



# *Hands-on Drug Discovery on Grid*

**Ana Lucia Da Costa**

HealthGrid, France

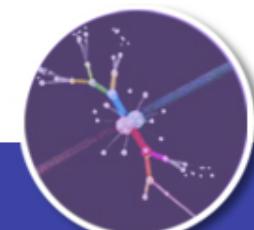
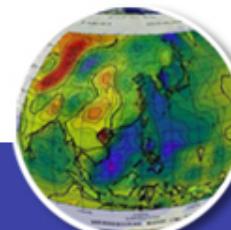


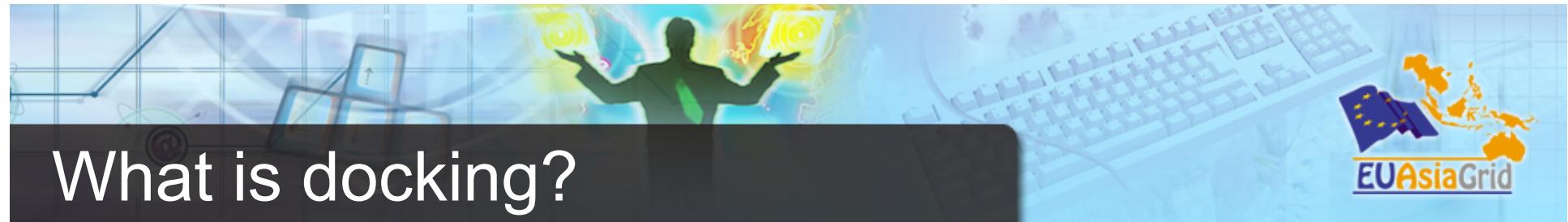
**ACGRID School**

*Kuala Lumpur, 10 Nov. 2009*

[www.euasiagrid.org](http://www.euasiagrid.org)

[www.euasiagrid.eu](http://www.euasiagrid.eu)





# What is docking?

*“Best ways to put two molecules together.”*

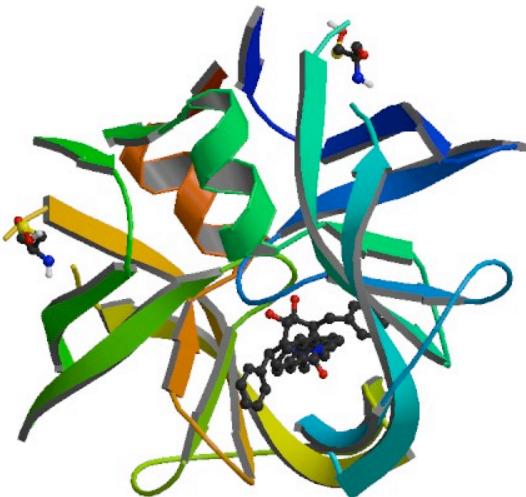


**Molecular Docking:** the prediction of the optimal bound conformation of two molecules exerting geometrical and chemical complementarity.

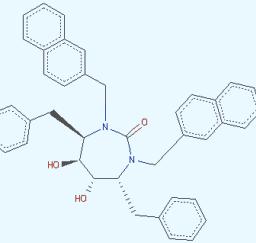
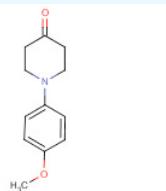
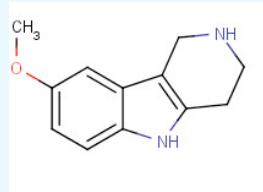
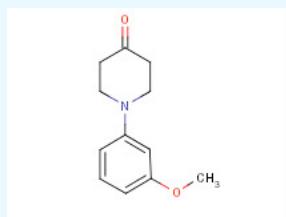
# Today example



## RECEPTOR



## LIGAND



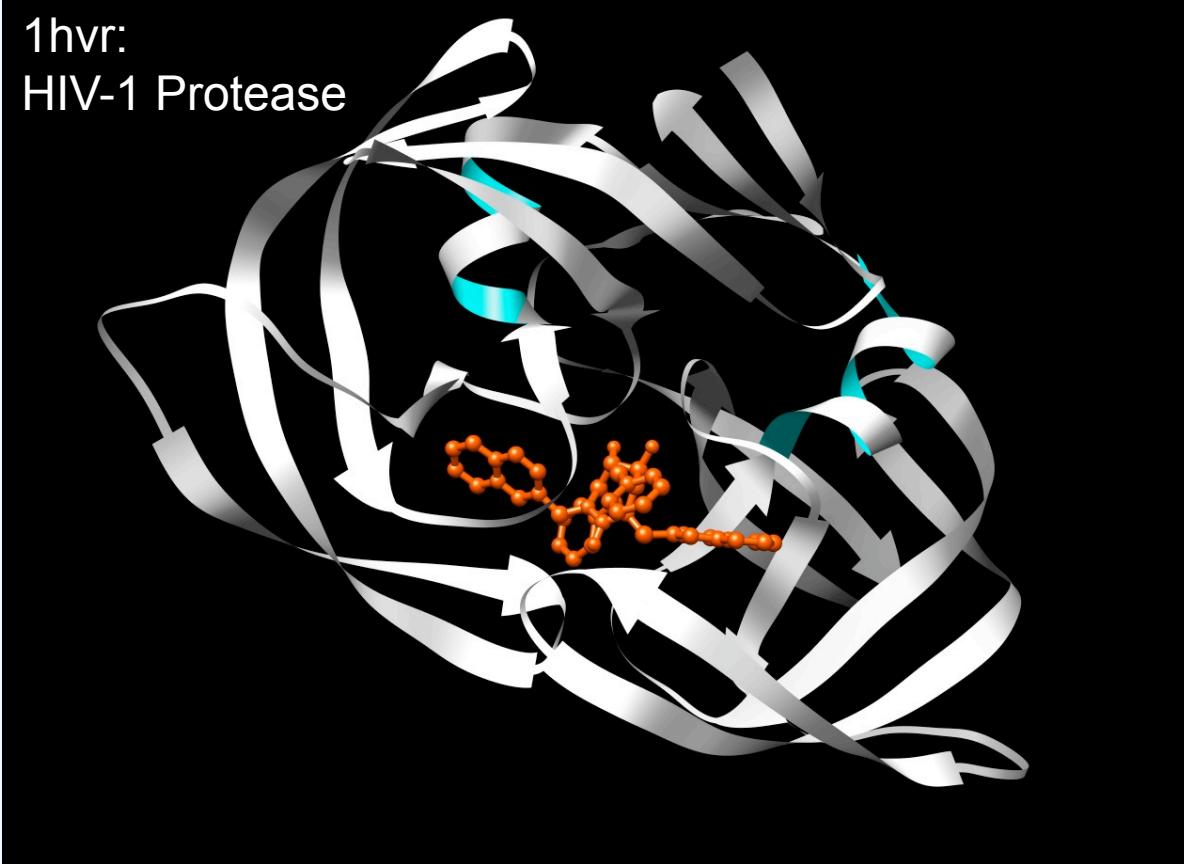
- Active HIV protease is required for viral infectivity
  - PDB code: 1hvr
  - Co-cristallized inhibitor: XK2

- Chembridge
  - 5 Random compounds

# Today example

RECEPTOR  
LIGAND

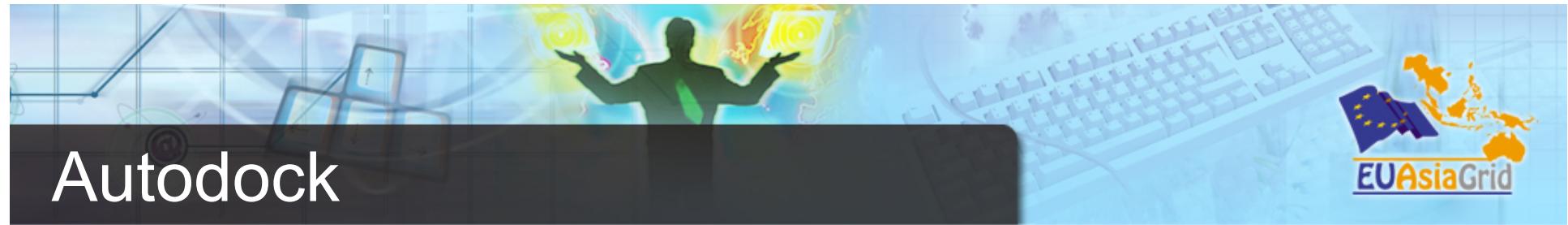
1hvr:  
HIV-1 Protease



1hvr\_lig:  
Co-cristallized  
inhibitor

# Docking Programs

- DOCK: (Kuntz et al. 1982)
- DOCK 4.0 (Ewing & Kuntz 1997)
- **AutoDOCK (Morris et al. 1998)**
- GOLD (Jones et al. 1997)
- FlexX: (Rarey et al. 1996)
- GLIDE: (Friesner et al. 2004)
- ADAM (Mizutani et al. 1994)
- CDOCKER (Wu et al. 2003)
- CombiDOCK (Sun et al. 1998)
- DIVALI (Clark & Ajay 1995)
- DockVision (Hart & Read 1992)
- FLOG (Miller et al. 1994)
- GEMDOCK (Yang & Chen 2004)
- Hammerhead (Welch et al. 1996)
- LIBDOCK (Diller & Merz 2001)
- MCDOCK (Liu & Wang 1999)
- PRO\_LEADS (Baxter et al. 1998)
- SDOCKER (Wu et al. 2004)
- QXP (McMartin & Bohacek 1997)
- Validate (Head et al. 1996)
- ....



# Autodesk

- Automated docking of flexible ligands to macromolecules and designed to predict how small molecules, such as substrates or drug candidates, bind to a receptor of known 3D structure.
  
- When is AutoDock not suitable
  - No 3D structures available;
  - Modelled structure of poor quality;
  - Too many (torsions, atoms, types);
  - Target protein too flexible.

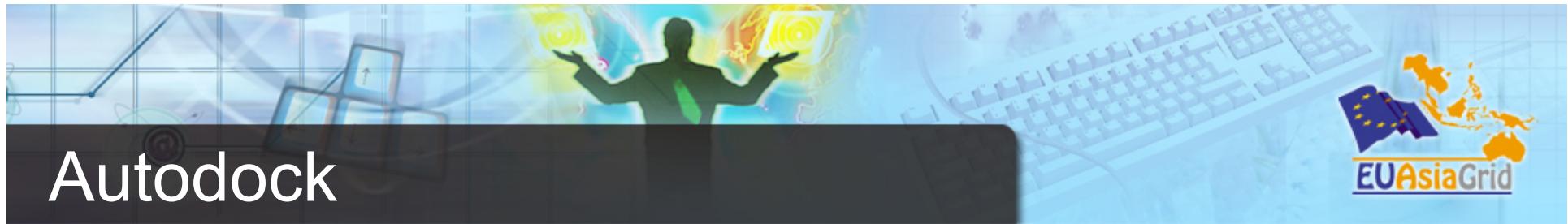
## How a Genetic algorithm works...

- Genetic algorithm is inspired by evolutionary biology
- Genotype = Ligand conformation : a chromosome constituted by real valued genes representing:
  - Ligand translational
  - Ligand orientational
  - Ligand conformational
- Phenotype = Coordinates of given conformation



Degrees of freedom





# Autodock

- For each run, Autodock successively:
  - Create a population of individuals
  - Assign a random translation, a random orientation and x random torsions to each individual
  - Begin Lamarckian Genetic Algorithm (LGA), with a maximum of energy evaluations and maximum generations
  
- Final Lamarckian genetic algorithm docked state
  - At the end of each run, Autodock outputs a result which is the lowest docked energy conformation of the ligand it found during that run

*Lower energy are “better” and in the genetic algorithm “fitter” > RANK*

# Scoring function

- Scoring function takes a pose as input and returns a number indicating the likelihood that the pose represents a favorable binding interaction

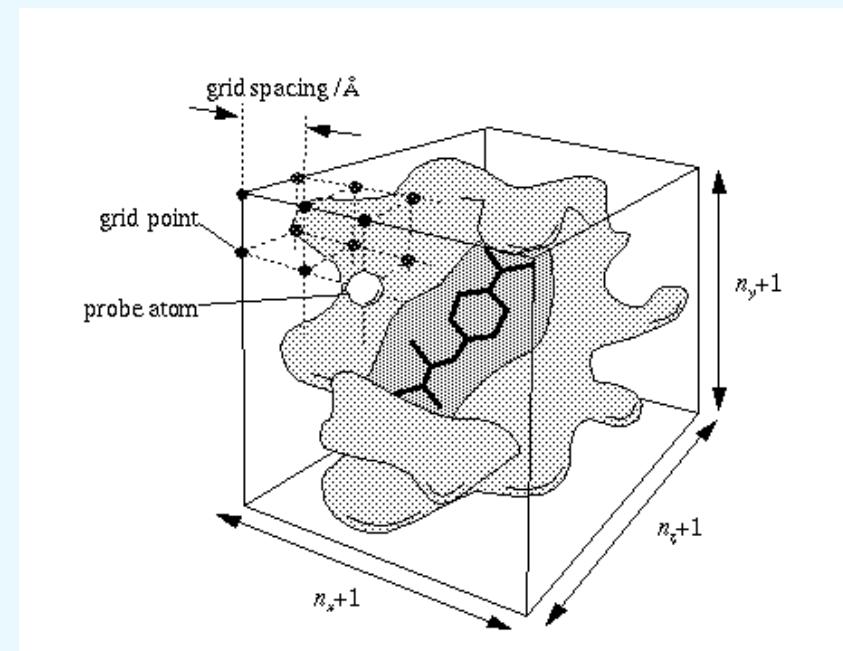
$$\Delta G_{binding} = \Delta G_{vdW} + \Delta G_{elec} + \Delta G_{hbond} + \\ \Delta G_{desolv} + \Delta G_{tors}$$

- $\Delta G_{vdW}$   
12-6 Lennard-Jones potential
  - $\Delta G_{elec}$   
Coulombic with Solmajer-dielectric
  - $\Delta G_{hbond}$   
12-10 Potential with Goodford Directionality
  - $\Delta G_{desolv}$   
Stouten Pairwise Atomic Solvation Parameters
  - $\Delta G_{tors}$   
Number of rotatable bonds
- Molecular mechanics

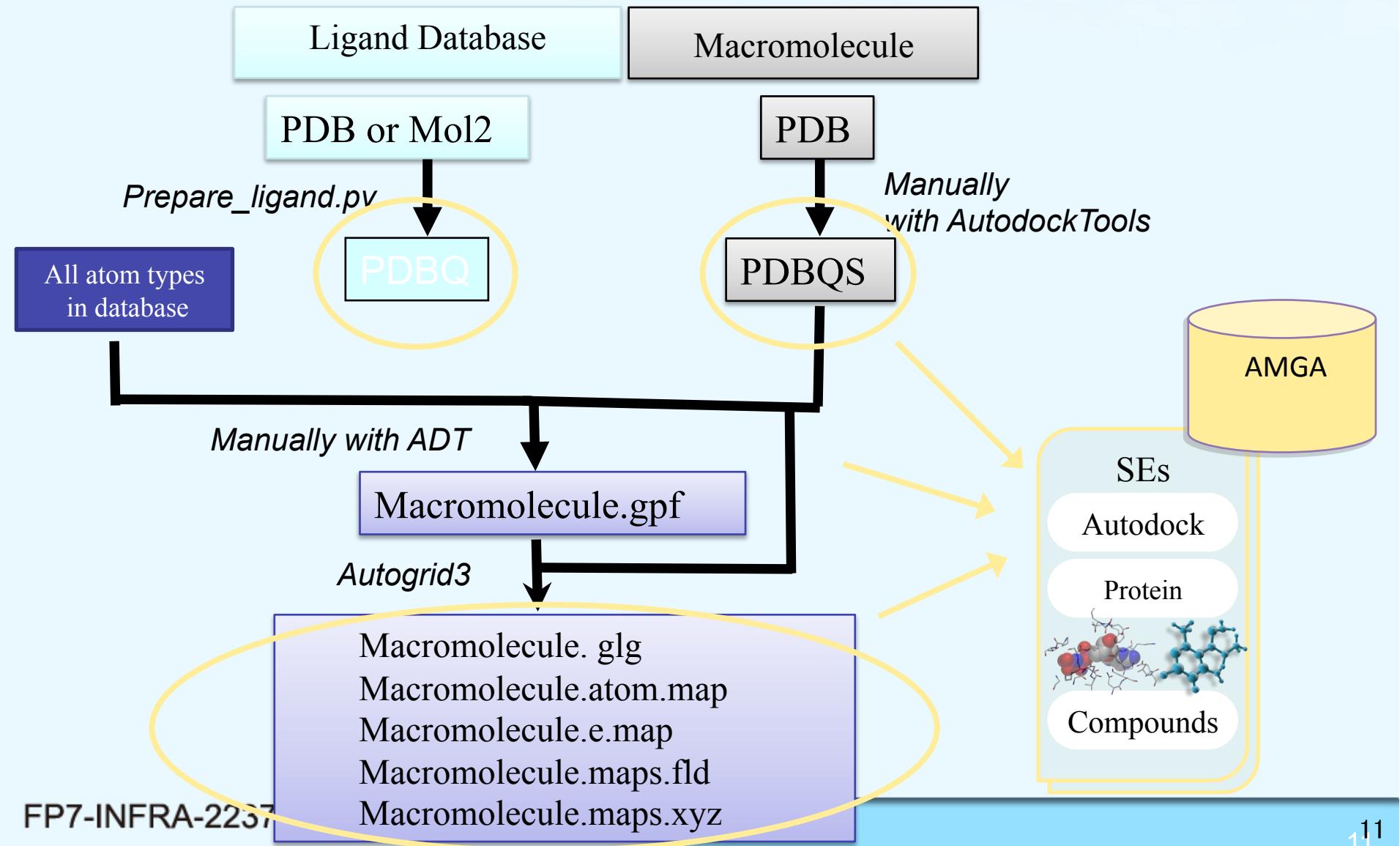
# Grid Maps



- Autogrid pre-calculate grid maps, one for each atom type present in the ligand database + electrostatic map
- Docking calculations extremely fast ~100x faster
- A grid map consists of a three dimensional lattice of regularly spaced points surrounding the active site
- Each point within the grid map stores the potential energy of a “probe” atom that is due to all the atoms in the macromolecule



# Preparation with Autogrid





Visit <http://www.pdb.org/>

**PDB PROTEIN DATA BANK**

[WHAT'S NEW](#) | [HELP](#) | [PRINT](#)

[MyPDB Login](#) A MEMBER OF THE OPDB

As of **Tuesday Nov 03, 2009 at 4 PM PST** there are 61248 Structures | [PDB Statistics](#)

[Summary](#) [Derived Data](#) [Sequence](#) [Seq. Similarity](#) [Literature](#) [Biol. & Chem.](#) [Methods](#) [Geometry](#) [Links](#)

**RATIONAL DESIGN OF POTENT, BIOAVAILABLE, NONPEPTIDE CYCLIC UREAS AS HIV PROTEASE INHIBITORS**

DOI:10.2210/pdb1hvr/pdb

**Primary Citation**

**Rational design of potent, bioavailable, nonpeptide cyclic ureas as HIV protease inhibitors.** Lam, P.Y., Jadhav, P.K., Eyermann, C.J., Hodge, C.N., Ru, Y., Bacheler, L.T., Meek, J.L., Otto, M.J., Rayner, M.M., Wong, Y.N., et al. (1994) *Science* **263**: 380-384

PubMed: [8278812](#)

[Search Related Articles in PubMed](#)

**PubMed Abstract:**

Mechanistic information and structure-based design methods have been used to design a series of nonpeptide cyclic ureas that are potent inhibitors of human immunodeficiency virus (HIV) protease and HIV replication. A fundamental feature of these inhibitors is the cyclic urea ...

[ [Read More & Search PubMed Abstracts](#) ]

**Molecular Description**

Classification: [Hydrolase\(acid Proteinase\)](#)  
Structure Weight: 22154.36

Molecule: HIV-1 PROTEASE  
Polymer: 1 Type: polypeptide(L)  
Chains: A, B Length: 99

**Source**

**1hvr**

[Display Files](#)   
[Download Files](#)   
[Print this Page](#)  
[Share this Page](#)

**Biological Molecule**

[More Images...](#)

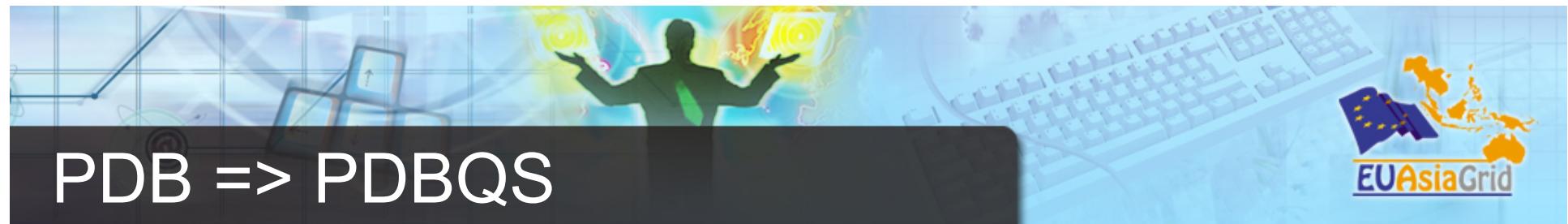
[View in Jmol](#) SimpleViewer  
Protein Workshop  
Other Viewers

FP7-INFRA-223791

12

# About protein preparation

- Structure from PDB is often uncorrect
- Check integrity of protein structure:
  - Presence of all atoms in all amino acid side-chains
  - “Bump” check to ensure all amino acid side chains atoms are not overlapping in the model
  - [PROCHECK](#) is a suite of programs to check the stereochemical quality of protein structures
- Remove all non essential heteroatoms:
  - Non-essential water molecules from the structure
  - Co-cristallized ligands - > save coordinates of known inhibitor in a new file
- Energy minimization is not mandatory to perform before docking. But yes, if you do it, you would get more genuine results.



# PDB => PDBQS

## □ Using AutodockTools:

- Add polar Hs. (Bonded to heteroatoms like nitrogen or oxygen)
- Remove non-polar Hs (bonded to carbon atoms)
  - Edit > Delete > Delete Hydrogens
  - Edit > Hydrogens > Add > Polar Only > No Bond Order > Yes to Renumbering

## □ Consider pH : Histidines

- File > Load Module > RepairCommands
- Edit Hydrogens > Edit Histidines Hydrogens > select HE2  
(neutral, HD1; neutral, HE2; or protonated)

## □ Create PDBQS file

- Grid > Macromolecule > Choose Macromolecule (AG3)

*Automatically :*

- \**Added Kollman charges*
- \**Merge Non-Polar Hydrogens*
- \**Added solvation parameters*

# Your files

## TARGET: protein.tar.gz

- ✓ 1hvr\_rec.pdbqt
- ✓ 1hvr\_rec.A.map
- ✓ 1hvr\_rec.C.map
- ✓ 1hvr\_rec.HD.map
- ✓ 1hvr\_rec.N.map
- ✓ 1hvr\_rec.OA.map
- ✓ 1hvr\_rec.e.map
- ✓ 1hvr\_rec.d.map
- ✓ 1hvr\_rec.maps.fld
- ✓ 1hvr\_rec.maps.fld

Target in autodock4 format



Floating point number : grid point energies

Check that all the maps it reads in are compatible

Size of the grid box in each dimension x, y, z

## LIGANDS: ligands.tar.gz

- ✓ 1hvr\_lig.pdbqt
- ✓ 4001115\_1.pdbqt
- ✓ 4001464\_1.pdbqt
- ✓ 4001711\_1.pdbqt
- ✓ 4002723\_1.pdbqt
- ✓ 4002577\_1.pdbqt

Co-crystallized ligand

Chembank ligands

# Your files

## ❑ DOCKING PARAMETER FILE: dpf.tar.gz

- ✓ 1hvr\_lig.dpf
- ✓ 4001115\_1\_1hvr\_rec.dpf
- ✓ 4001464\_1\_1hvr\_rec.dpf
- ✓ 4001711\_1\_1hvr\_rec.dpf
- ✓ 4002723\_1\_1hvr\_rec.dpf
- ✓ 4002577\_1\_1hvr\_rec.dpf

## ❑ SOFTWARE: autodock4.tar.bz2

# Docking parameter file

```

putlev 0                      # diagnostic output level
intelec                         # calculate internal electrostatics
seed pid time                   # seeds for random generator
ligand_types A C HD N OA       # atoms types in ligand
fld_1hvr_rec.maps fld          # grid_data_file
map 1hvr_rec.A.map             # atom-specific affinity map
map 1hvr_rec.C.map             # atom-specific affinity map
map 1hvr_rec.HD.map            # atom-specific affinity map
map 1hvr_rec.N.map             # atom-specific affinity map
map 1hvr_rec.OA.map            # atom-specific affinity map
elecmap 1hvr_rec.e.map          # electrostatics map
desolvmap 1hvr_rec.d.map        # desolvation map
move 1hvr_lig.pdbqt            # small molecule
about -9.2131 16.2654 28.1101  # small molecule center
crane random                   # initial coordinates/A or random
quat0 random                   # initial quaternion
ndihe 10                        # number of active torsions
dihe0 random                   # initial dihedrals (relative) or random
tstep 2.0                       # translation step/A
qstep 50.0                      # quaternion step/deg
dstep 50.0                      # torsion step/deg
torsdof 8                       # torsional degrees of freedom and coefficient
unbound 0.0                     # free energy of ligand's unbound state
rhistol 0.5                     # cluster_tolerance/A
extnrg 1000.0                  # external grid energy
e0max 0.0 10000                # max initial energy; max number of retries
ga_pop_size 150                 # number of individuals in population
ga_num_evals 25000               # maximum number of energy evaluations
ga_num_generations 27000         # maximum number of generations
ga_elitism 1                    # number of top individuals to survive to next generation
ga_mutation_rate 0.02            # rate of gene mutation
ga_crossover_rate 0.8            # rate of crossover
ga_window_size 10                #
ga_cauchy_alpha 0.0             # Alpha parameter of Cauchy distribution
ga_cauchy_beta 1.0              # Beta parameter Cauchy distribution
set_ga                           # set the above parameters for GA or LGA
sw_max_its 300                  # iterations of Solis & Wets local search
sw_max_succ 4                   # consecutive successes before changing rho
sw_max_fail 4                   # consecutive failures before changing rho
sw_rho 1.0                       # size of local search space to sample
sw_lb_rho 0.01                  # lower bound on rho
ls_search_freq 0.06              # probability of performing local search on individual
set_sw1                          # set the above Solis & Wets parameters
compute_unbound_extended        # compute extended ligand energy
ga_run 10                        # do this many hybrid GA-LS runs
analysis                         # perform a ranked cluster analysis

```

Output Level, 0 = minimal

Link to map file

Ligand

Translation/Quaternion/Torsions

Genetic algorithm parameters

Local Search parameters

Docking runs / Clustering



# Starting....

## Connect to the UI

\$ ssh kualalumpurXX@glite-tutor.ct.infn.it

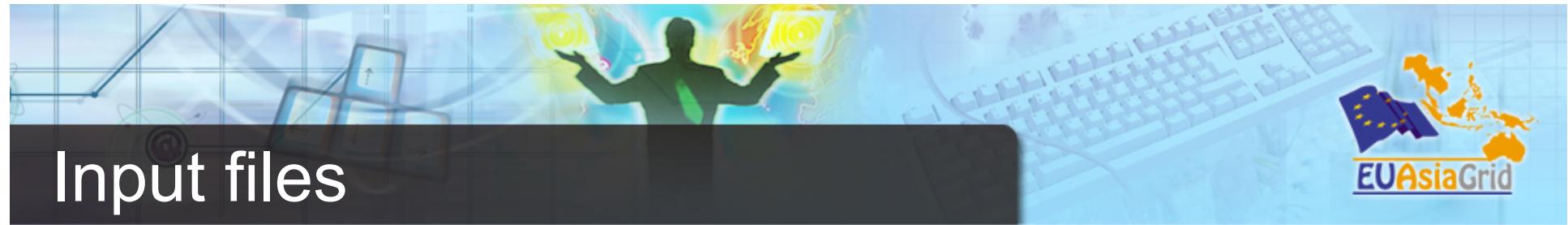
Passwd: GridKUAXX

where XX=01.,,35

## Start your proxy

\$ voms-proxy-init –voms gilda

PassPhrase : KUALALUMPUR



## Input files

- Make docking directory in your home

```
$ mkdir docking
```

- Copy input files in your home and untar archive

```
$ cp /tmp/data_docking/* ./docking
```

```
$ cd docking
```

```
$ tar zxvf ligands.tar.gz
```

```
$ tar zxvf protein.tar.gz
```

```
$ tar zxvf dpf.tar.gz
```

- Make a single archive

```
$ tar cvzf input.tar.gz *.dpf *.pdbqt 1hvr_rec*
```



# Exercice 2

## Glite SUBMISSION

# Store input data

## Select your storage element

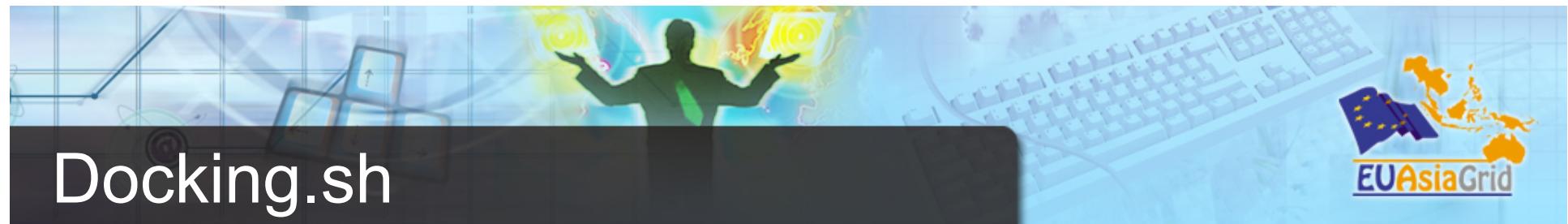
```
$ lcg-infosites --vo gilda se
```

Avail Space(Kb)	Used Space(Kb)	Type	SEs
30079879	8271579	n.a	gilda-02.pd.infn.it
53044869	5367469	n.a	vega-se.ct.infn.it
97179521	6516678	n.a	dgt02.ui.savba.sk
51184138	7228200	n.a	sirius-se.ct.infn.it
8768190	10937157	n.a	se.hpc.iit.bme.hu
295128383	98710449	n.a	aliserv6.ct.infn.it
2416637181	71077003	n.a	se1-egee.srce.hr
1097655286	1856340	n.a	gilda-se.rediris.es
60866711	6418562	n.a	iceage-se-01.ct.infn.it
1060343301	39033059	n.a	fn2.hpcc.sztaki.hu

## Copy and register data on grid

```
$ lcg-cr --vo gilda -d aliserv6.ct.infn.it -l Ifn:/grid/gilda/users/kualalumpurXX  
/input.tar.gz file:/home/kualalumpurXX/docking/input.tar.gz
```

```
$ lcg-cr --vo gilda -d aliserv6.ct.infn.it -l Ifn:/grid/gilda/users/kualalumpurXX  
/autodock4.tar.bz2 file:/home/kualalumpurXX/docking/autodock4.tar.bz2
```



# Docking.sh



## □ Edit the script and change modify

```
#!/bin/bash

export LCG_GFAL_INFOSYS=glite-rb.ct.infn.it:2170
export LFC_HOST=lfc-gilda.ct.infn.it
export LCG_CATALOG_TYPE="lfc"

#retrieve the software
lcg-cp --vo gilda lfn:/grid/gilda/users/kualalumpur26/docking/autodock4.tar.bz2 file:`pwd`/autodock4.tar.bz2
tar -jxf autodock4.tar.bz2

#retrieve the input data
lcg-cp --vo gilda lfn:/grid/gilda/users/kualalumpur26/docking/input.tar.gz file:`pwd`/input.tar.gz
tar -zxf input.tar.gz

#run autodock4
ulimit -s unlimited
for i in `ls *.dpf`
do
./autodock4 -p $i -l $i.dlg
done

#save the results back
for i in `ls *.dlg`
do
lcg-cr --vo gilda file:`pwd`/$i -d iceage-se-01.ct.infn.it -l lfn:/grid/gilda/users/kualalumpur26/docking/$i
done
```



# Docking.jdl

```
Type = "Job";
JobType = "Normal";
Executable = "/bin/bash";
StdOutput = "job.out";
StdError = "job.err";
InputSandbox = {"docking.sh"};
OutputSandbox = {"job.err","job.out"};
Arguments = "docking.sh";
ShallowRetryCount = 0;
```



# Job submission



- ❑ Create a delegation ID to the WMS by creating a delegation identifier using your username

```
$ glite-wms-job-delegate-proxy -d kualalumpurXX
```

- ❑ Submit a job

```
$ glite-wms-job-submit -d kualalumpurXX docking.jdl
```

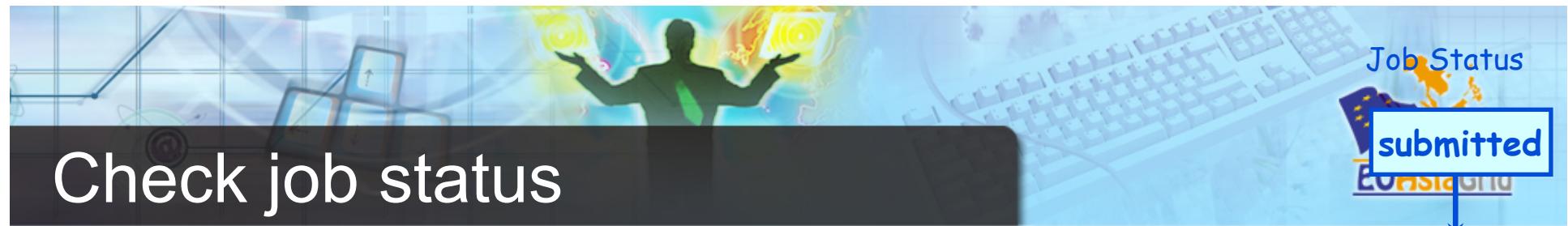
```
Connecting to the service https://gilda-wms-02.ct.infn.it:7443/glite_wms_wmproxy_server
```

```
===== glite-wms-job-submit Success =====
```

```
The job has been successfully submitted to the WMProxy  
Your job identifier is:
```

```
https://gilda-lb-01.ct.infn.it:9000/XfDhzPvFVuthZLKgTcszqg
```

```
=====
```



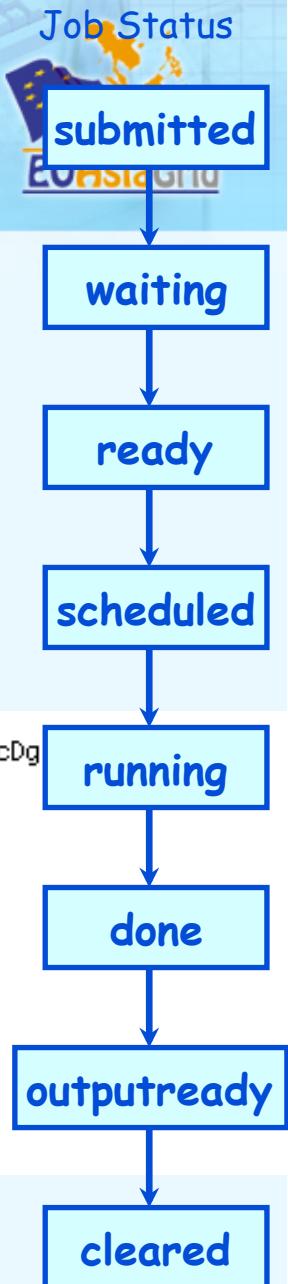
# Check job status

## Check your job status

```
$ glite-wms-job-status https://gilda-lb-01.ct.infn.it:9000  
/ST7NwtFLDUOLLsNQmo9BwA
```

```
[varennna01@glite-tutor docking]$ glite-wms-job-status https://gilda-lb-01.ct.infn.it:9000/pDYyXN5iKooZrZjedsocDg
```

```
*****  
BOOKKEEPING INFORMATION:  
  
Status info for the Job : https://gilda-lb-01.ct.infn.it:9000/pDYyXN5iKooZrZjedsocDg  
Current Status: Running  
Status Reason: Job successfully submitted to Globus  
Destination: ce1-egee.srce.hr:2119/jobmanager-sge-prod  
Submitted: Tue May 12 11:59:24 2009 CEST  
*****
```





## Retrieve your results

- Retrieve you dlg from grid in your home

```
$ for i in `cat list_dlg`; do lcg-cp --vo gilda lfn:/grid/gilda/users  
/kualalumpurXX/docking/$i file:`pwd`/$i ; done
```

- Results analysis...

# Results analysis

- Extract and rank Binding Energy for best conformations
- \$ ./energy.sh

```

MODEL      7
USER Run = 7
USER Cluster Rank = 1
USER Number of conformations in this cluster = 7
USER
USER RMSD from reference structure      = 37.600 Å
USER
USER Estimated Free Energy of Binding   = -7.22 kcal/mol [==(1)+(2)+(3)-(4)]
USER Estimated Inhibition Constant, Ki  = 5.11 μM (micromolar) [Temperature = 298.15 K]
USER
USER (1) Final Intermolecular Energy     = -7.49 kcal/mol
USER    vdW + Hbond + desolv Energy       = -4.52 kcal/mol
USER    Electrostatic Energy              = -2.98 kcal/mol
USER (2) Final Total Internal Energy     = -0.02 kcal/mol
USER (3) Torsional Free Energy          = +0.27 kcal/mol
USER (4) Unbound System's Energy         = -0.02 kcal/mol
USER
USER
USER DPF = 4002723_1_1hvr_rec.dpf
USER NEWDPF move 4002723_1.pdbqt
USER NEWDPF about -3.207000 -1.889600 -2.574900
USER NEWDPF trans0 -10.227483 16.611496 38.661568
USER NEWDPF axisangle0 -0.769077 0.345317 -0.537845 115.819078
USER NEWDPF quaternion0 -0.651570 0.292556 -0.455668 0.531258
USER NEWDPF ndihe 1
USER NEWDPF dihe0 -128.36
USER
USER
ATOM      x        y        z        vdw      Elec      q      RMS
ATOM    1 C1  lig d -8.964  14.775 32.507 -0.34 -0.00 +0.033 37.600
ATOM    2 C2  lig d -9.980  14.610 33.451 -0.29 -0.01 +0.039 37.600
ATOM    3 C3  lig d -11.461 16.873 32.201 -0.14 -0.02 +0.054 37.600
ATOM    4 C4  lig d -10.451 16.257 31.234 -0.26 -0.00 +0.003 37.600
ATOM    5 C5  lig d -10.355 16.999 30.026 -0.32 -0.00 +0.003 37.600
ATOM    6 C6  lig d -9.226 15.683 31.408 -0.27 -0.01 +0.043 37.600
ATOM    7 C7  lig d -11.211 15.248 33.302 -0.26 -0.01 +0.069 37.600
ATOM    8 C8  lig d -9.098 16.777 29.503 -0.26 -0.01 +0.020 37.600
ATOM    9 C9  lig d -11.348 17.866 29.335 -0.27 -0.47 +0.273 37.600
ATOM   10 C10 lig d -8.559 17.373 28.251 -0.27 -0.07 +0.085 37.600
ATOM   11 C11 lig d -9.714 17.894 27.384 -0.27 -0.33 +0.236 37.600
ATOM   12 N1  lig d -8.421 15.940 30.340 -0.12 +0.10 -0.358 37.600
ATOM   13 N2  lig d -10.684 18.677 28.236 -0.36 +0.19 -0.071 37.600
ATOM   14 H   lig d -7.487 15.591 30.189 +0.87 -0.01 +0.166 37.600
ATOM   15 H   lig d -11.488 19.083 27.570 -0.23 -1.20 +0.277 37.600
ATOM   16 H30 lig d -10.135 19.427 28.703 -0.22 -1.14 +0.277 37.600
ATOM   17 O1  lig d -12.183 15.065 34.239 -0.42 +0.07 -0.356 37.600
ATOM   18 C12 lig d -11.818 15.287 35.601 -0.30 -0.05 +0.210 37.600
TER

```

```

[kualalumpur30@glite-tutor results_analysis]$ ./energy.sh
-10.35 1hvr_lig.dpf.dlg
-7.19 4002723_1_1hvr_rec.dpf.dlg
-6.59 4001711_1_1hvr_rec.dpf.dlg
-5.68 4001115_1_1hvr_rec.dpf.dlg
-5.31 4000329_1_1hvr_rec.dpf.dlg
-4.79 4000321_1_1hvr_rec.dpf.dlg

```

« The lowest is the best »

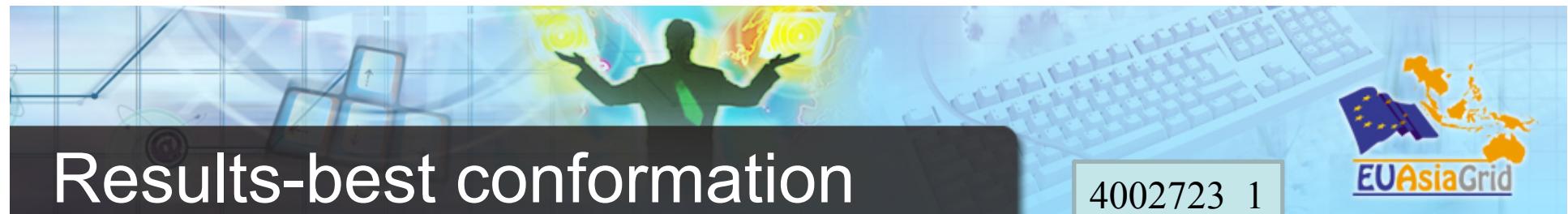
# Results analysis

- Select best conformations and extract its coordinates from dlg file

Copy the « best dlg » (having the lowest energy) in  
/home/kualalumpurXX/docking/results\_analysis

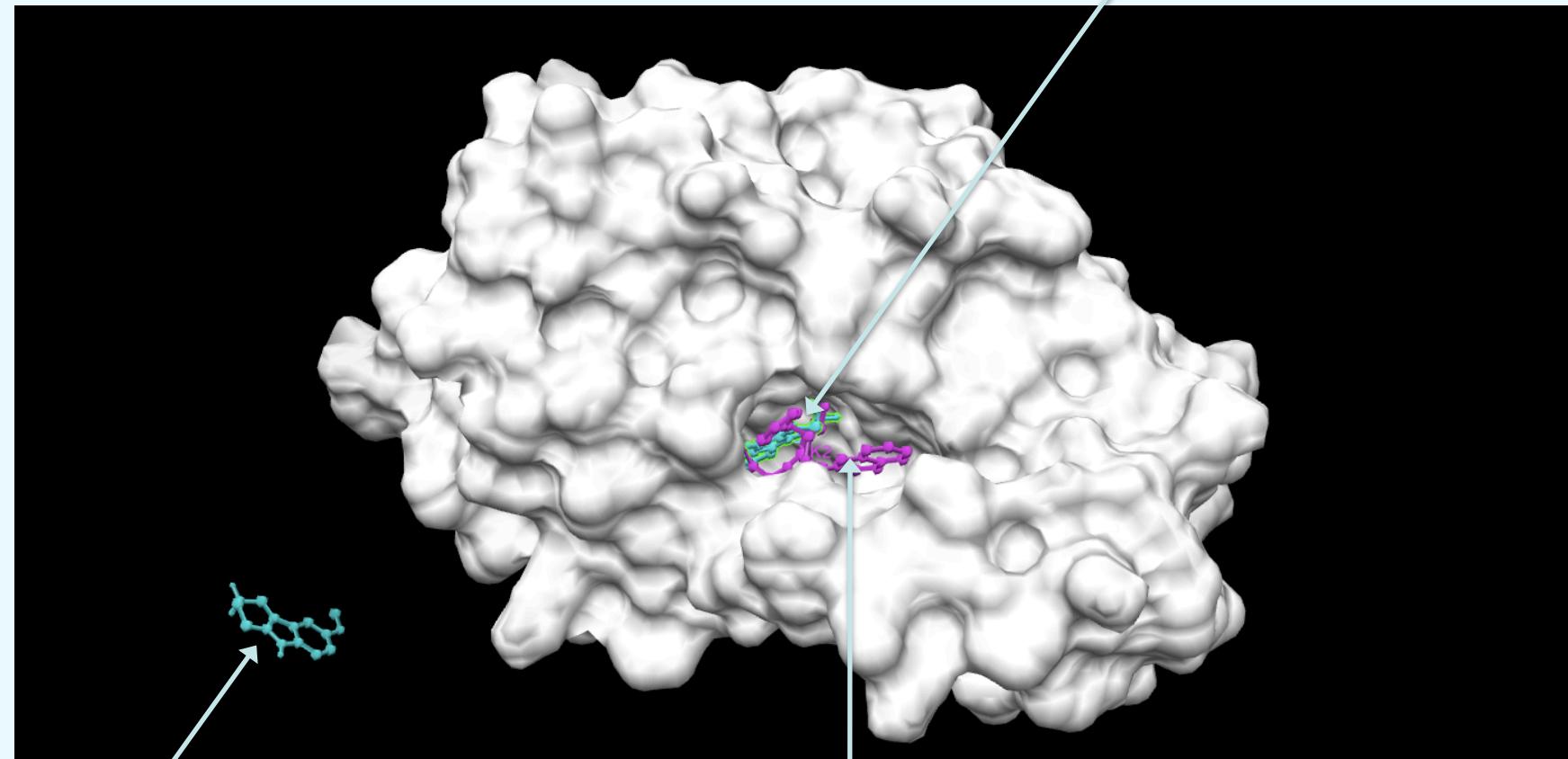
```
$ ./get_best_docking.sh 4002723_1_1hvr_rec.dpf.dlg
```

- You obtain the docked conformation  
4002723\_1\_1hvr\_rec.dpf.pdbq



# Results-best conformation

4002723\_1  
docked



4002723\_1  
23791

1hvr\_lig



## Exercice 2

# WISDOM SUBMISSION

# Wisdom Production Env



Client Services

Database Service

Transfer Manager

FTP

HTTP

Data Manager

SE

EGEE

SE

OSG

Job Manager

Job Submitter

Data Management APIs

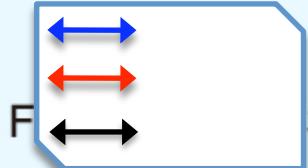
CE

WISDOM Information System

AMGA

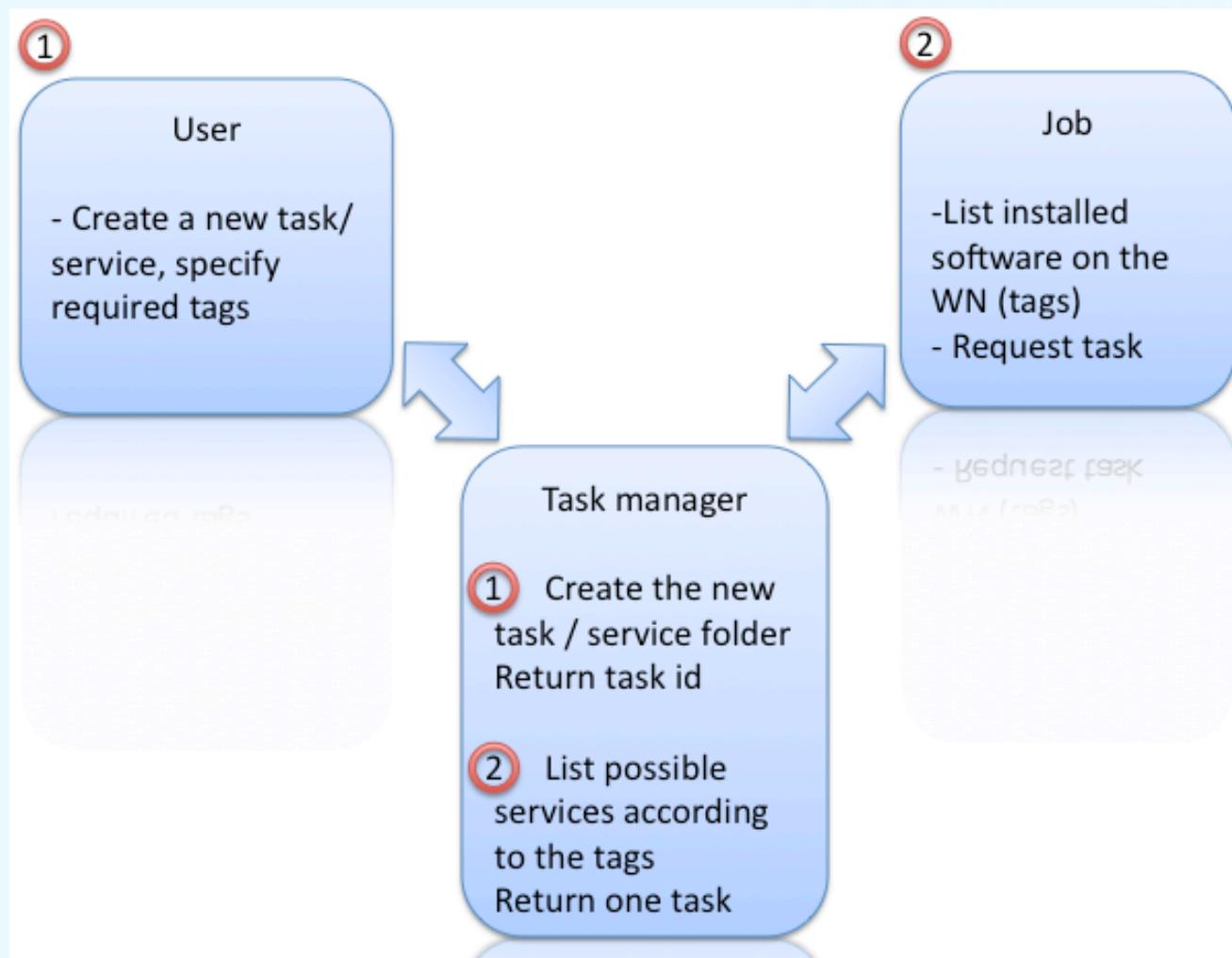
Task Manager

Tasks Management APIs

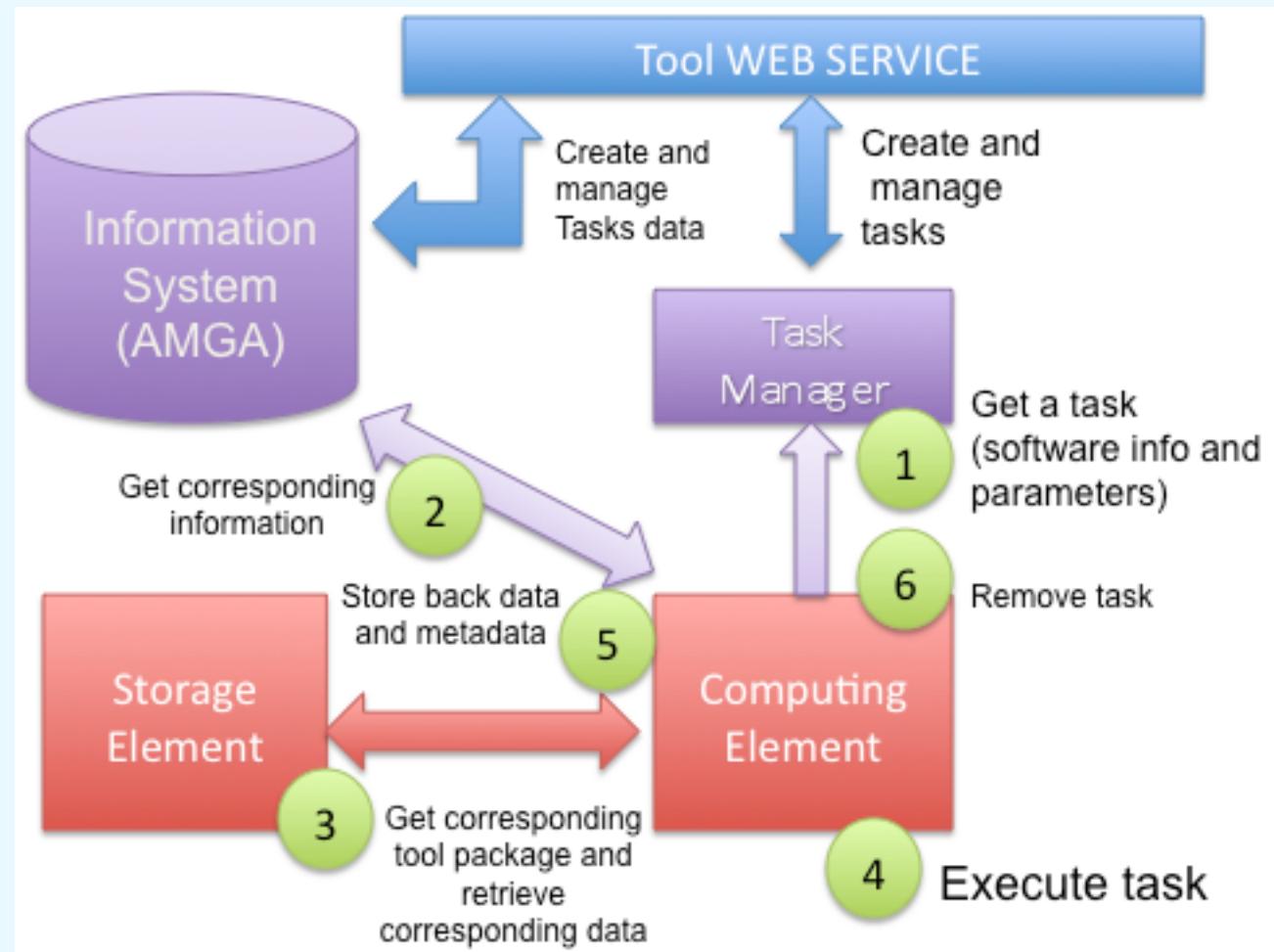


3791

# Task Manager Interactions



# Task Submission process

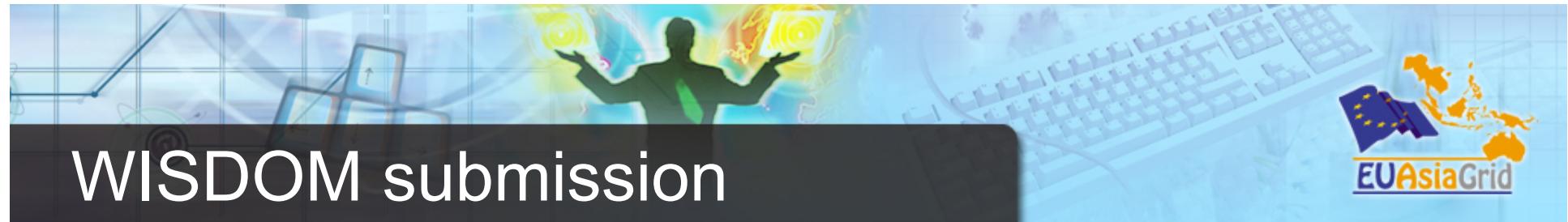




# WISDOM submission



- Store a tar.gz of all the necessary files on the grid
  - Already done in the previous exercise
- Service DOCKING.tar.gz is already deployed (/grid/gilda /users/services/DOCKING.tar.gz)
- Create a task
  - Use the script createTask
  - ./createTask -s {name\_service} -u {nom\_user} -a {arguments}
    - With:
      - *Nom\_service=DOCKING*
      - *Nom\_utilisateur=varennaxX*
      - *Arguments=name of the archive without tar.gz: input*
      - *./createTask -s DOCKING -u varennaxX -a input*
  - Result obtain:
    - {name\_service}.{name\_user}.{number}  
Example: DOCKING.varennaxX.7



# WISDOM submission

## □ Check the status of the task

- Use the script getStatus
  - *Put in parameter the TaskId*
  - *Example: ./getStatus DOCKING.varennaxX.7*
  - *3 possible results: waiting, running, done*
  - *You would be able to see the agents running on the following web site:  
<http://amga02.lpc-rd.fr/plate-forme-bioInfo/>*

## □ Retrieve the results

- Results are stored in: /grid/gilda/users/varennaxX/docking/
- The name of the archive created corresponds to {name of the archive}\_result.tar.gz