

# Commanding c++ with python

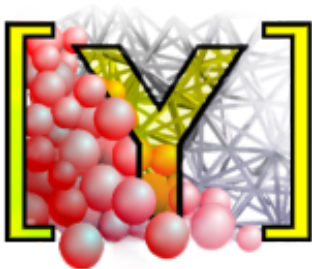
Václav Šmilauer  
(UJF Grenoble + CTU Prague)

Aussois, 9. 10. 2008

# History and motivation: tired of clicking?

Commanding  
c++ with  
python

Václav  
Šmilauer  
(UJF  
Grenoble +  
CTU Pra-  
gue)



- How to make computers do more work for you?
- ☕ Automation.
  - 🖱️ Load, run for a while, save simulation and quit.
  - 🖱️ Run preprocessor with some parameters.
- ☠️ Parametric study with mouse?

# More about python

Commanding  
c++ with  
python

Václav  
Šmilauer  
(UJF  
Grenoble +  
CTU Pra-  
gue)



👉 [www.python.org](http://www.python.org)

- Object-oriented scripting (interpreted) language.
- Easy to learn.
- 📖 Huge standard library.
- Excellent 📖 documentation including 📖 tutorial and 📖 books.
- “Easy” to 📖 embed in a c/c++ program.



Syntax note: Python uses brackets [...] for lists, {...} for dictionaries, (...) for tuples; code blocks are marked by indentation.

# Creating simulation in python

Commanding  
c++ with  
python

Václav  
Šmilauer  
(UJF  
Grenoble +  
CTU Pra-  
gue)

## Basic steps:

- 1 create and place bodies
- 2 create initializers and engines
- 3 set time step
- 4 save/run

# Manipulating c++ objects from python

Commanding  
c++ with  
python

Václav  
Šmilauer  
(UJF  
Grenoble +  
CTU Pra-  
gue)

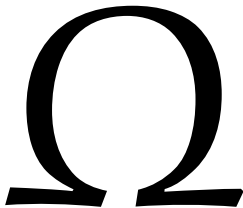
Yade root classes: `Body`, `Interaction`,  
`PhysicalAction`, `GeometricalModel`,  
`InteractingGeometry`,  
`PhysicalParameters`, `BoundingVolume`,  
`InteractionPhysics`,  
`InteractionGeometry`, `MetaEngine`,  
`StandAloneEngine`, `EngineUnit`

- Creating an instance of `root` c++ class in python.
- Instance of `any derived class`.
- Setting/getting `basic` (“fundamental”) attributes.
- Setting/getting `attributes which are themselves Yade classes`.

# Omega elements

Commanding  
c++ with  
python

Václav  
Šmilauer  
(UJF  
Grenoble +  
CTU Pra-  
gue)



- **Omega()**.
  - engines
  - initializers
  - bodies
  - actions
  - interactions
- [scripts/simple-scene.py](#)

# Travaux pratiques

Commanding  
c++ with  
python

Václav  
Šmilauer  
(UJF  
Grenoble +  
CTU Pra-  
gue)

- 1 ➡ Create simple simulation
- 2 (intermezzo) ➡ defining functions, importing ➡ modules
- 3 ➡ Create more complex simulation (cylindrical packing)
- 4 ➡ Inspect simulation at runtime
- 5 Call python from c++ (PythonPeriodicRunner)
- 6 Complex example: ➡ uniaxial tension-compression test

# Thanks for you attention

Commanding  
c++ with  
python

Václav  
Šmilauer  
(UJF  
Grenoble +  
CTU Pra-  
gue)

Got  
questions



 [yade-users@lists.launchpad.net](mailto:yade-users@lists.launchpad.net)

 [yade.wikia.com](http://yade.wikia.com)