodes: setConfigFile () assigna el nom de l'arxiu de configuració. SetMoleculaFile () assigna el nom de l'arxiu de molècula. getMolecule retorna una instància de la classe Molecule amb la informació de la molècula. Els arxius d'entrada poden tenir el format **xml** especificat en aquest document, o el format de tipus **xyz**. Els mètodes setConfigfile, i setMoleculefile assignant els noms dels arxius a llegir, de configuració i de les dades de la molècula respectivament. Els mètodes readMolecule() i readMoleculeXYZ() realitzen l'acció de la lectura i la incorporació de les dades al TAD Molecule de la molècula. El mètode getMolecule() retorna una instància de la classe Molecule amb tota la informació necessària de la molècula. readConfig() Llegeix l'arxiu de configuració i emplena el vector àtoms amb la informació dels àtoms. Aquest mètode existeix com a opció alternativa. Inicialment l'aplicació ja inclou les dades necessàries per a visualitzar les molècules amb el mètode setAtoms, inclòs en aquesta mateixa classe. readMolecule() mètode que llegeix un arxiu amb format XML i complimenta el TAD de molècula. Aquest mètode es alternatiu a la funcionalitat de la aplicació. setAtoms () Assigna directament la informació dels àtoms. Dades recollides d'arxiu atosizes rasMol. readMoleculeXYZ () llegeix la informació de la molècula des de un arxiu de tipus .xyz

Config
-configfile:string -moleculfile:string -content_src_configfile:string -content_src_moleculfile:string -xml_config:XML -xml_molecul:XML -atoms:Hashtable -molecul:Molecule
+setConfigFile (configfile:string):void +setMoleculFile (moleculfile:string):void +getMolecule ():Molecule +readConfig():void +readMolecul() +setAtoms() +readMoleculXYZ() +toString():string

La classe **Molecule** conté tota la informació necessària de la molècula per poder-la representar gràficament. També realitza les accions de dibuixar-la visualment.

Atributs: atoms es un Vector que conté tota la relació d'àtoms de la molècula, connexions fa el mateix amb els lligams, name es el nom de la molècula, volume conté el volum, panel es un enllaç a l'entorn de visualització, cen el centre de la molècula (i també el centre de l'escena), obs es la posició on es troba l'observador, pini i pfin defineixen la capsa contenidora on es troba situada la molècula.

Mètodes: setPanel guarda una referència a l'entorn de visualització. get_down() i get_up() retornen els punts que defineixen la capsa contenidora, get_centre () i get_observador() retornen el centre i la posició del observador respectivament. add() afegeix un àtom en el procés de lectura del arxiu.

Els mètodes draw (alpha, tetha, mode) i draw (angle, eix) fan l'operació de visualització o dibuixat de la molècula. El mètode setConexions estableix les connexions entre àtoms usant el criteri de que dos àtoms estan connectats si la suma dels seus radis es major que la distància que els separa.

El mètode set_capsa_contenidora() assigna els valors als atributs de classe pini, pfin que defineixen la capsa contenidora. Els mètodes centre() i observador() estableixen el punt central o centre de l'escena i el punt on es trobarà l'observador respectivament, per poder facilitar la visualització de la molècula.

getNumSelectedAtoms() retorna el nombre d'àtoms seleccionats amb l'objectiu de comprovar quan es demana la distància o l'angle de torsió que es trobin seleccionats el nombre correcte d'àtoms. getSelectedAtoms() retorna la relació d'àtoms seleccionats.

Molecule
-atoms:Vector -conexions:Vector -name:string -volume:double -panel:GLJPanel -cen:Punt3D -obs:Punt3D -pini:Punt3D -pfin:Punt3D
+Molecule (name:string, volume:double) +setPanel (GLJPanel) +get_down():Punt3D +get_up():Punt3D +get_centre():Punt3D +get_observador();Punt3D +add(a:Atom) +init() +selectAtom(index:i) +getNumSelectedAtoms():int +getSelectedAtoms():Vector +draw(alpha:float,tetha:float) +draw(ang:double,eix:Vector3D) -setConexions() -set_capsa_contenedora() -centre():Punt3D -observador():Punt3D

La classe **Atom** defineix un àtom dins el TAD molécula. L'àtom es dibuixa per mitjà d'una esfera aproximada per polígons plans amb l'objectiu de optimitzar la visualització.

Atributs: Com a atributs conté tots els necessaris per definir el àtom: símbol, nom, volum, radi, i la posició amb x,y,z. L'atribut selected, de tipus boolean, indica si aquest àtom ha estat seleccionat o no.

Mètodes: En el moment de dibuixar cada àtom es crida a getMinRadi(), que dona un radi de visualització menor al radi real del àtom, d'aquesta manera podem establir la relació de connexió entre àtoms mitjançant un cilindre. Si no fos així, l'establiment no es podria veure.

setPosition () assigna la posició del àtom. Els mètodes setId(), getId() permeten identificar de manera unívoca l'àtom dins la molécula i poder establir la selecció. SetSelected () marca l'àtom com a seleccionat quan clickem dins l'entorn de visualització. getVolum() ens retorna el volum.

Atom
-simbol:String -nom:String -volum:double -radi:double -color:String -id:Integer -selected:boolean
+Atom(simbol:String,volum:nom,radi:double,color:String) +setPosition(x:double, y:double, z:double) +setId(n:int) +getId():int +setSelected(s:boolean) +getSelected():boolean +getSimbol():string +getNom():string +getVolum():double +getRadi():double +getMinRadi():double +getColor():string +getX():string +getY():string +getZ():string +clone():Atom +equals(atom:Atom):boolean

La classe **Lligam** defineix la connexió entre 2 àtoms determinats dins el TAD molècula. El lligam es dibuixa mitjançant un cilindre, però fem l'aproximació mitjançant un paral·lelepípede amb l'objectiu d'optimitzar la visualització.

Atributs: Conté com a atributs els 2 àtoms a través dels quals defineix la connexió i la distància que els separa.

Mètodes: Amb els mètodes **draw** (alpha,theta), i **draw** (angle, eix) realitza la visualització de la connexió.

Lligam
- a1:Atom - a2:Atom -distancia: double
+Conecta(a1:Atom, a2:Atom) +setPanel(panel:GLJPanel) +setCentre(p:Punt3D) +getDistancia():double +getA1():Atom +getA2():Atom +draw(alpha:float,theta:float) +draw(ang:double,eix:Vector3D)

La classe **Escena** conté tota la informació necessària per a visualitzar una molècula i tots els elements accessoris addicionals.

Atributs: centre_escena, el centre de l'escena i la molècula, observador, la posició del observador, u, el vector perpendicular del observador. OC, un vector que està format pels punts centre de l'escena i posició de l'observador, molecule una instància de la classe Molecule i que conté la informació de la molècula, is_set_molecule identifica si hem seleccionat una molècula, panel conté una referència als elements de visualització.

Mètodes: setMolecule() assigna una referència a l'objecte molecule que conté la informació de

la molécula. `getMolecule()` retorna una referència al objecte molecule, `get_down()` i `get_up()` retornen els punts que defineixen la capsula contenidora on es troba la molécula. `getCentre()`, `getObservador()` retornen els punts del centre de l'escena i la posició del observador respectivament. `getU()` retorna el vector vertical. `draw_molècula()` fa l'acció de dibuixar la molécula dins l'entorn gràfic. `draw_axis()` dibuixa un eix de referència.

Escena
-centre_escena:Punt3D -observador:Punt3D -u:Vector3D -OC:Vector3D -molecule:Molecule -is_set_molecule:boolean=false -panel:GLJPanel
+setMolecule (molecule:Molecule) +setPanel (panel:GLJPanel) +getMolecule():molecule +get_down():Punt3D +get_up():Punt3D +selectAtom(index:int) +setLights() +getOC():Vector3D +getCentre():Punt3D +getObservador():Punt3D +getU():Punt3D +draw_molecule(alpha:double,tetha:double:move:int) +draw_molecule(angle:double,eix:Vector3D) +draw_axis()

La classe **Eix** fa la definició i la visualització de l'eix de rotació de la molécula, de manera que l'usuari pugui seleccionar de manera interactiva aquest eix.

Atributs: rx,ry,rz corresponen als angles de rotació en els eixos, x,y,z respectivament, ox,oy,oz, spinx,spiny,spinz es fan servir com a punts de referència en la toma de dades de la posició del mouse, mentre l'usuari identifica l'eix desitjat. panel guarda una referència a l'entorn de visualització.

Mètodes: El mètode `getEix()` retorna un vector amb la definició de l'eix seleccionat. `Draw()` dibuixa una representació de l'eix en un moment determinat, `mousePressed()`, `mouseReleased()` i `mouseDragged()` agafen els events del mouse a partir de les accions que realitza l'usuari.

Eix
-rx:double -ry:double -rz:double -ox:double -oy:double -oz:double -spinx:double -spiny:double -spinz:double -panel: GLJPanel
+Eix () +getEix():Vector3D +draw() +draw(rx:double,ry:double,rz:double) +mousePressed(e:MouseEvent) +mouseReleased(e:MouseEvent) +mouseDragged(e:MouseEvent)

Classe **Dibuix**, aquesta classe conté l'entorn de visualització on es dibuixa tota l'escena i la molècula. Conté els elements interactius necessaris per a recollir l'entrada de l'usuari vers les rotacions i el zoom.

Atributs: Els atributs `cx,cy,cz,ox,oy,oz` corresponen al centre de l'escena i a la posició del observador respectivament. L'atribut `escena` conté una referència a una instància de l'objecte `Escena`. L'atribut `eix` conté el vector director de l'eix definit per efectuar la rotació. `width`, `height` defineixen el tamany de la finestra gràfica, `alpha`, `tetha` contenen els angles transversal i longitudinal de la rotació interactiva.

Mètodes: El mètode `display()` conté tot el necessari per efectuar la visualització de la molècula. `set_rotacio()` estableix si iniciem la rotació de la molècula. `get_rotacio()` comprova si estem fent la rotació. `setMolecule ()` assigna una referència a la molècula a visualitzar. `init()` inicia el context gràfic. `reshape()` s'invoca quan redimensionem la finestra gràfica. `display()` s'invoca cada vegada que necessitem redibuixar la finestra gràfica. Els mètodes `up()`,`down()`, `left()`,`right()`,`in()`,`out()` realitzen la funcionalitat de la navegació, rotar transversal o longitudinalment a la posició del observador, fet un `+zoom` o `-zoom`. El mètode `pick_mouse()` recull el click del mouse per part de l'usuari i tracta d'identificar si correspon a un àtom, si es així el marca com a seleccionat. El mètode `capture_imatge()` es crida per fer_rotació en el cas de que s'estigui fent guarden la rotació al disc.

Dibuix
-alpha: double -tetha:double -width:int -height:int -cx:double -cy:double -cz:double -ox:double -oy:double -oz:double -escena:Escena -eix:Vector3D
+Dibuix (width:int,height:int) +set_rotacio(fer_rotacio:boolean) +get_rotacio():boolean +setMolecule(molecule:Molecule) +init() +reshape(width:int,height:int) +display() +set_projection() +up() +down() +left() +in() +out() +evaluate() +keyPressed(e:KeyEvent) +set_press_mouse(x:int,y:int) +pick_mouse() +fer_rotacio(eix:Vector3D) +capture_imatge()

La classe **video** agafa l'informació necessària dins el moviment de rotació i la guarda al disc.

Atributs: `fos,out` son elements del model d'entrada/sortida de java necessaris per escriure a disc. `frames` conté el nombre màxim de frames a visualitzar. `angle` conté l'angle de rotació màxim a visualitzar. `arxiu` emmagatzema la carpeta de sortida.

Mètodes: `setArxiu ()` assigna la carpeta de sortida, `setFrames ()` assigna el nombre de frames màxim a visualitzar, `setAngle` assignen el angle màxim a visualitzar, `getAngle()` retorna el

angle, getFrames() retorna el nombre de frames. writeFramePPM() escriu a disc la imatge en format PPM que correspon al frame indicat. writeFrameTGA() escriu a disc la imatge en format TGA que correspon al frame indicat.

Video
-fos:FileOutputStream -out:ObjectOutputStream -frames:int -angle:int -arxiu:File
+setArxiu(f:File) +setFrames(frames:int) +setAngle(angle:int) +getFrames():int +getAngle():int +writeFramePPM(wi:int,hi:int,pixels:byte[],numframe:int) +writeFrameTGA(wi:int,hi:int,pixels:byte[],numframe:int)

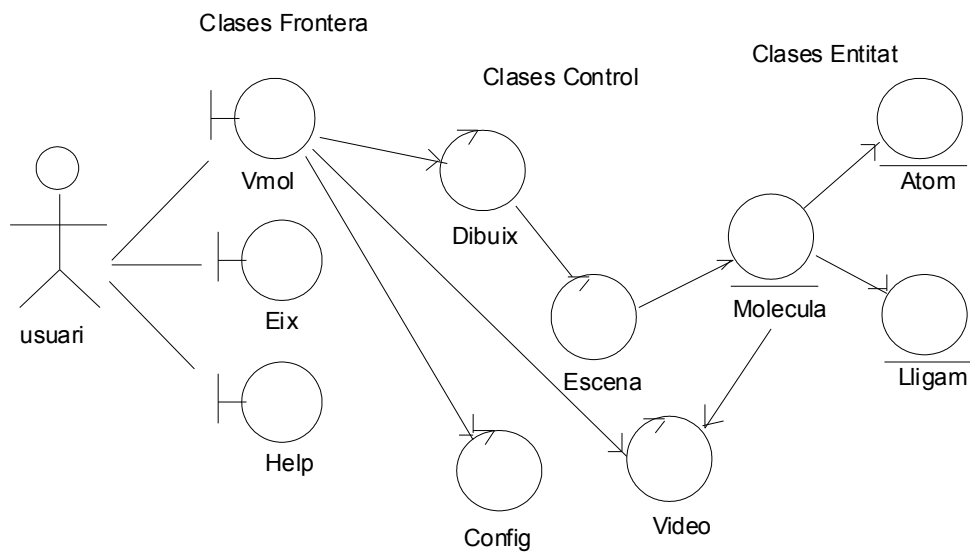
La classe **Help** serveix per a mostrar un quadre amb informació de ajuda sobre l'ús de la aplicació

Atributs: text, conté el text a mostrar, title, conté el títol a mostrar, url, conté la direcció de disc o d'internet a mostrar.

Mètodes: showContent(), mostra el quadre de diàleg. setTitle() assigna el títol, setContent() assigna el contingut, setUrl() assigna la url, setCenter() posiciona el quadre de diàleg al centre de la pantalla. openURL() sol·licita el contingut, el llegeix i el guarda a l'atribut text.

Help
-text: string -title: string -url: string
+showContent() +setTitle(title:string) +setContent(text:string) +setUrl(url:string) -createUI() -setCenter() -openURL(url:string)

5.6 Diagrama de col·laboració



5.7 Tipus de arxius de dades de molècula

Tipus d'arxiu .xyz

El contingut es troba en format de text pla (ascii), i la posició i significat dels diferents continguts són els següents:

línia 0, valor numèric que correspon al volum de la molècula

línia 1, nom de la molècula

línia 2 i successives: Lletres identificatives de l'àtom, espais, posició x, espais, posició y, espais, posició z.

Exemple de capçalera

```
33
XYZ file for : RutrpyCl3 in space group P21/n
Cl      -0.22851      1.54664      -2.38508
H       2.34562      3.28661      -0.36909
H       4.84634      2.90495      -0.10077
H       0.66805     -4.46990     -0.21726
H      -5.52481     -1.08419      0.03231
```

Tipus d'arxiu .xml

Format del arxiu d'entrada:

El format de definició d'una determinada molècula serà un fitxer en format XML que contindrà la següent informació:

```
<molecule name="denominació de la molècula" volume="[valor numèric]">
<atom name="[símbol de l'àtom]" x="[coord. x]" y="[coordenada y]" z="[coordenada z]" />
[...]
```

Format de l'arxiu de configuració:

L'arxiu de configuració conté les dades dels àtoms, el seu radi i el seu volum

```
<atoms>
<atom simbol="[símbol]" nom="[nom]" volum="[volum]" radi="[valor numèric]"
color="[identificació textual del color]" />
[...]
```

La configuració també es troba dins el codi de l'aplicació com a opció enllloc de ser necessari llegir-la d'un arxiu. Això es així partint del fet que les dades dels àtoms canviaran difícilment amb el pas del tems.

Els colors que es fan servir per a cada tipus d'àtoms són aquests: Hidrogen: verd, Carbó: vermell, Nitrogen: blau, Silici: groc, Ruteni: magenta, Clor: groc-verdós, Wolframi: gris-fosc. Solament es troben inserits un part dels àtoms de forma representativa, l'assignació dels colors es donada per l'arxiu atosizes donat coma especificació. En el cas de posar a producció l'aplicació serà necessari incloure totes les dades dels àtoms disponibles.

5.6 Format d'arxiu de sortida de la pel·lícula.

Inicialment he escollit com a format de sortida una seqüència d'imatges amb el format PPM, però l'abanic d'aplicacions que codifiquen en una arxiu de pel·lícula bassantse en imatges PPM es molt reduït.

Per tant he decidit finalment pel format TGA. Aquest format es acceptat per l'aplicació virtualdub. www.virtualdub.org eina de codi obert que es troba també al repositori de sourceforge.net.

Especificació del format d'imatge TGA.

La pel·lícula es guarda en una seqüència de imatges amb el format TGA. Amb el nom:

frame_[nombre].tga

On "[nombre]" es correspon amb la posició del fotograma dins la seqüència d'imatges.

L'arxiu es compon de dos parts diferenciades, una capçalera on es defineixen els paràmetres com ample, alçada, profunditat de color i un cos on es troba una serialització dels bytes que defineixen el color de cada píxel.

Capçalera:

1. 2 bytes amb el valor 0
2. 1 byte amb el valor 2 per a indicar que l'arxiu serà en format TGA no comprimit.
3. 5 bytes amb el valor 0
4. 2 bytes indicant l'origen de coordenada x on comença l'imatge, en el nostre cas amb el valor 0.
5. 2 bytes indicant l'origen de coordenada y on comença l'imatge, en el nostre cas amb el valor 0.
6. 1 byte amb la part de més pes de l'enter que defineix l'ample
7. 1 byte amb la part de menys pes del valor entre que defineix l'ample.
8. 1 byte amb la part de més pes de l'enter que defineix l'alçada
9. 1 byte amb la part de menys pes del valor entre que defineix l'alçada.
10. 1 byte amb el valor 24 que defineix una profunditat de color de 24 bits.
11. 1 byte amb el valor 0.

Cos:

Cada píxel es necessari codificar-lo en mode BGR. 3 bytes per a cada píxel, el primer byte amb la informació del blau, el segon amb la informació del verd i el tercer amb la informació del vermell.

Una explicació més ampla sobre el format TGA el podem trobar a la direcció <http://astronomy.swin.edu.au/~pbourke/dataformats/tga/>

Com a element comparatiu, faig també la descripció del **format PPM**. També consta de una capçalera amb la descripció dels paràmetres de l'imatge i d'un cos. La diferència més interessant es que el cos es codifica com a RGB i no com a BGR que es el cas del format TGA.

Capçalera:

1. L'arxiu comença amb la especificació del tipus de format PPM. En aquest cas escollim format "P6", es a dir, el fitxer comença amb la cadena P6.
2. Pot variar des de espai en blanc, retorn, tabulador o CR o LF.
3. L'ample de la imatge en codi ASCII i decimal.
4. Espai en blanc.
5. L'alçada de la imatge en codi ASCII i decimal.
6. Espai en blanc.
7. El valor màxim de identificació del color, de nou en ASCII i decimal, ha de ser menor de 65535 i major de zero.
8. Nova línia o un espai en blanc.

Cos:

9. Seqüència de bytes que corresponguin a tot el contingut de la imatge, consecutivament indicarem el color vermell, verd i blau, sense cap altre element divisor.

Eines de conversió i codificació

Posteriorment es possible agafar una aplicació d'edició de video y codificar aquesta seqüència d'imatges en una pel.lícula.

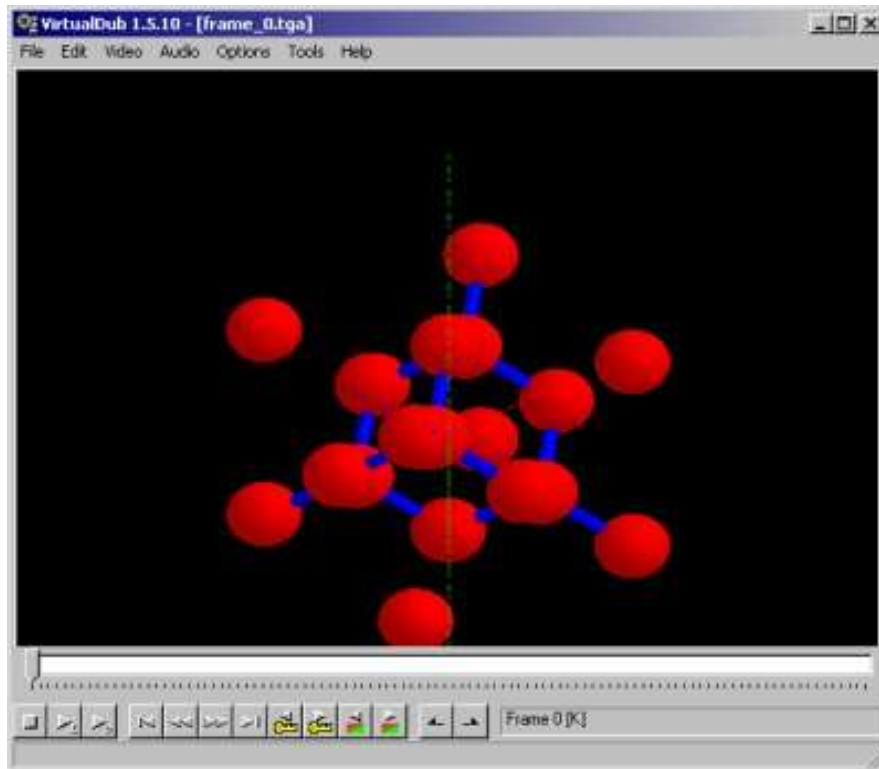
L'aplicació seleccionada es diu VirtualDub, que es pot trobar a www.virtualdub.org. Es una eina de codi obert, que es pot descarregar a la direcció:

<http://prdownloads.sourceforge.net/virtualdub/VirtualDub-1.5.10.zip?download>

Us de l'aplicació:

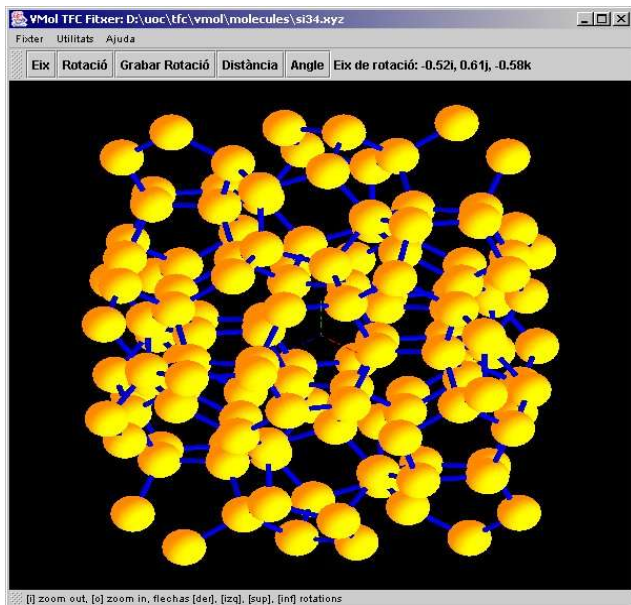
Una vegada descarregat l'arxiu zip que conté el virtualdub, es suficient crear un directori i desempaquetar el zip.

1. Executem l'aplicació i anem a l'opció obrir arxiu. (File->Open Vídeo File)
2. Escollim com a tipus d'arxiu "Imatge Secuence" i seleccionem el primer arxiu que correspon al primer fotograma de la nostra pel.lícula. Es necessari que les imatges tinguin com a nom un índex a través del qual l'aplicació pugui deduir l'ordre dels fotogrames.
3. Una vegada carregada la seqüència d'imatges anem a (video->Compression) i seleccionem un dels 'codecs' de video que tinguem instal·lats a la màquina. Per exemple 'Indeo'.
4. També tenim l'opció de redimensionar la nostra pel.lícula, si es el nostre cas podem anar a Video->Filters->Add i seleccionar l'opció resize escollint el tamany que desitgem.
5. Anem a File->Save as Avi i escollim un nom d'arxiu per la nostra pel.lícula.
6. Pulseu en acceptar i es codificarà la nostra pel.lícula.

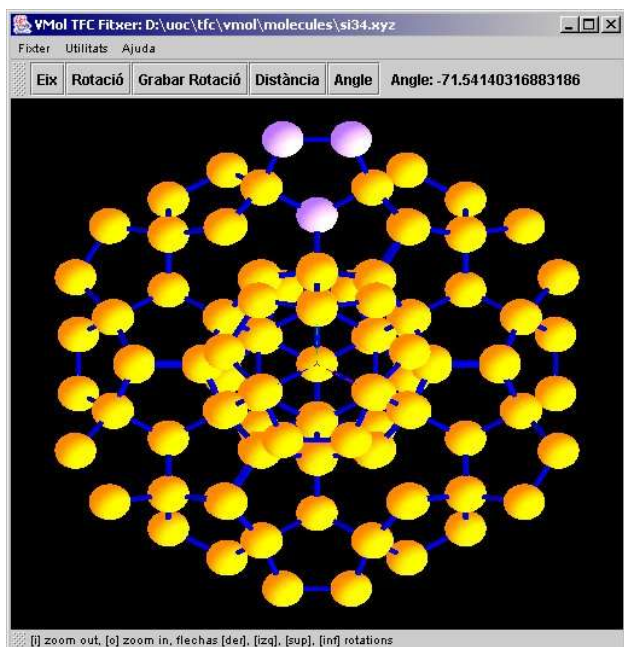


6. Captures de pantalla de l'aplicació amb diferents tipus de molècules.

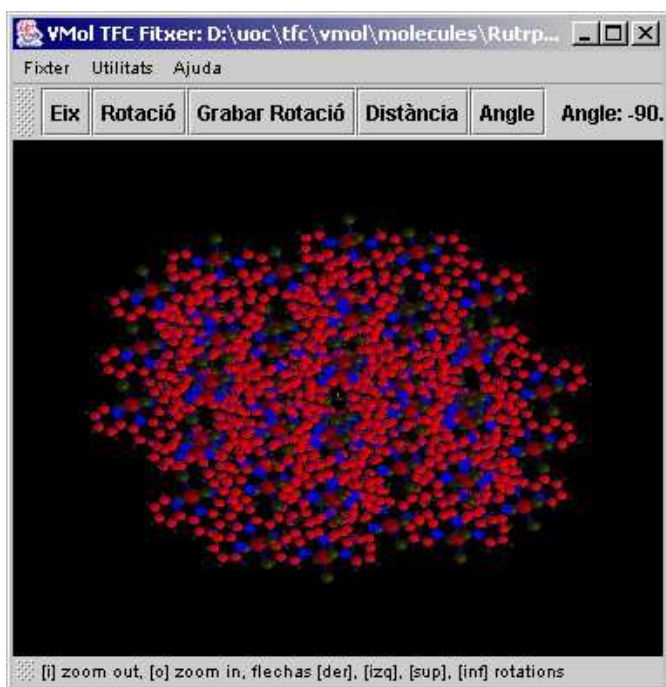
Imatge d'una molècula de silici. Els àtoms de silici estan identificats amb esferes de color groc i els enllaços amb cilindres de color blau.



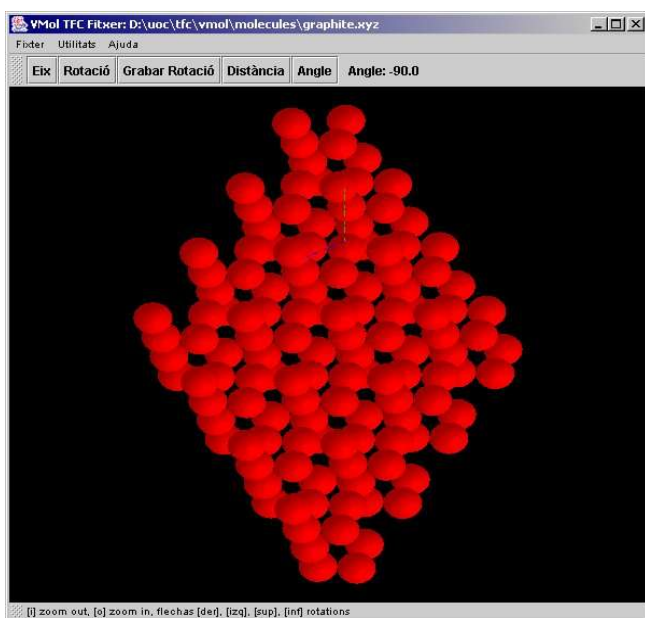
En aquesta altra imatge hem rotat la molècula amb l'ajuda de les tecles fletxa esquerra i em clicat sobre els tres àtoms indenticats amb color blanc. Una vegada clicats pulsem sobre el botó angle i ens mostra l'angle de les rectes que formen els àtoms en graus.



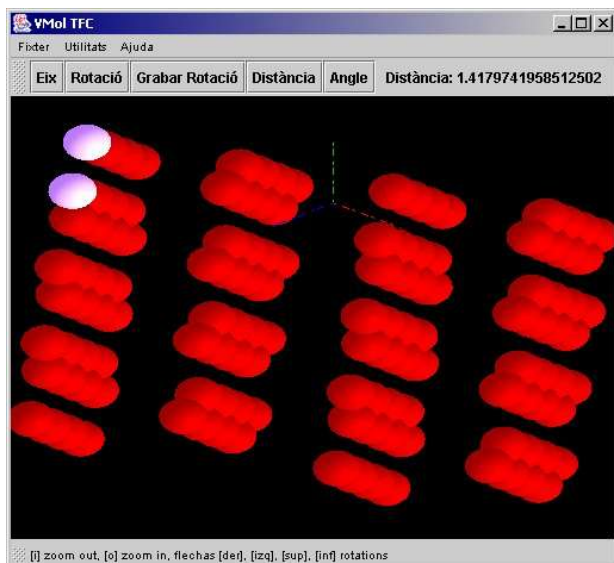
Aquí visualitzem la molècula RutrpyCl₃, els àtoms de clor estan identificats amb color groc-verdós, els de Ruteni amb color magenta, els de Hidrogen amb color verd i els de nitrogen amb color blau.



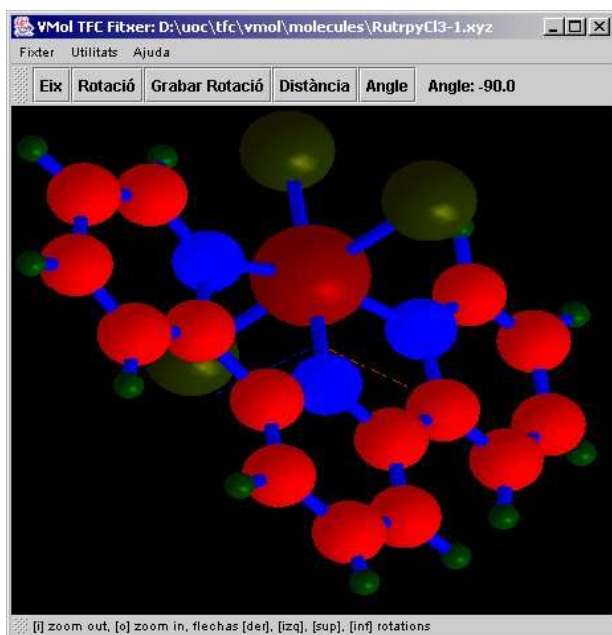
Aquí visualitzem una molècula de grafit, els àtoms de carbó estan identificats amb el color vermell.



Aquí podem veure la mateixa molècula de grafit amb una rotació i un zoom. Hem identificat 2 àtoms i em sol.licitat sàpiguer la distància que els separa.

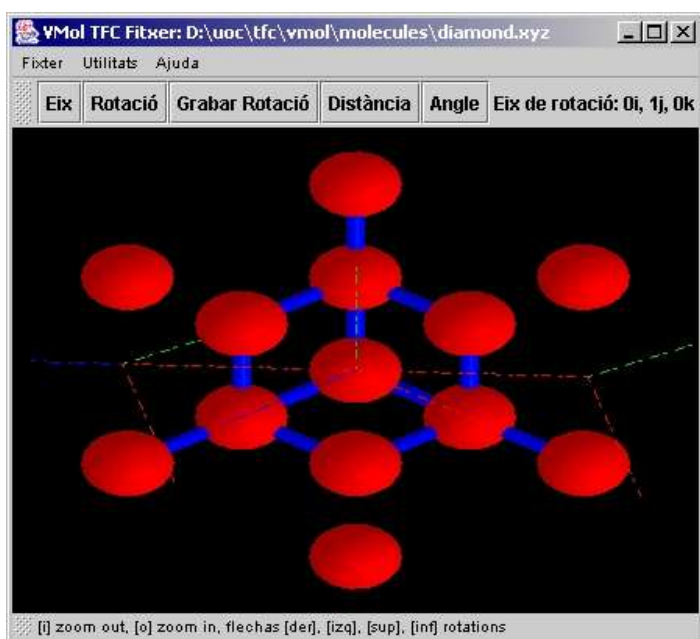


Molècula RutrpyCl3-1, els àtoms de Clor estan identificats amb color groc-verdós, els de carbó amb color vermell, els de Hidrogen amb color verd, i el de Ruteni amb color magenta.



Aquí visualitzem una molècula de diamant. Hem sol·licitat definir un eix de rotació i ens apareix una representació d'un eix mitjançant una línia puntejada amb els extrems identificats amb parells de eixos coordenats. Aquesta representació la podem rotar entorn el centre de representació amb l'ajuda del mouse. Desplacem el mouse dintre de l'àrea de visualització pulsant amb el botó dret.

Una vegada identificat ens apareix el valor del vector director que defineix aquest eix, amb coordenades i, j, k .



A l'opció fitxer->Info podem veure la definició de color que l'aplicació fa pels diferents àtoms i la informació llegida de la molècula.

Info				
simbol: W	nom: Wolframi	volum: 1.0	radi: 1.3	color: gris-fosc
simbol: Ru	nom: Ruteni	volum: 8.3	radi: 1.25	color: magenta
simbol: Na	nom: -	volum: 1.0	radi: 1.54	color: gris
simbol: Cl	nom: Clor	volum: 18.7	radi: 0.99	color: groc-verdos
simbol: H	nom: Hidrogen	volum: 14.3	radi: 0.32	color: verd
simbol: Si	nom: Silici	volum: 18.7	radi: 1.11	color: groc
simbol: N	nom: Nitrogen	volum: 17.33	radi: 0.75	color: blau
simbol: C	nom: Carbon	volum: 5.31	radi: 0.77	color: vermell
name: C (Diamond) & Fd(-3)m & #227 & O_h^7 & cF8 & A4 volum: 18.0				
simbol: C	nom: Carbon	volum: 5.31	radi: 0.77	color: vermell
	x: 4.01625	y: 0.44625	z: 0.44625	
simbol: C	nom: Carbon	volum: 5.31	radi: 0.77	color: vermell
	x: 0.44625	y: 0.44625	z: 0.44625	
simbol: C	nom: Carbon	volum: 5.31	radi: 0.77	color: vermell
	x: 2.23125	y: 2.23125	z: 0.44625	
simbol: C	nom: Carbon	volum: 5.31	radi: 0.77	color: vermell
	x: 4.01625	y: 4.01625	z: 0.44625	
simbol: C	nom: Carbon	volum: 5.31	radi: 0.77	color: vermell

7. Implementacions futures o millores

- Incloure a la finestra de visualització les dades dels àtoms participants com per exemple: nom, pes atòmic, radi covalent.
- lectura d'altres formats d'arxiu de molècules, com per exemple el format pdb.
- Definir interactivament dins l'aplicació el color de cada tipus d'àtom.

8. Identificació del procés de desenvolupament portat del TFC.

8.1. Planificació temporal.

Taula

Pas	data inici	data fi	PAC	operacions
1	14/09/04	27/09/04		Documentació, Presentació i definició de requeriments.
2	14/09/04	27/09/04	1	Elaboració del Pla de treball .
	27/09/04	13/12/04		(4.2) Implementació de la aplicació.
3	27/09/04	10/10/04	2	(4.2.a) Lectura de la informació de la molécula
4	10/10/04	24/10/04	2	(4.2.b) Modelatge de la molécula i representació virtual
5	24/10/04	02/11/04	2	(4.2.c) Creació de l'interfície d'usuari per a permetre realitzar la funcionalitat de moure la molécula dintre dels eixos, la identificació i la selecció de àtoms.
6	02/11/04	21/11/04	3	(4.2.d) Funcionalitats de càlcul de distàncies, angles de torsió i posicionament de l'eix per poder realitzar una rotació.
7	21/11/04	28/11/04	3	(4.2.e) definició del model d'il·luminació per la representació de l'escena.
8	28/11/04	13/12/04	3	(4.2.f) definició i disseny del format de la pel·lícula. Implementació de la captura de les imatges i de la generació del arxiu de sortida corresponent a aquesta pel·lícula.
9	13/12/04	10/01/05	Final	(4.3) Elaboració de la memòria
10	13/12/04	10/01/05	Final	(4.4) Elaboració de la presentació virtual

8.2 Lliuraments de PAC

Lliurament de PAC1

Un document en format pdf i en format sxw (openoffice) que contingui el pla de treball. Es adir, els arxius de nom vesteban_PlaTreball.pdf i vesteban_PlaTreball.sxw

Lliurament de PAC2

Part de la aplicació funcional que tingui implementades segons la taula 5.1 el pas 3,4 i 5. Es a

dir, la capacitat de llegir un fitxer d'entrada tal com es descriu i fer la representació en 3D en una finestra gràfica.

Es donarà també un fitxer de molécula d'exemple i un arxiu de configuració d'exemple per poder fer una execució de la aplicació i poder comprovar el passos descrits. També es donaran unes captures de pantalla en arxius .jpg de l'aplicació funcionant.

Es donaran els arxius .class compilats i els arxius font amb extensió .java.

Els documents descrits es lliuraran empaquetats en un arxiu .zip de nom vesteban_PAC2.zip

Lliurament de PAC3

Part de la aplicació funcional que tingui implementades la part de la PAC3 (pas 3,4 i 5) i els passos 6,7 i 8. Es a dir, la totalitat de la funcionalitat de l'aplicació descrita en el pla de treball.

De la mateixa manera es donarà també un fitxer de molécula d'exemple i un arxiu de configuració d'exemple per poder fer una execució de la aplicació i poder comprovar el passos descrits. També es donaran unes captures de pantalla en arxius .jpg de l'aplicació funcionant.

Es donaran els arxius .class compilats i els arxius font amb extensió .java.

Els documents descrits es lliuraran empaquetats en un arxiu .zip de nom vesteban_PAC3.zip

Lliurament de Treball Final

Al lliurament del treball final s'afegirà a més de l'aplicació la memòria, en format pdf i l'arxiu sxw (de openoffice) tal com es descriu i la presentació virtual en format .swf (flash), que permet una execució en mode de transparències virtuals.

Els documents descrits es lliuraran empaquetats en un arxiu .zip de nom vesteban_TreballFinal.zip

8.3 Identificació gràfica del temps de desenvolupament de les parts del projecte a lliurar.

PAC1, PAC2, PAC3, TFC Final

Setembre						
Dilluns	Dimarts	Dimecres	Dijous	Divendres	Dissabte	Diumenge
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			
Octubre						
Dilluns	Dimarts	Dimecres	Dijous	Divendres	Dissabte	Diumenge
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17

18	19	20	21	22	23	24
25	26	27	28	29	30	31
Novembre						
Dilluns	Dimarts	Dimecres	Dijous	Divendres	Dissabte	Diumenge
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					
Desembre						
Dilluns	Dimarts	Dimecres	Dijous	Divendres	Dissabte	Diumenge
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		
Gener						
Dilluns	Dimarts	Dimecres	Dijous	Divendres	Dissabte	Diumenge
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

9. Manual d'instal·lació

El procediment d'instal·lació constaria dels següents punts:

1. Instal·lar l'aplicació java (JDK 1.4.2)
2. Instal·lar la llibreria de java GL4Java
3. Instal·lar el visualitzador de molècules.
4. Opcionalment, si desitgem codificar video hem d'instal·lar també l'aplicació virtualdub.

Instal·lació JDK

Per a efectuar l'instal·lació de la aplicació es necessari tenir correctament instal·lades els paquets de software JAVA JDK 1.4 i GL4JAVA.

El software java JDK es pot trobar a <http://java.sun.com>, es disposa d'una aplicació executable que s'instal·la de manera interactiva i visual. El desenvolupament s'ha fet concretament amb la versió de java 1.4.2

L'aplicació té la forma d'un arxiu .exe que una vegada l'executem farà l'instal·lació de la màquina virtual de java. El procediment d'instal·lació ens demanarà l'ubicació de l'aplicació. Podem deixar el valor que surt per defecte.

Instal·lació GL4JAVA

Posteriorment es necessari instal·lar el paquet de GL4JAVA que es troba disponible a <http://www.jausoft.com/gl4java.html> Desde la pàgina principal hem d'anar a downloads i seleccionar l'opció de l'instal·lador.

Concretament podem disposar de l'instal·lador a la direcció web:

<http://www.jausoft.com/Files/Java/1.1.X/GL4Java/binpkg/gl4java-INSTALLER.zip>

Aquest paquet disposa d'un arxiu .bat que automatitza el procediment d'instal·lació. L'instal·lador es també una aplicació java. Executem l'arxiu .bat, farà la detecció de la instal·lació de java i ens sortirà un quadre de diàleg, l'hem de donar a acceptar. L'instal·lador descarregarà tots els components necessaris i els instal·larà dins de la nostra instal·lació de java.

Una vegada tenim aquests programaris instal·lats correctament podem desempaquetar l'aplicatiu desenvolupat vmol en un directori determinat i invocar al arxiu vmol.bat que iniciarà l'execució de l'aplicació.

Instal·lació visualitzador de molècules.

El visualitzador bé en la format d'un arxiu .zip. El podem desempaquetar a una carpeta nova i executar l'arxiu vmol.bat. Que el que fa es invocar a l'interpret de java amb el paràmetre vmol.VMol, es a dir

```
java vmol.VMol
```

Tenim uns arxius molécula d'exemple a la carpeta src\vmol\molecules

Compilació visualitzador de molècules.

Alternativament, si volem compilar el codi font, la distribució de la aplicació conté un arxiu amb denominació compilacio.bat que treballa directament sobre el directori src_vmol. Que

conté el codi font de la aplicació, els arxius .java.

Si executem compilacio.bat compilarà l'aplicació i copiarà els arxius executables per l'interpert de java a la carpeta vmol.

Posteriorment podem executar l'arxiu vmol.bat per iniciar l'aplicació amb la nostra compilació.

Per a poder compilar es necessari també tenir instal.lades els llibreries GL4JAVA i la màquina virtual.

L'entorn de desenvolupament de l'aplicació, l'aplicació IDE ha sigut Jcreator (www.jcreator.com), que també proveeix d'un entorn de compilació dins la mateixa eina.

10. Funcionament, manual d'ús

10.1 Introducció

L'aplicació s'inicia invocant a l'arxiu vmol.bat que llençarà l'aplicació. Si ja havíem iniciat l'aplicació anteriorment disposarem d'un arxiu de configuració on es queda emmagatzemada l'última selecció d'arxiu de molècula. (Arxiu de configuració de nom vmol.conf.tmp que es guarda a la mateixa carpeta on es troba el programari).

En aquest cas llegeix directament l'última selecció feta. Si no, es necessari indicar-li un arxiu de molècula abans de poder fer servir l'aplicació.

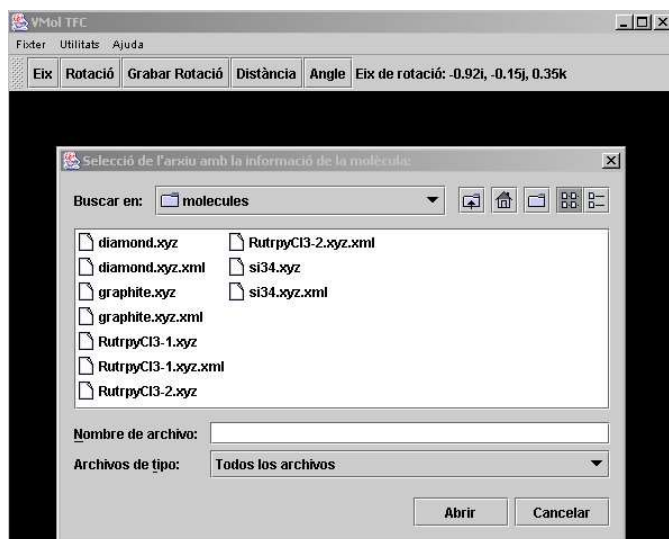
10.2 Menú d'opcions

Al menú tenim les opcions **Fitxer**, **utilitats** i **Ajuda**.

Fitxer



Amb **obre** podem seleccionar un arxiu de molècula que vulguem visualitzar. Els tipus d'arxius admesos son .xyz i .xml tal que estan descrits en l'especificació dels formats. En cas contrari no deixarà visualitzar-lo.



info, A Info Podem veure un volcat del contingut del arxiu de molècula i de les dades dels

àtoms emmagatzemades dins l'aplicació.

simbol:	nom:	volum:	radi:	color:
W	Wolframi	1.0	1.3	gris-fosc
Ru	Ruteni	8.3	1.25	magenta
Na	-	1.0	1.54	gris
Cl	Clor	18.7	0.99	groc-verdos
H	Hidrogen	14.3	0.32	verd
Si	Silici	18.7	1.11	groc
N	Nitrogen	17.33	0.75	blau
C	Carbon	5.31	0.77	vermell
name: Si(34) Clathrate & Fd(-3m) (#227) & cF136 volum: 136.0				
Si	Silici	18.7	1.11	groc
	x: 13.0788336	y: 2.7111936	z: 1.0048064	
Si	Silici	18.7	1.11	groc
	x: 13.0788336	y: 1.0048064	z: 2.7111936	
Si	Silici	18.7	1.11	groc
	x: 1.858	y: 1.858	z: 1.858	

sortir, A Sortir, sortim de la aplicació.

Utilitats

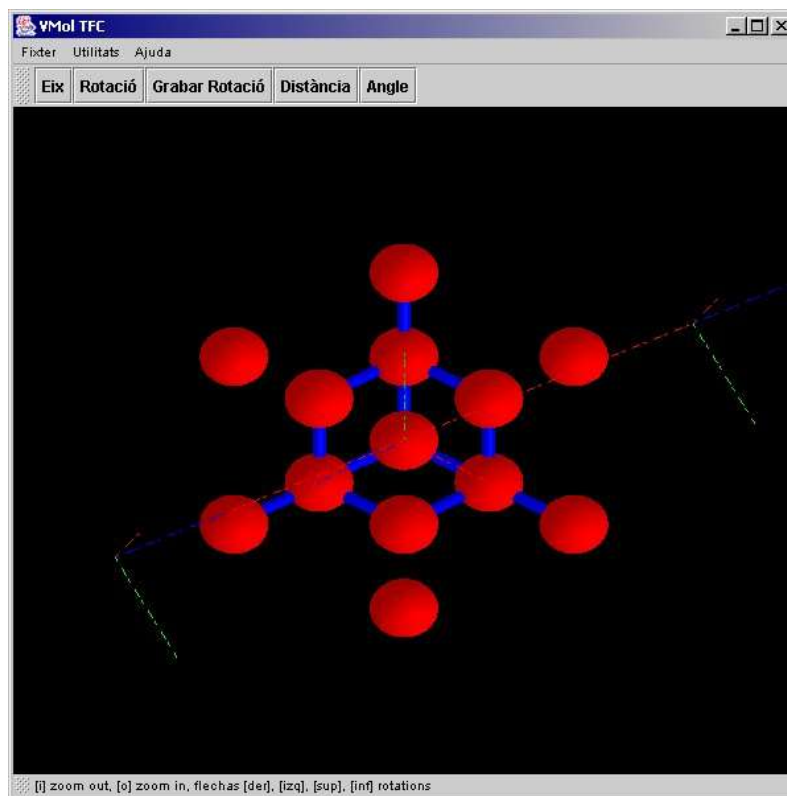


Tenim l'opció de mostrar u ocultar la barra d'utilitats. També disposem d'escollir entre els colors negre, gris, blanc i blau com a fons de pantalla en el moment de visualitzar la molécula. Aquesta opció ens es d'utilitat en el cas de que algunes molècules tinguin un color molt semblant al del fons i ens costi fer la identificació. L'opció deseleccionar àtoms ens deselecciona els àtoms que hagem pogut seleccionat prèviament.

A la barra d'utilitats tenim els següents buttons:



- **Eix**, que permet visualitzar un eix a través del qual nosaltres podem definir un eix de rotació de la molécula. Per ocultar l'eix es suficient amb tornar a pulsar sobre el mateix botó. Per defecte hi ha un eix definit que es el vector 0,1,0, es a dir, l'eix y.



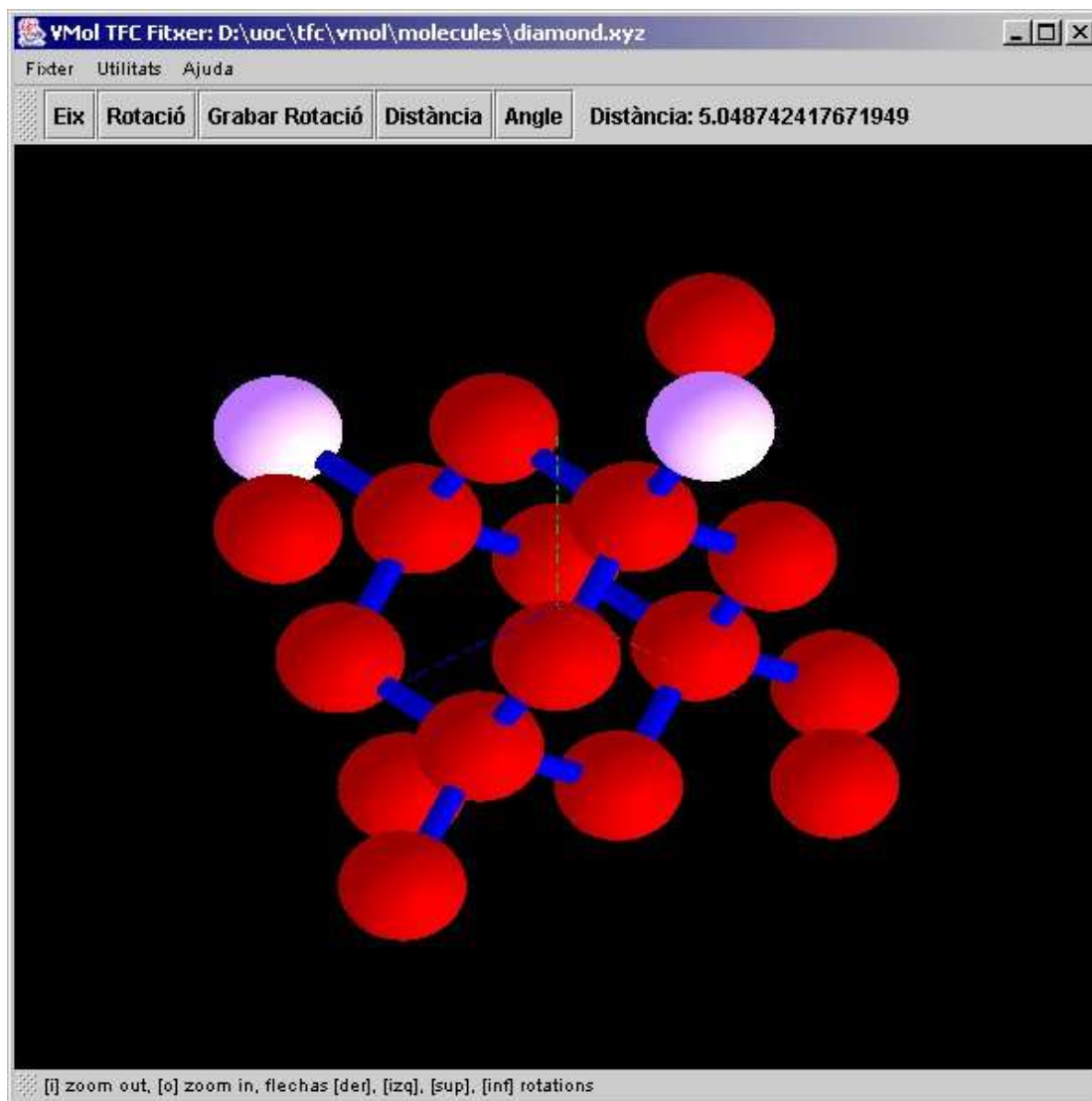
Es suficient tancant la ventana, ja es guarda la selecció de l'eix de rotació feta. Posteriorment es pot pulsar al botó rotació per veure una animació de la molècula rotant sobre l'eix definit. En cas de que no s'hagi seleccionat prèviament cap eix, es pren per defecte el vector (0,1,0);

- **Rotació**, que permet fer una rotació a través de l'eix prèviament definit.
- **Exportar**, que permet exportar la sortida de la rotació en forma de pel·lícula.

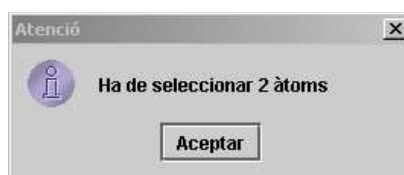
Al pulsar sobre gravar rotació apareix aquest quadre de diàleg, el qual ens demana seleccionar la carpeta de sortida de les imatges, indicar el nombre de fotogrames a capturar o alternativament l'angle màxim de rotació. Una vegada fet això pulsem sobre el botó iniciar i començarà la gravació de la pel·lícula.



- **Distància**, una vegada pulsat, es pot seleccionar 2 àtoms i ens donarà el valor numèric que correspon a la distància que separa els seus centres.

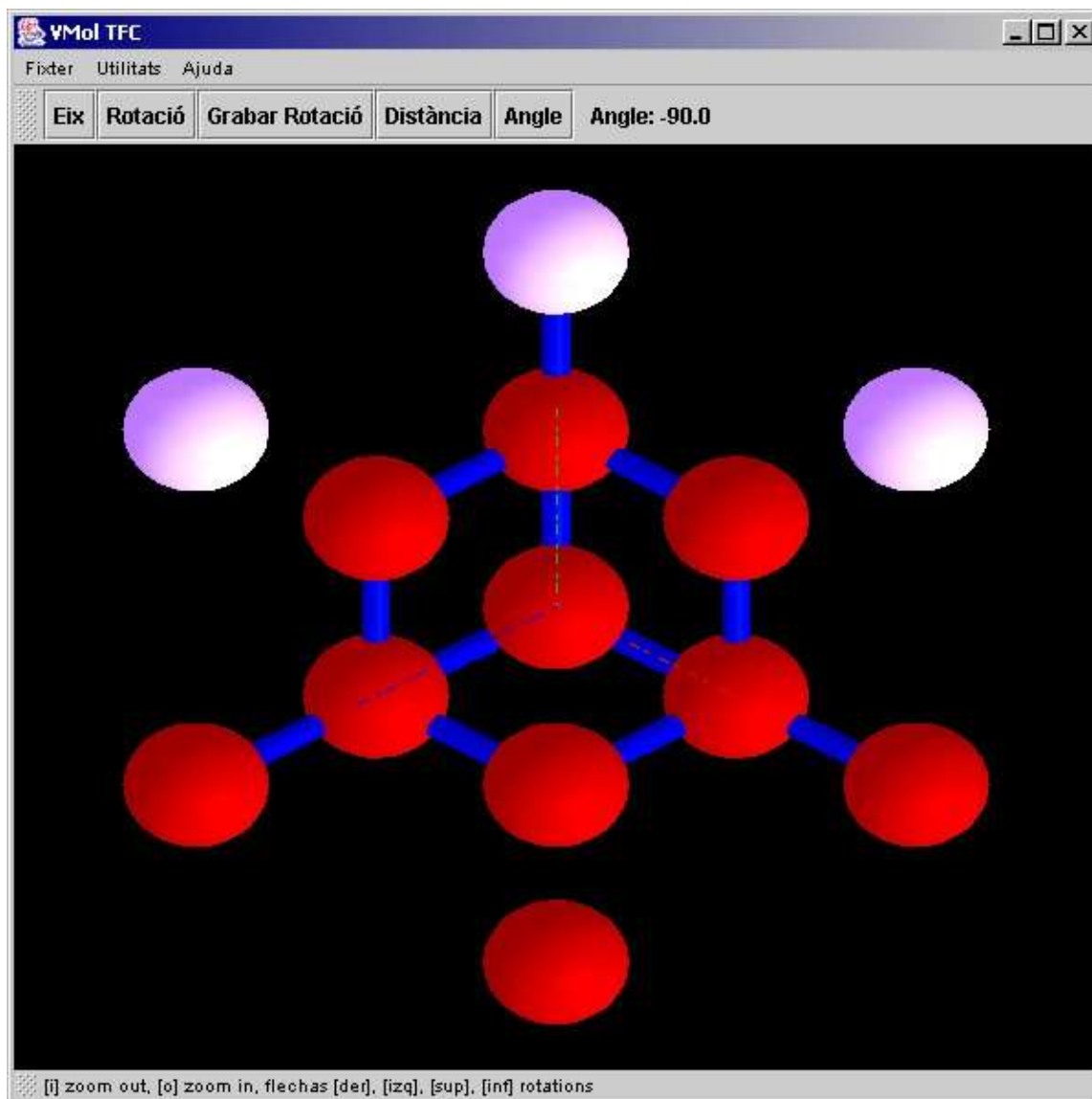


El resultat apareix al costat de la barra d'utilitats. En cas de que no haguem seleccionat 2 àtoms ens sortirà un missatge avisant-nos.



La selecció d'àtoms s'identifica pintant els àtoms de color blanc.

- **Angle**, una vegada pulsat, ens donarà el valor numèric del angle que forma les dues rectes que uneixen els àtoms. Si seleccionem consecutivament l'àtom 1,2 i 3, llavors tindrem les rectes 1-2, i 2-3, per tant l'angle que ens donarà serà el que formen aquestes dues rectes.



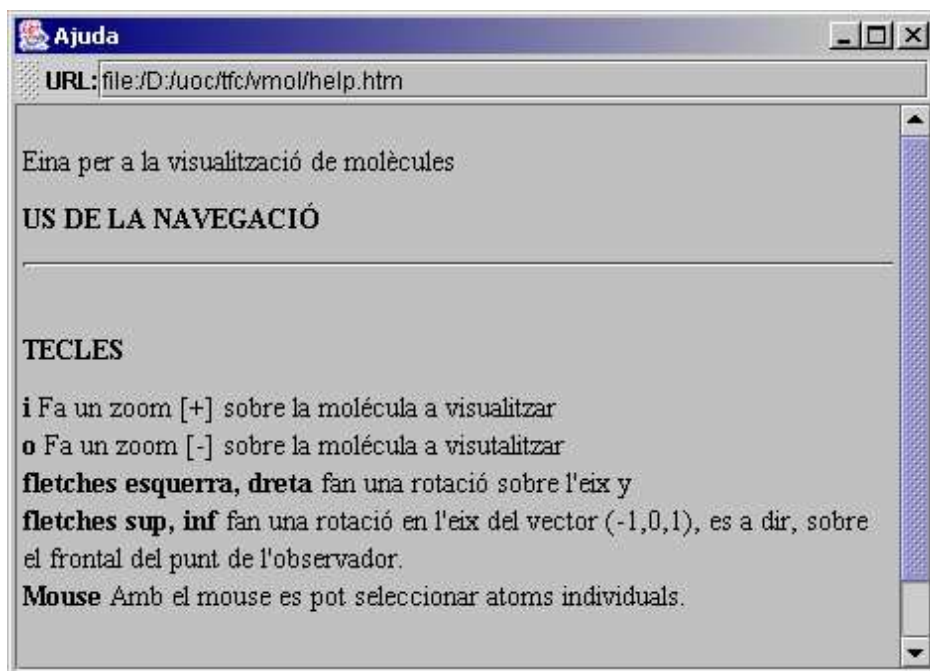
De la mateixa manera aquí ens apareix el valor numèric en graus (0-360) que correspon al angle format per les rectes que van a centre de cada un dels tres àtoms seleccionats. En cas contrari ens hauria sortit un missatge demanant-nos que seleccionem 3 àtoms.

El valor numèric ens apareix també al costat de la barra d'utilitats.

Ajuda

A Ajuda veiem una ventana amb informació sobre com podem fer la navegació de la molècula.





Navegació

Es pot navegar amb l'ajuda del teclat: [i] Fa un zoom in, [o] Fa un zoom out, fletxa esquera i dreta fa una rotació sobre l'eix y i les fletxes superior e inferior fan una rotació transversal a la visualització, es a dir, en l'eix (-1,0,1).

També es possible navegar desplaçant el mouse dins l'area de visualització.

11. Bibliografia

11.1 Referències

1. **OPENGL** <http://www.opengl.org/>
2. The Bluebook - http://www.opengl.org/documentation/blue_book_1.0/
3. The RedBook - http://www.opengl.org/documentation/red_book_1.0/
4. OpenGL Programming Guide, Addison Wesley Dev. Press, 2nd Edition 1997.
5. GL4Java, <http://www.jausoft.com>
6. Protein DataBank, <http://www.rcsb.org/pdb/>
7. RasMol, <http://www.umass.edu/microbio/rasmol/>
8. Format d'imatge TGA <http://astronomy.swin.edu.au/~pbourke/dataformats/tga/>
9. Aplicació de codificació de vídeo virtualdub www.virtualdub.org
10. IDE de desenvolupament Java Jcreator (www.jcreator.com)

11.2 tutorials

1. <http://java.sun.com/j2se/1.3/docs/api>
2. <http://www.xmission.com/%7Enate/opengl.html>
3. http://www.gametutorials.com/Tutorials/OpenGL/OpenGL_Pg1.htm