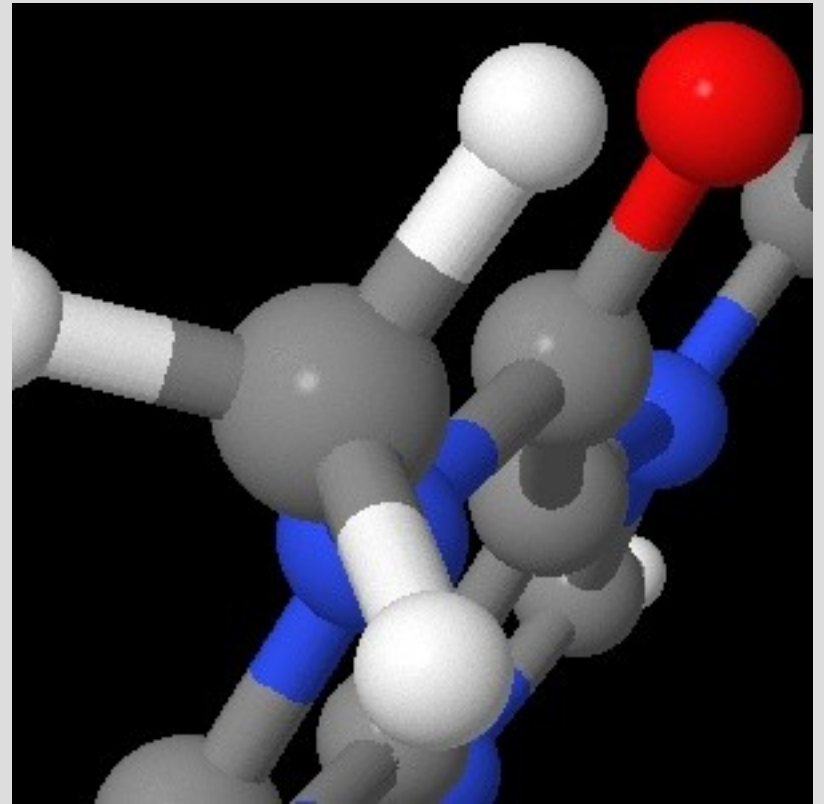


Jmol

Ein Kurzreferat von Sebastian Schneider

Jmol - Übersicht

- 1) Einleitung und Historie
- 2) Anwendungsgebiete und -beispiele
- 3) Jmol Funktionen
- 4) Jmol im CellMicrocosmos
- 5) Dokumentationen
- 6) Quellen



Jmol - Einführung

- Jmol ist ein molecular viewer
- Jmol ist open source und damit gratis
- Jmol gibt es in drei Varianten:
 - das **Jmol Applet** zur Integration in Web-Pages
 - die **Jmol Standalone** Applikation
 - der **Jmol Viewer** zur Integration in Java-Programme

Jmol - Historie

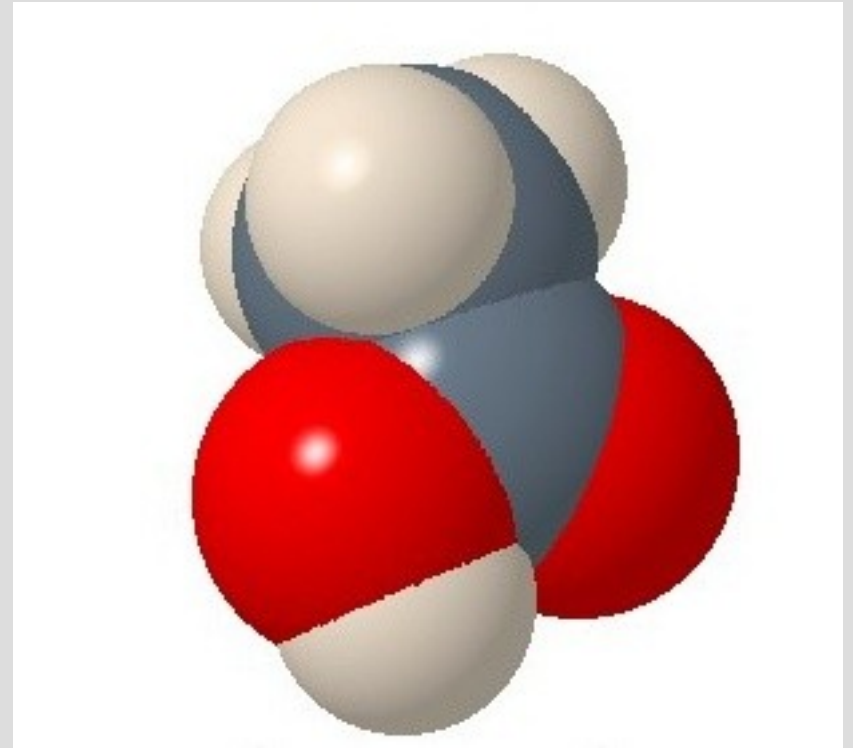
- Als Ersatz für das - nicht freie - Xmol gedacht
- Entstand im Rahmen eines Openscience Projekts
- 2002 neue Zielsetzung: Ersatz für das Chime Plug-In.
- Die Version v10.0 erscheint Dez.2004
- Ende 2006 erscheint 11.0 Beta

Jmol - Anwendungsgebiete

- Jmol wird in der Chemie und Biochemie verwendet
- u.A. zur Darstellung von:

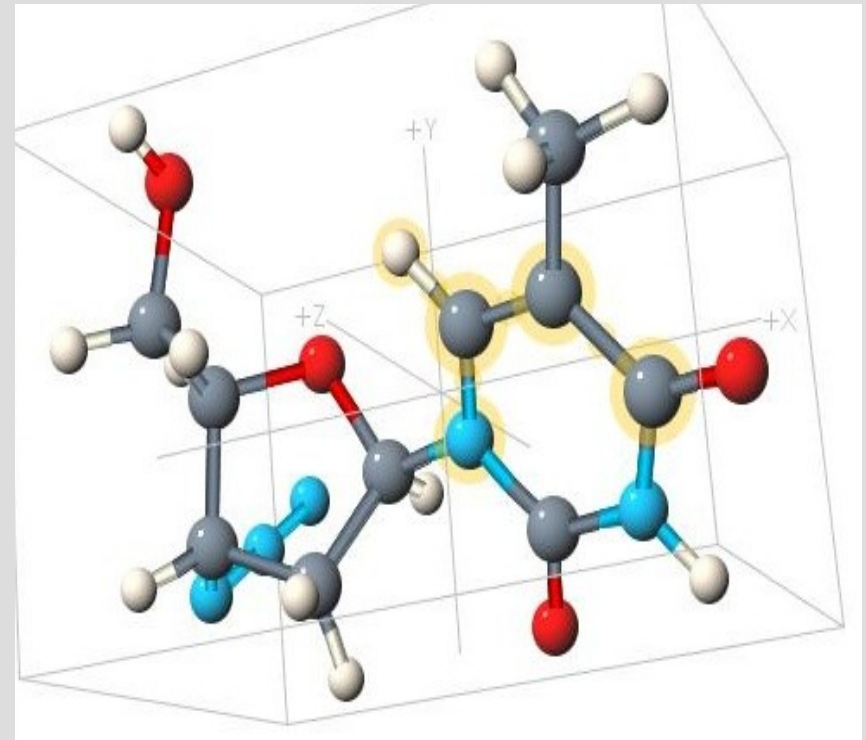
Jmol - Anwendungsbeispiele

- Kleinen Molekülen



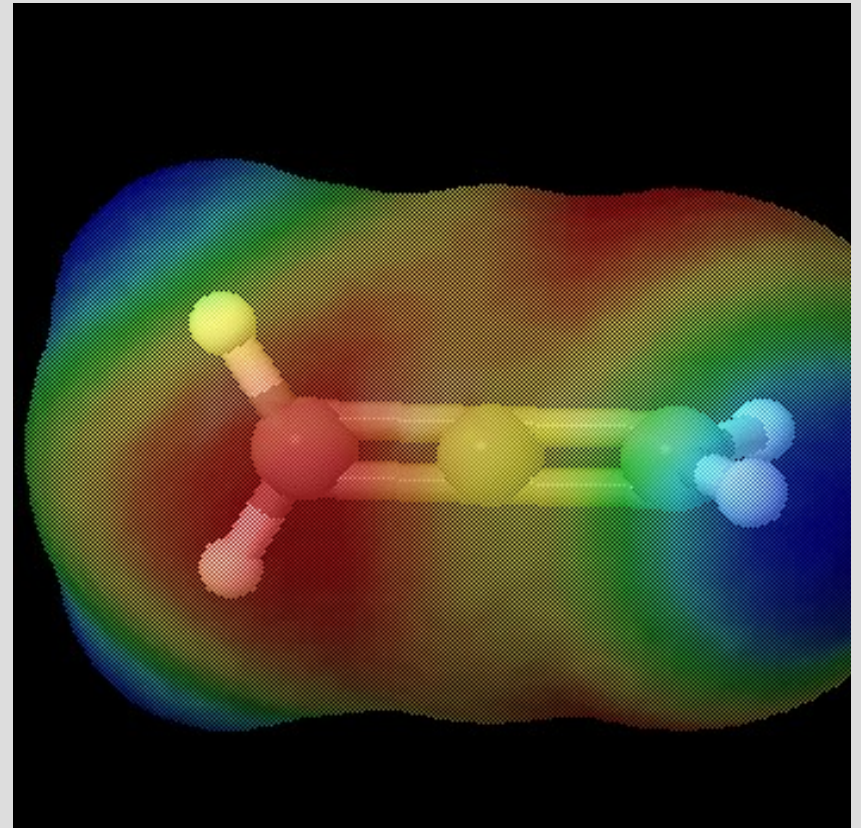
Jmol - Anwendungsbeispiele

- Räumlichen Orientierung von Molekülen



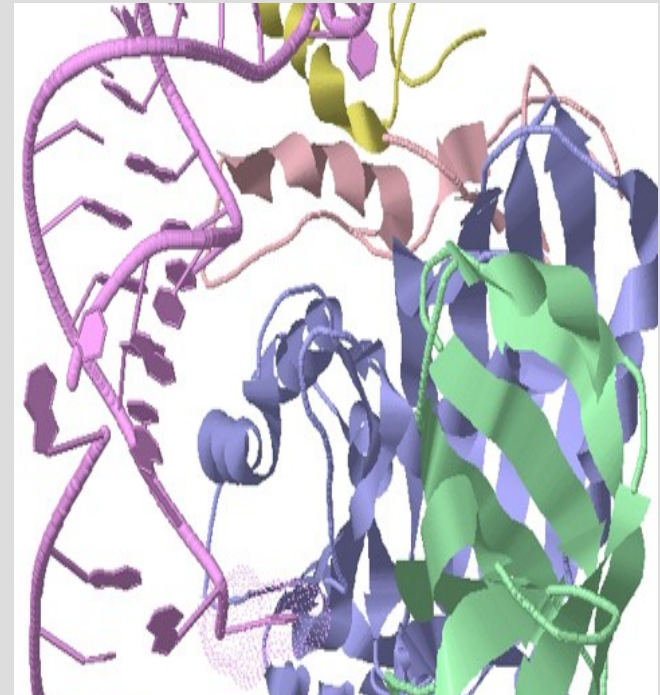
Jmol - Anwendungsbeispiele

- Elektrostatischen Potentialen



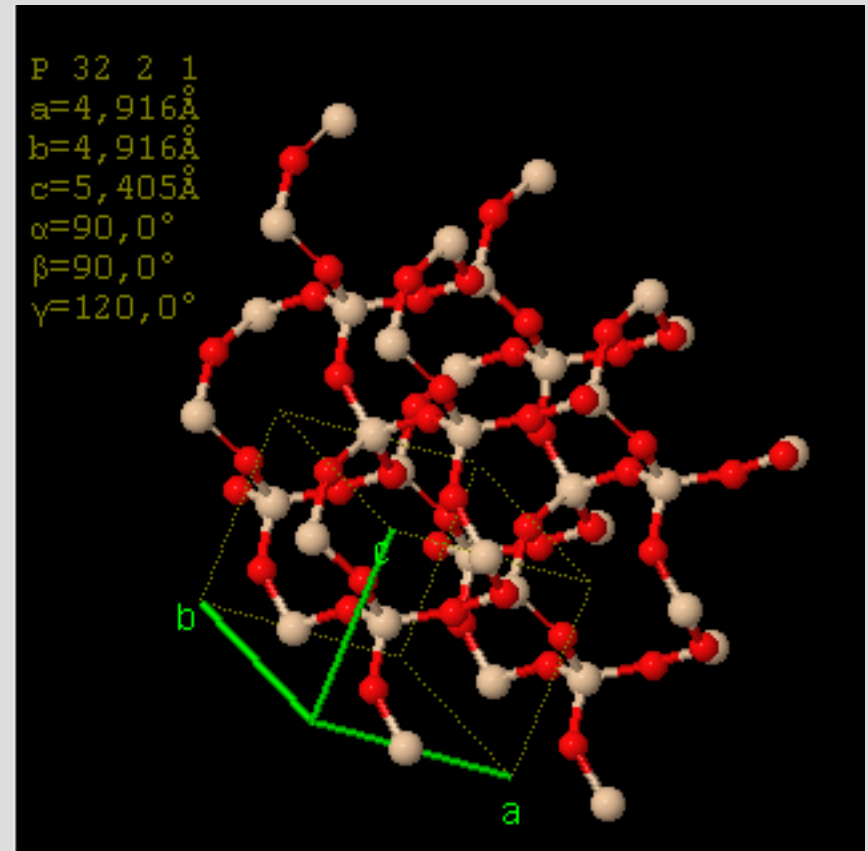
Jmol - Anwendungsbeispiele

- Makromolekülen



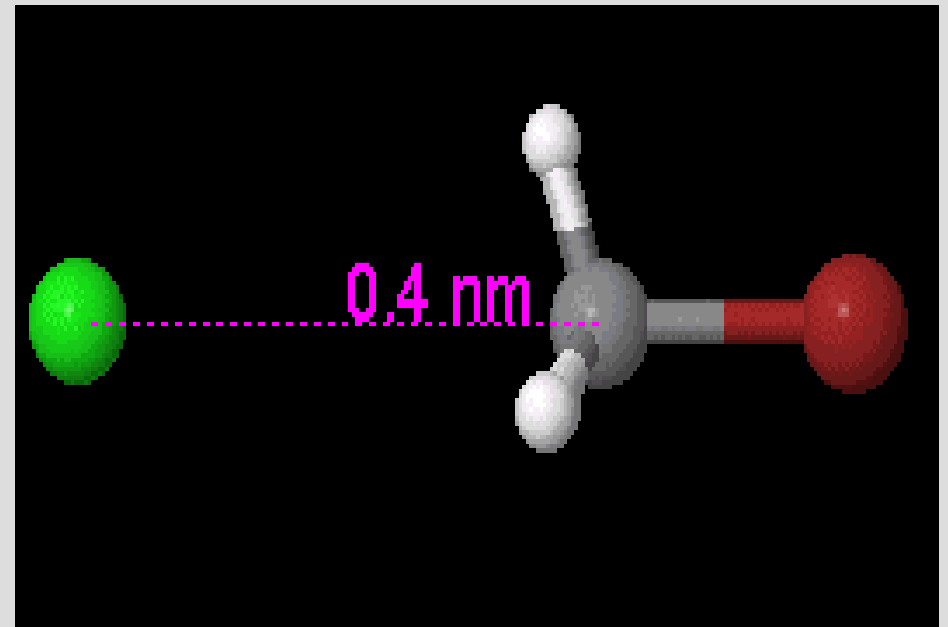
Jmol - Anwendungsbeispiele

- Kristallinen Strukturen



Jmol - Anwendungsbeispiele

- Chemischen Reaktionen



Jmol v11

- **Unterstützt:**
 - die gängigen **OS**
 - die **Formate**: PDB, CML, MOL, u.v.a.

Jmol v11

- **Bietet:**
 - high-performance **3D-Rendering** ohne Hardwareunterstützung
 - **Animationen**
 - **Vibrationen**
 - Schematische Darstellung von **Sekundärstrukturen**

Jmol v11

- Messungen von Distanzen, Winkeln und der Lage im Raum
- Skriptsprachenunterstützung
- Export als JPG, PNG, PovRay, u.a.

Jmol – Grafik Engine

- Die Grafik-Engine wurde komplett in Java geschrieben
- Verzichtet somit auf Java2D, Java3D, OpenGL, etc.
- Die Rendering-Engine wurde speziell zur Darstellung von Molekülen entwickelt

Jmol – Rendering

- Funktionsweise des Renderns:
 - **Farbe** der Pixel wird gebuffert
 - **Koordinaten** der Pixel werden gebuffert
 - „Hinter der Kulisse“ wird **gerendert** und in pixelBuffer geschrieben
 - erst dann wird ein einziges **Graphics.drawImage()** aufgerufen

Jmol – Darstellung

- Jmol hat eine eigene Farbpalette für:
- Elemente

H																					He
Li	Be														B	C	N	O	F	Ne	
Na	Mg														Al	Si	P	S	Cl	Ar	
K	Ca	Sc	Ti	V	Cr	Mn	Fe	Co	Ni	Cu	Zn	Ga	Ge	As	Se	Br	Kr				
Rb	Sr	Y	Zr	Nb	Mo	Tc	Ru	Rh	Pd	Ag	Cd	In	Sn	Sb	Te	I	Xe				
Cs	Ba	L*	Hf	Ta	W	Re	Os	Ir	Pt	Au	Hg	Tl	Pb	Bi	Po	At	Rn				
Fr	Ra	A*	Rf	Db	Sg	Bh	Hs	Mt													
(L:)	La	Ce	Pr	Nd	Pm	Sm	Eu	Gd	Tb	Dy	Ho	Er	Tm	Yb	Lu						
(A:)	Ac	Th	Pa	U	Np	Pu	Am	Cm	Bk	Cf	Es	Fm	Md	No	Lr						













Jmol - Darstellung

- Ketten

Chain ID	ATOM	HETATM
A, a	[192, 208, 255] C0D0FF	[144, 160, 207] 90A0CF
B, b	[176, 255, 176] B0FFB0	[128, 207, 152] 80CF98
C, c	[255, 192, 200] FFC0C8	[207, 144, 176] CF90B0
D, d	[255, 255, 128] FFFF80	[207, 207, 112] CFCF70
E, e	[255, 192, 255] FFC0FF	[207, 144, 207] CF90CF
F, f	[176, 240, 240] B0F0F0	[128, 192, 192] 80C0C0
G, g	[255, 208, 112] FFD070	[207, 160, 96] CFA060
H, h	[240, 128, 128] F08080	[192, 80, 112] C05070
I, h	[245, 222, 179] F5DEB3	[197, 174, 131] C5AE83
J, j	[0, 191, 255] 00BFFF	[0, 167, 207] 00A7CF
K, k	[205, 92, 92] CD5C5C	[181, 76, 76] B54C4C
L, l	[102, 205, 170] 66CDAA	[86, 181, 146] 56B592
M, m	[154, 205, 50] 9ACD32	[138, 181, 42] 8AB52A
N, n	[238, 130, 238] EE82EE	[190, 114, 190] BE72BE
O, o	[0, 206, 209] 00CED1	[0, 182, 161] 00B6A1
P, p	[0, 255, 127] 00FF7F	[0, 207, 111] 00CF6F
Q, q	[60, 179, 113] 3CB371	[52, 155, 97] 349B61
R, r	[0, 0, 139] 00008B	[0, 0, 187] 0000BB
S, s	[189, 183, 107] BDB76B	[165, 159, 91] A59F5B
T, t	[0, 100, 0] 006400	[0, 148, 0] 009400
U, u	[128, 0, 0] 800000	[176, 0, 0] B00000
V, v	[128, 128, 0] 808000	[176, 176, 0] B0B000
W, w	[128, 0, 128] 800080	[176, 0, 176] B000B0
X, x	[0, 128, 128] 008080	[0, 176, 176] 00B0B0
Y, y	[184, 134, 11] B8860B	[232, 182, 19] E8B613
Z, z	[178, 34, 34] B22222	[194, 50, 50] C23232
none/ numeric	[255, 255, 255] FFFFFFFF	[255, 255, 255] FFFFFFFF

Jmol - Darstellung

- Ladung

Charge			
-4	[255,0,0]	FF0000	
-3	[255,64,64]	FF4040	
-2	[255,128,128]	FF8080	
-1	[255,192,192]	FFC0C0	
0	[255,255,255]	FFFFFF	
1	[216,216,255]	D8D8FF	
2	[180,180,255]	B4B4FF	
3	[144,144,255]	9090FF	
4	[108,108,255]	6C6CFF	
5	[72,72,255]	4848FF	
6	[36,36,255]	2424FF	
7	[0,0,255]	0000FF	

Jmol - Darstellung

- Wasserstoffbrücken

Hydrogen bonds

Related command: `color hb`

Colors hydrogen bonds
hydrogen bonds in nucleic acids

Distance	
+2	FFFFFF
+3 (turns)	FF00FF
+4 (α -helix)	FF0000
+5	FFA500
-3	00FFFF
-4	00FF00
other (β -sheet)	FFFF00
nucleotide	FF8080

Jmol - Darstellung

- Aminosäuren

Protein "amino"		Protein "shapely"	
Ala	[200,200,200] C8C8C8	[140,255,140] 8CFF8C	
Arg	[20,90,255] 145AFF	[0,0,124] 00007C	
Asn	[0,220,220] 00DCDC	[255,124,112] FF7C70	
Asp	[230,10,10] E60A0A	[160,0,66] A00042	
Cys	[230,230,0] E6E600	[255,255,112] FFFF70	
Gln	[0,220,220] 00DCDC	[255,76,76] FF4C4C	
Glu	[230,10,10] E60A0A	[102,0,0] 660000	
Gly	[235,235,235] EBEBEB	[255,255,255] FFFFFFFF	
His	[130,130,210] 8282D2	[112,112,255] 7070FF	
Ile	[15,130,15] 0F820F	[0,76,0] 004C00	
Leu	[15,130,15] 0F820F	[69,94,69] 455E45	
Lys	[20,90,255] 145AFF	[71,71,184] 4747B8	
Met	[230,230,0] E6E600	[184,160,66] B8A042	
Phe	[50,50,170] 3232AA	[83,76,82] 534C52	
Pro	[220,150,130] DC9682	[82,82,82] 525252	
Ser	[250,150,0] FA9600	[255,112,66] FF7042	
Thr	[250,150,0] FA9600	[184,76,0] B84C00	
Trp	[180,90,180] B45AB4	[79,70,0] 4F4600	
Tyr	[50,50,170] 3232AA	[140,112,76] 8C704C	
Val	[15,130,15] 0F820F	[255,140,255] FF8CFF	
Asx	[255,105,180] FF69B4	[255,0,255] FF00FF	
Glx	[255,105,180] FF69B4	[255,0,255] FF00FF	
other	[190,160,110] BEA06E	[255,0,255] FF00FF	

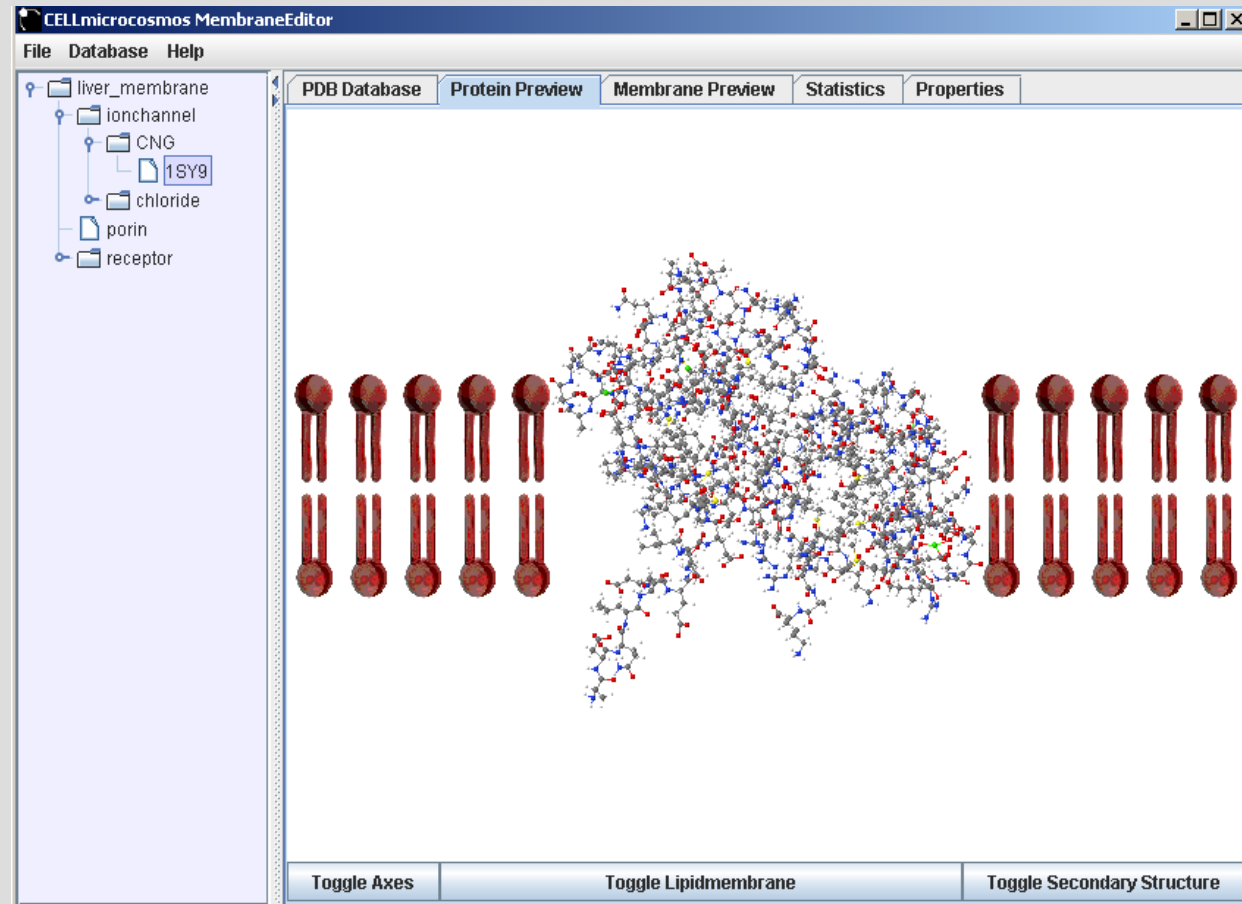
Jmol - Darstellung

- Sekundärstrukturen von Proteinen
- Und vieles mehr
- Jede Farbpalette kann individuelle angepasst werden

Protein		
α -helix	[255,0,128]	FF0080
β -sheet	[255,200,0]	FFC800
(β) turn	[96,128,255]	6080FF
other	[255,255,255]	FFFFFF

Jmol im Cellmicrocosmos

- Jmol wird verwendet um Proteine und Lipide darzustellen
- Lipide im Moment noch abstrakt



Jmol im Cellmicrocosmos

- Proteine können über Jmol:
 - Im Raum gedreht
 - Eingefärbt werden
 - Strukturen können dargestellt werden
 - etc.
- Für den Cellmicrocosmos wichtig:
Orientierung der Proteine in der Membran

Jmol im Cellmicrocosmos

- Für die Zukunft eventuell wichtig:
 - Genaue Darstellung der Lipide
 - Räumliche Orientierung der Lipide
 - Maßstabs gerechte Darstellung der Übergänge Lipide – Proteine
 - Darstellung der gesamten Membran als Einheit
 - Navigation in der Membran

Jmol - Dokumentationen

- Die Jmol Homepage:
<http://jmol.sourceforge.net/>
- Die Jmol Mailinglist -> Anmeldung auf der Jmol Page
- Ein Jmol Forum:
<http://www.nabble.com/Jmol-f4403.html>
- Ein kleines Jmol Tutorial:
<http://www.bioc.unizh.ch/nanowelt/JmolTutorial/index.html>

Jmol - Dokumentationen

- Das Jmol Wiki:
http://wiki.jmol.org:81/index.php/Main_Page
- Jmol Users Forum auf sourceforge.net:
<http://sourceforge.net/mailarchive/forum.php?forum=jmol-users>

Quellen

- <http://jmol.sourceforge.net/>