

Robot Model Development on Steroids

by Kai von Szadkowski





```
<?xml version="1.0"?>
<!-- created with Phobos 0.7 -->
<robot name="mantis">

  <link name="body/0">
    <visual name="visual_torso_left_socket">
      <origin xyz="0.0198386 0.082563 0.0100624" rpy="0 0 5e-07"/>
      <geometry>
        <mesh filename="../meshes/torso_left_socket.bobj" scale="1.0 1.0 1.0"/>
      </geometry>
      <material name="joint"/>
    </visual>
    <collision name="collision_torso_left_socket">
      <origin xyz="0.0198387 0.082563 0.0100624" rpy="0 0 5e-07"/>
      <geometry>
        <box size="0.1397003 0.2348451 0.1699999"/>
      </geometry>
    </collision>
  </link>

  <link name="body/1">
    <visual name="visual_torso_right_socket">
      <origin xyz="0.0198383 -0.0825922 -0.0099376" rpy="0 0 5e-07"/>
      <geometry>
        <mesh filename="../meshes/torso_right_socket.bobj" scale="1.0 1.0 1.0"/>
      </geometry>
      <material name="joint"/>
    </visual>
    <collision name="collision_torso_right_socket">
      <origin xyz="0.0192199 -0.0990479 -0.01" rpy="0 0 5e-07"/>
      <geometry>
        <box size="0.1397003 0.2348451 0.1699998"/>
      </geometry>
    </collision>
  </link>

  <link name="body/2">
    <inertial>
      <origin xyz="0.0099378 -1.14e-05 -0.1320033" rpy="1e-07 0 0"/>
      <mass value="2.3"/>
      <inertia ixx="0.1088414" ixy="0" ixz="0" iyy="0.1248771" iyz="-1e-07"
        izz="0.0271677"/>
    </inertial>
    <visual name="visual_hip_left">
      <origin xyz="0.0099378 -1.14e-05 -0.1320033" rpy="1.5707977 8e-07 1.5707963"/>
      <geometry>
        <mesh filename="../meshes/hip_right.bobj" scale="1.0 1.0 1.0"/>
      </geometry>
      <material name="limb"/>
    </visual>
  </link>
</robot>
```



1. Introduction
2. Background
3. Methodology
4. Results
5. Discussion
6. Conclusion
7. References
8. Appendix
9. Glossary
10. Index
11. Bibliography
12. Acknowledgments
13. Contact Information
14. Disclaimer
15. Copyright
16. Privacy Policy
17. Terms of Service
18. About Us
19. Mission Statement
20. Vision Statement
21. Core Values
22. History
23. Leadership
24. Team
25. Partners
26. Clients
27. Products
28. Services
29. Pricing
30. Support
31. Feedback
32. News
33. Events
34. Awards
35. Certifications
36. Licenses
37. Trademarks
38. Patents
39. Intellectual Property
40. Legal
41. Compliance
42. Security
43. Risk Management
44. Quality Assurance
45. Sustainability
46. Social Responsibility
47. Environmental Impact
48. Community Engagement
49. Stakeholder Relations
50. Governance
51. Ethics
52. Transparency
53. Accountability
54. Integrity
55. Honesty
56. Fairness
57. Respect
58. Inclusivity
59. Diversity
60. Equality
61. Justice
62. Equity
63. Access
64. Affordability
65. Usability
66. Reliability
67. Durability
68. Safety
69. Health
60. Well-being
71. Happiness
72. Satisfaction
73. Engagement
74. Loyalty
75. Retention
76. Churn
77. Conversion
78. Growth
79. Innovation
80. Creativity
81. Collaboration
82. Communication
83. Coordination
84. Cooperation
85. Consensus
86. Consistency
87. Continuity
88. Clarity
89. Conciseness
90. Comprehensibility
91. Comprehensiveness
92. Completeness
93. Correctness
94. Credibility
95. Currency
96. Currentness
97. Currency
98. Currency
99. Currency
100. Currency



What is Phobos?

- Blender AddOn
- free & open source

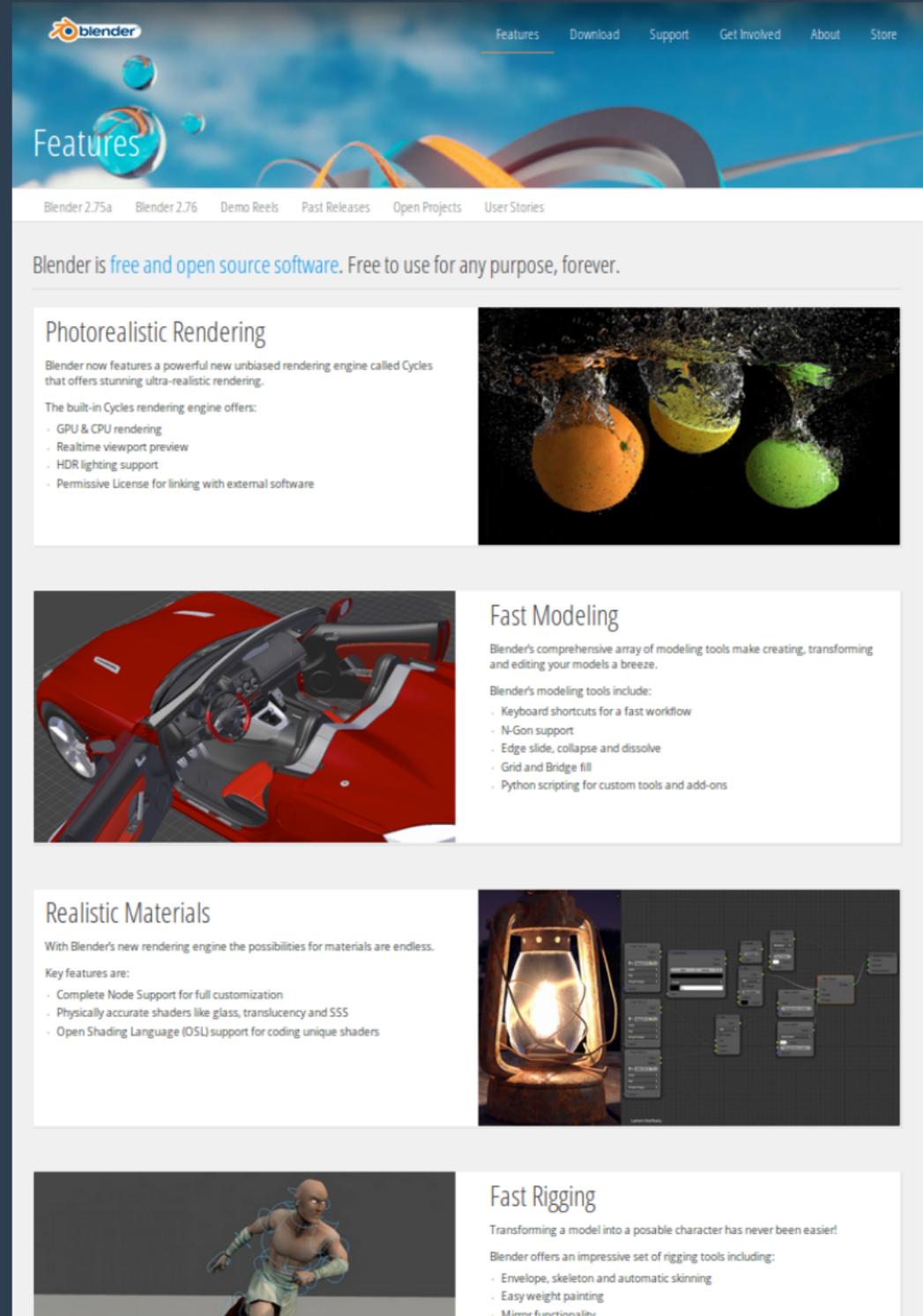
<https://github.com/rock-simulation/phobos>

- created as a tool for  **MARS**

<https://github.com/rock-simulation/mars>

Blender

- comprehensive 3D modelling and rendering tool
- editing tools for:
 - geometry
 - materials
 - animation "armatures"
 - various file formats
 - rendering and video rendering
- free & open source
- all major platforms
- Python scripting API



The screenshot shows the Blender website homepage. At the top, there is a navigation bar with the Blender logo and links for Features, Download, Support, Get Involved, About, and Store. Below the navigation bar, there is a large banner with the word "Features" and a background image of a 3D rendered scene. Underneath the banner, there is a section titled "Blender is free and open source software. Free to use for any purpose, forever." followed by several feature highlights, each with a title, a brief description, a list of key features, and a corresponding image.

Blender 2.75a | Blender 2.76 | Demo Reels | Past Releases | Open Projects | User Stories

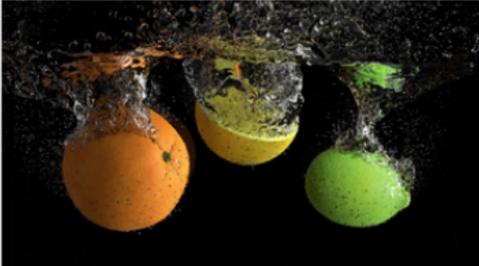
Blender is [free and open source software](#). Free to use for any purpose, forever.

Photorealistic Rendering

Blender now features a powerful new unbiased rendering engine called Cycles that offers stunning ultra-realistic rendering.

The built-in Cycles rendering engine offers:

- GPU & CPU rendering
- Realtime viewport preview
- HDR lighting support
- Permissive License for linking with external software



Fast Modeling

Blender's comprehensive array of modeling tools make creating, transforming and editing your models a breeze.

Blender's modeling tools include:

- Keyboard shortcuts for a fast workflow
- N-Gon support
- Edge slide, collapse and dissolve
- Grid and Bridge fill
- Python scripting for custom tools and add-ons

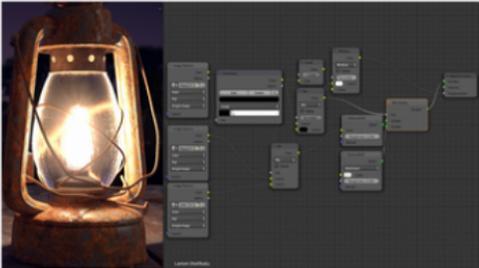


Realistic Materials

With Blender's new rendering engine the possibilities for materials are endless.

Key features are:

- Complete Node Support for full customization
- Physically accurate shaders like glass, transluency and SSS
- Open Shading Language (OSL) support for coding unique shaders



Fast Rigging

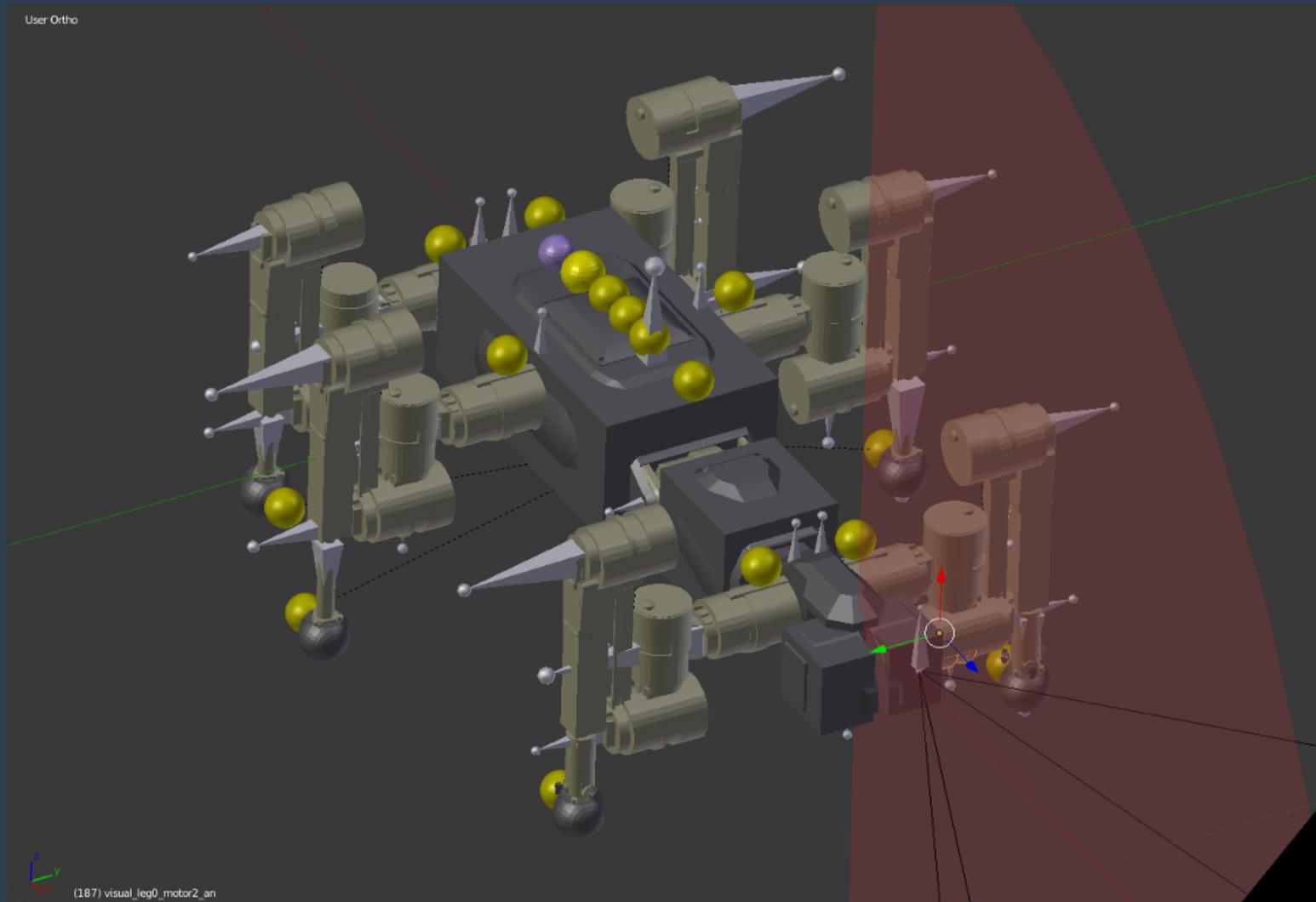
Transforming a model into a posable character has never been easier!

Blender offers an impressive set of rigging tools including:

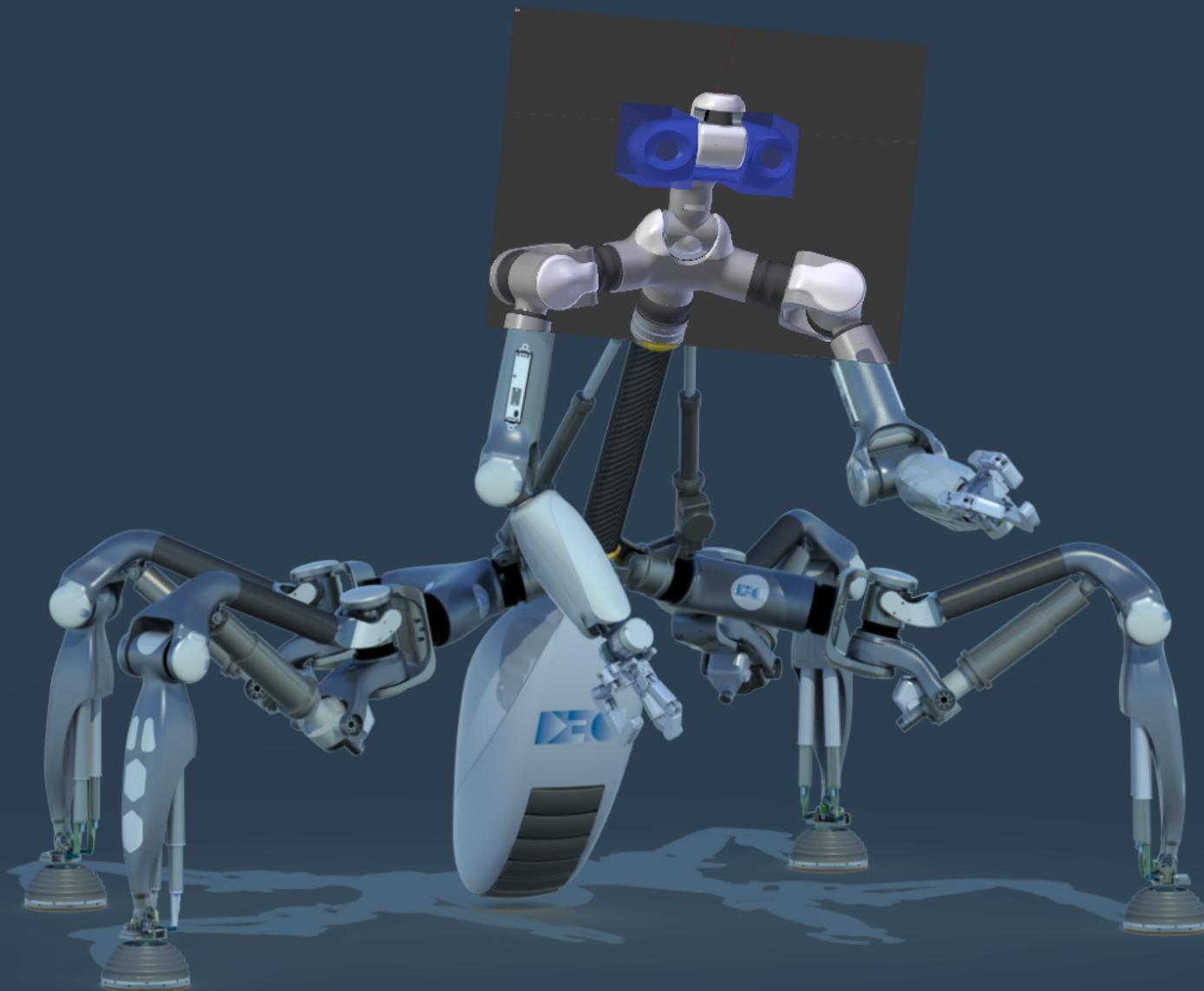
- Envelope, skeleton and automatic skinning
- Easy weight painting
- Mirror functionality

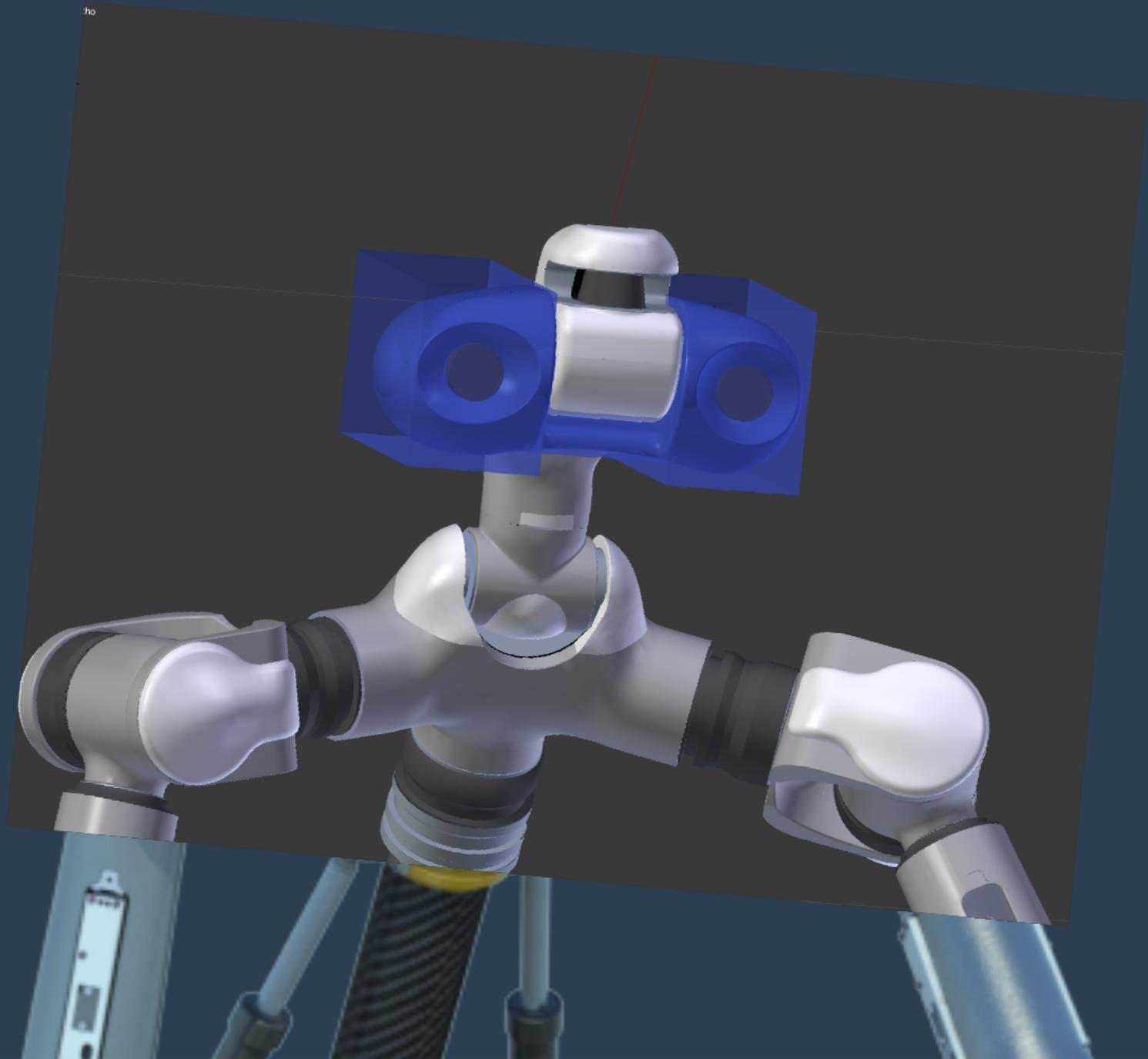


URDF vs. Blender

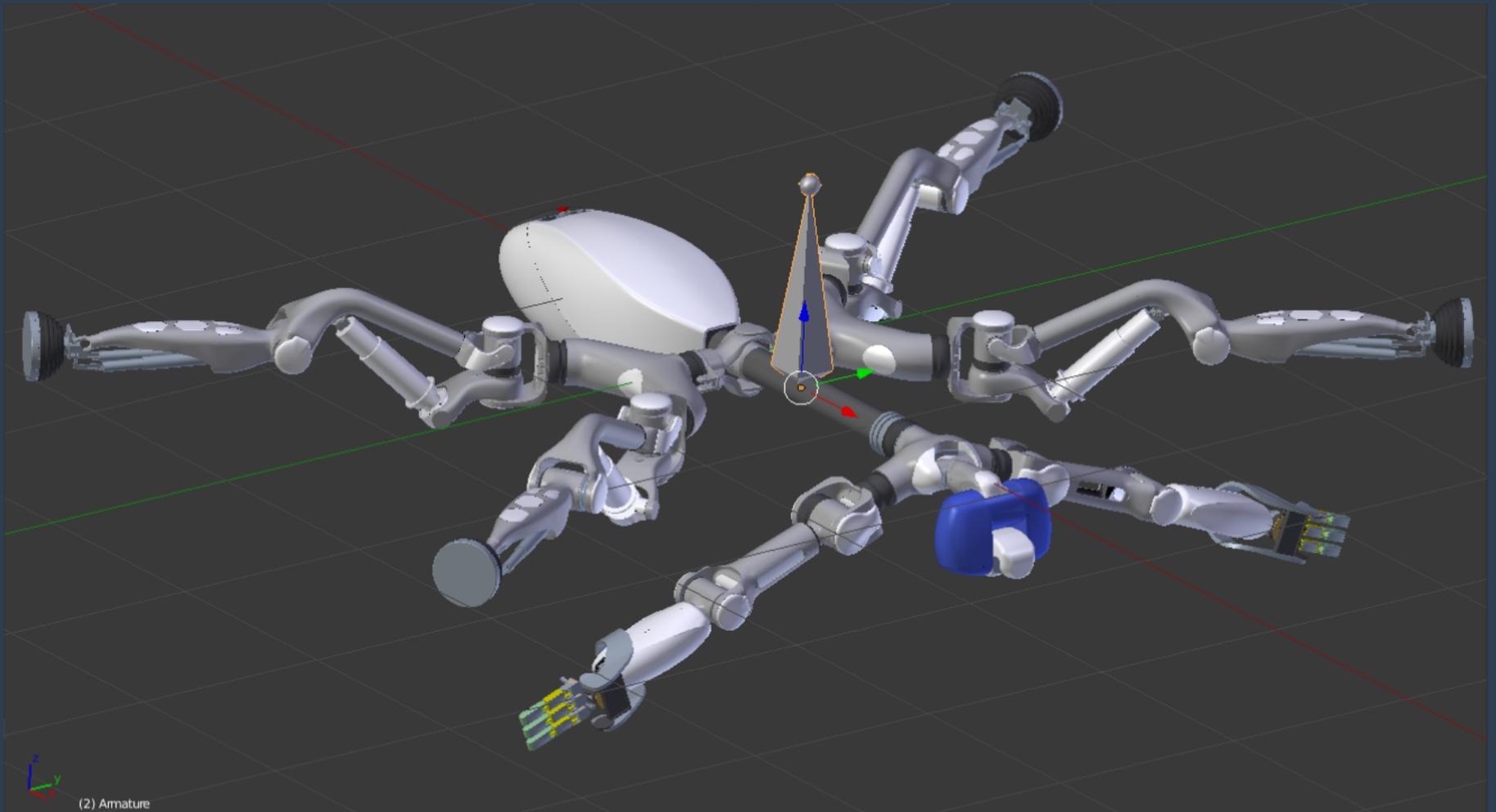


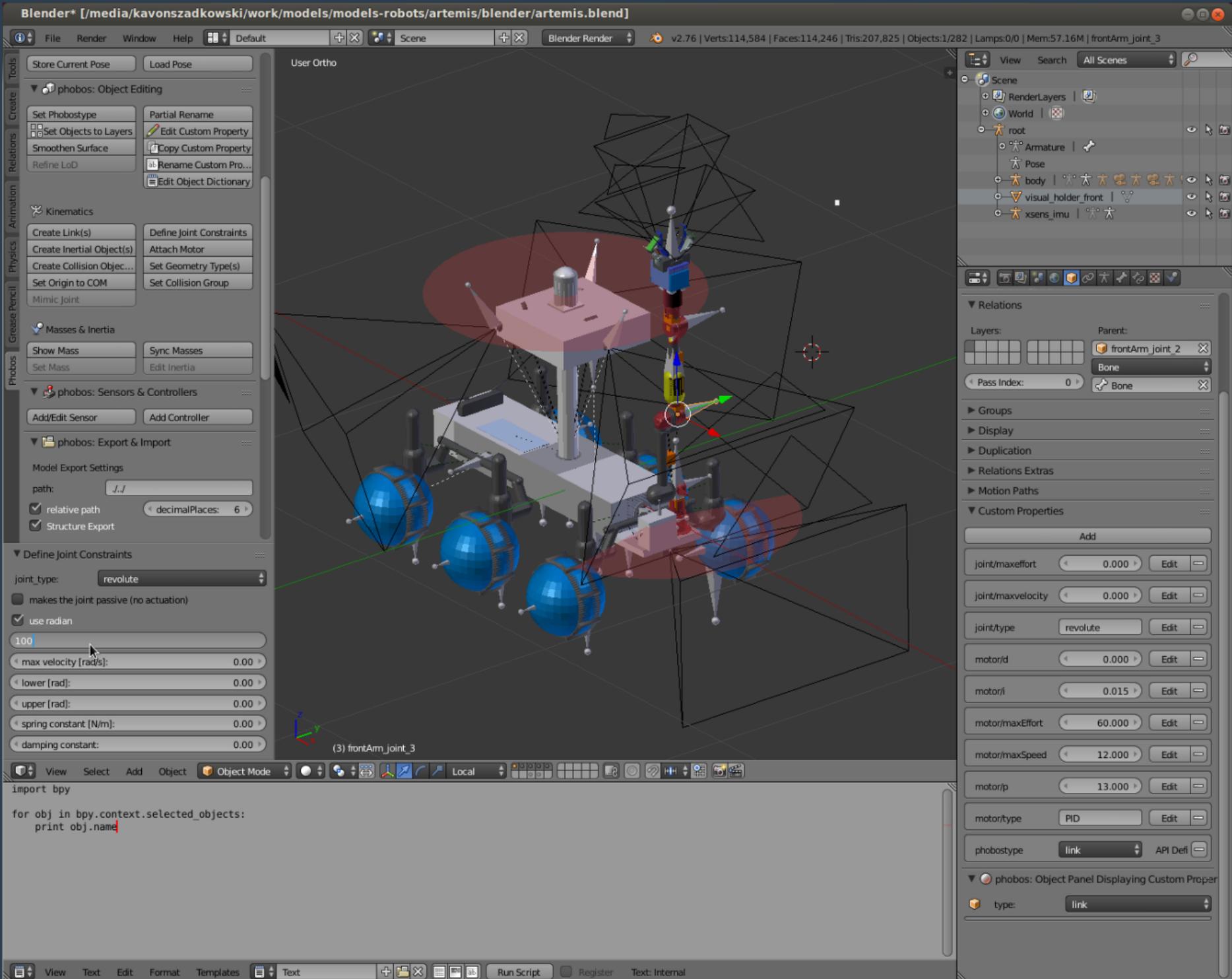
WYSIWYG





Importing from CAD





Tools Store Current Pose Load Pose

phobos: Object Editing

- Set Phobostype
- Partial Rename
- Set Objects to Layers
- Edit Custom Property
- Smoothen Surface
- Copy Custom Property
- Refine LoD
- Rename Custom Pro...
- Edit Object Dictionary

Kinematics

- Create Link(s)
- Define Joint Constraints
- Create Inertial Object(s)
- Attach Motor
- Create Collision Objec...
- Set Geometry Type(s)
- Set Origin to COM
- Set Collision Group
- Mimic Joint

Masses & Inertia

- Show Mass
- Sync Masses
- Set Mass
- Edit Inertia

phobos: Sensors & Controllers

- Add/Edit Sensor
- Add Controller

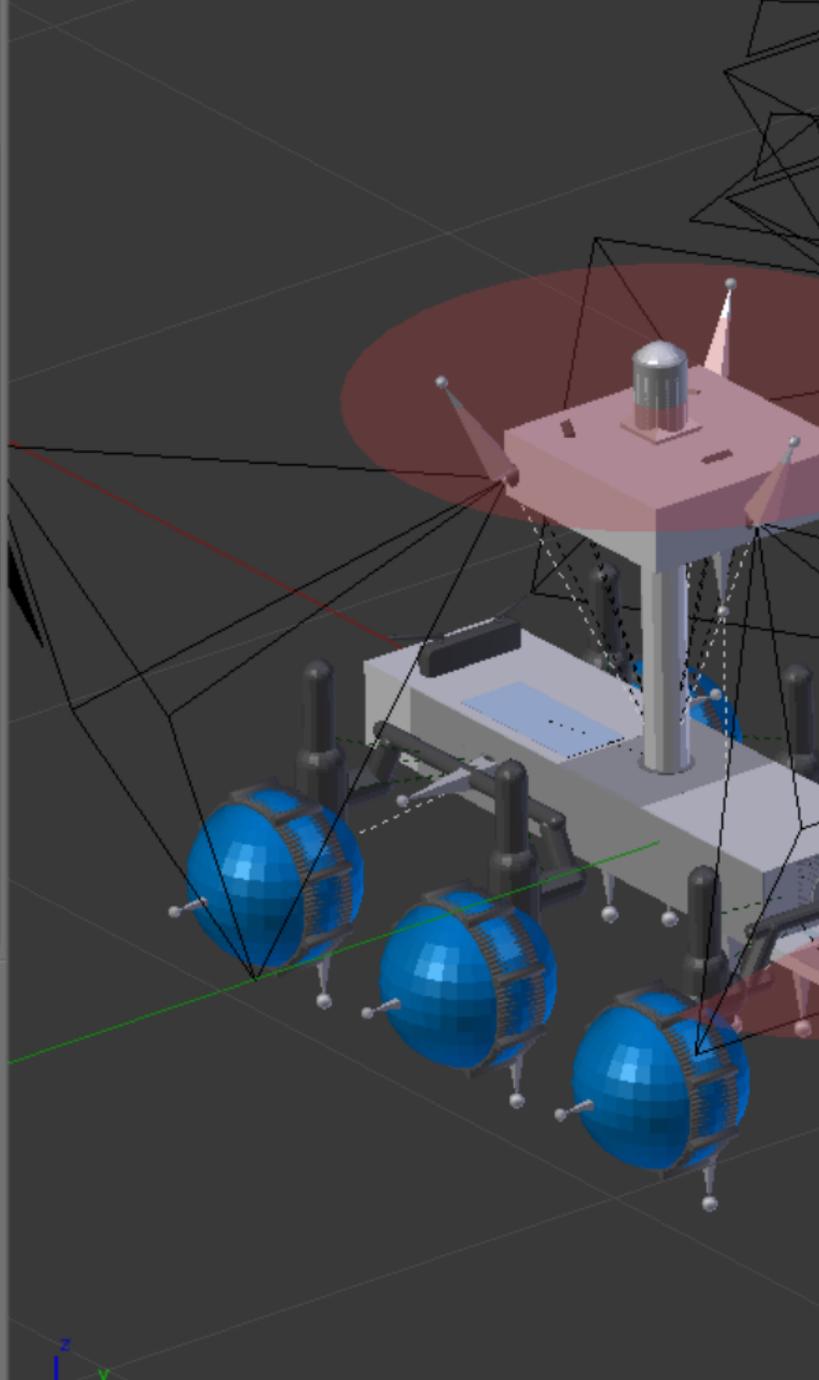
phobos: Export & Import

- Model Export Settings
- path: ./
- relative path
- decimalPlaces: 6
- Structure Export

Define Joint Constraints

- joint_type: revolute
- makes the joint passive (no actuation)
- use radian
- 100
- max velocity [rad/s]: 0.00
- lower [rad]: 0.00
- upper [rad]: 0.00
- spring constant [N/m]: 0.00
- damping constant: 0.00

User Ortho



(3) frontArm_joint_3

use radian

100

max velocity [rad/s]: 0.00

lower [rad]: 0.00

upper [rad]: 0.00

spring constant [N/m]: 0.00

damping constant: 0.00

(3) frontArm_joint_3

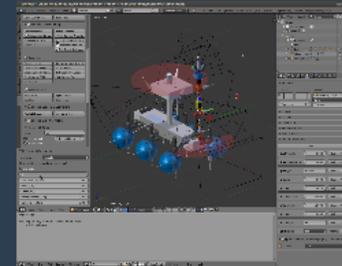
View Select Add Object Object Mode Local

```
import bpy  
  
for obj in bpy.context.selected_objects:  
    print obj.name
```

View Text Edit Format Templates Text Run Script

Importing from CAD

WYSIWYG



Object Tree

- allows matching of URDF and Blender model structures



URDF vs. Blender



Blender

- comprehensive 3D modelling and rendering tool
- editing tools for:
 - geometry
 - materials
 - animation "armatures"
 - various file formats
 - rendering and video rendering
- free & open source
- all major platforms
- Python scripting API



Robot dictionary

- Python dictionary
- decouples Blender model from im- and export
- allows generic definitions of components
- simplifies model debugging

SMURF

=

Supplementable

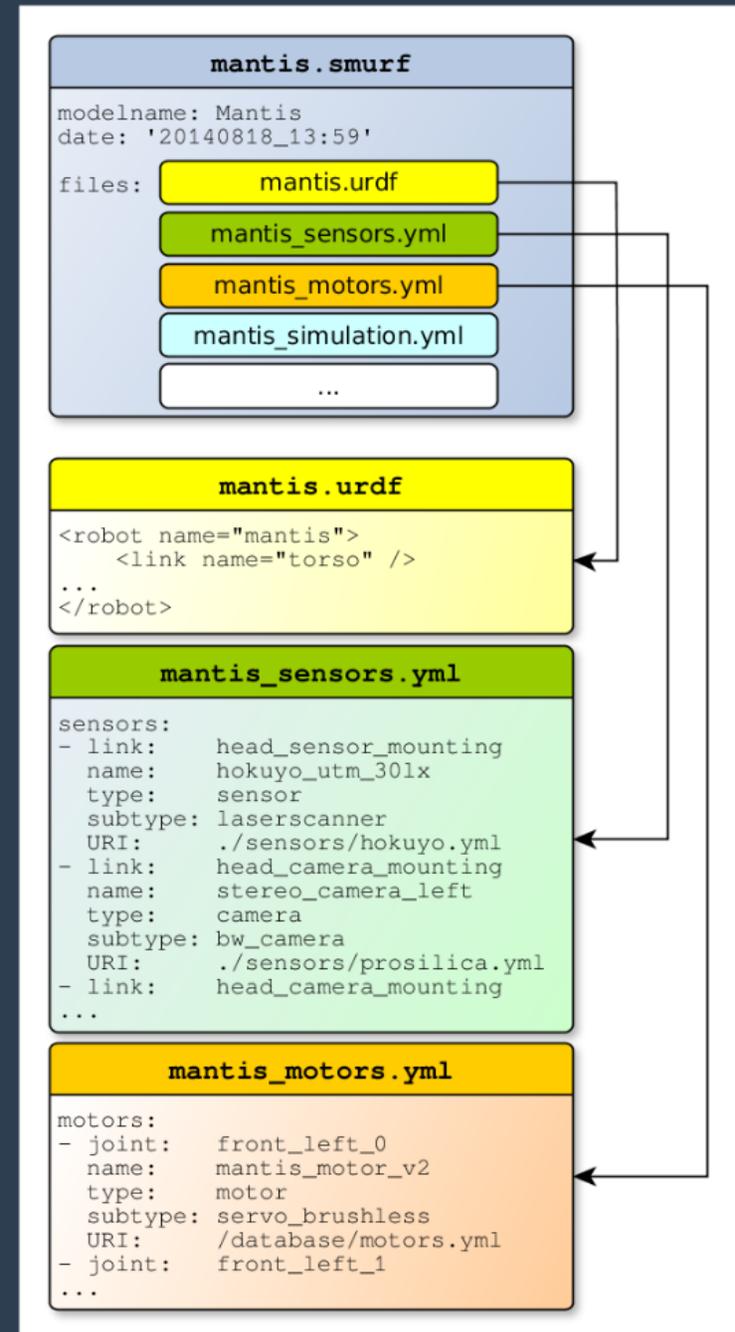
Mostly

Universal

Robot

Format

- URDF + YAML
- hierarchical structure



Models and scenes

- SMURF scenes:
 - multiple *smurfs* get exported as individual smurfs
 - URDF joints as scene anchors

Phobos as an upstream tool

Workflow:

- 1) create models in Phobos
- 2) export model to URDF & SMURF

Pro:

- all information in one place

Con:

- version control becomes difficult

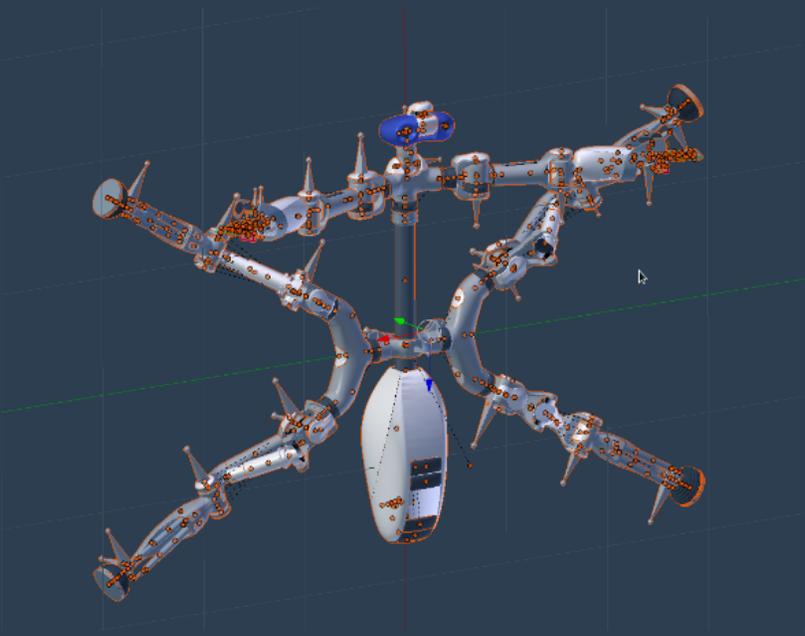


URDF

Models and scenes

- SMURF scenes:
 - multiple *smurfs* get exported as individual smurfs
 - URDF joints as scene anchors

SMURF



SMURF

=
Supplementable
Mostly
Universal
Robot
Format

- URDF + YAML
- hierarchical structure



Phobos as an upstream tool

Workflow:
1) create models in Phobos
2) export model to URDF & SMURF

Pro:
• all information in one place

Con:
• version control becomes difficult



Github

<https://github.com/rock-simulation/phobos>



Development



Documentation



Github Wiki

Github Pages:
Sphinx

The road ahead...

- URDF not yet supported 100%
- e.g. mimic joints
- dictionary and custom properties not fully specified (depend on SMURF's current status)
- code refactoring and debugging

GitHub

<https://github.com/rock-simulation/phobos>

The screenshot shows the GitHub repository page for `rock-simulation / phobos`. The repository is described as "An add-on for Blender allowing editing and exporting of robots for the MARS simulation." It has 1,037 commits, 33 branches, 7 releases, and 6 contributors. The current branch is `master`. The file list includes:

File	Description	Last Commit
definitions	tagged joints UNUSED	5 months ago
doc	reordered instructions for installation	10 months ago
models	Add phobos default materials to blender startup file	a month ago
source	retrack index	8 months ago
src	Merge pull request #91 from rock-simulation/mesh_inertia	a day ago
yaml	blendertools: included yaml-lib in marstools	2 years ago
.gitignore	clean .gitignore	5 months ago
COPYING	Phobos is now licensed under GNU LGPL	a year ago
COPYING.LESSER	Phobos is now licensed under GNU LGPL	a year ago

On the right side, there are links for Code, Issues (21), Pull requests (5), Wiki, Pulse, Graphs, and Settings. At the bottom, the SSH clone URL is `git@github.com:rc` and there is a button to Download ZIP.

Documentation

Robot Dictionary
Kai von Szadkowski edited this page on 11 Aug · 1 revision

To facilitate simple im- and export of robot models in various formats, Phobos uses a file-format-independent Python dictionary representation to store robot data. This dictionary representation is loosely based on URDF/SDF, with some added MARS' and Blender's own naming conventions. It is layed out in the following in YAML-notation:

```
model
- {joints}
  - {joint_1}
    - [axis]: (d, d, d)
    - child: str
    - {limits}:
      - lower: d
      - upper: d
      - velocity: d
      - effort: d
    - parent: str
    - type: str ("hinge", "continuous", "linear", "prismatic", "revolute")
  - {joint_2}
  - ...
- {links}
  - {link_1}
    - filename: str
    - [pose]: (d, d, d, d, d, d) #x, y, z, w, x, y, z
    - {visual}:
      - {visual_1}
        - [pose]: (d, d, d, d, d, d) #x, y, z, w, x, y, z
        - {material}:
          - name: str
          - [diffuseColor]: (d, d, d, d)
          - [ambientColor]: (d, d, d, d)
          - [emissionColor]: (d, d, d, d)
          - [specularColor]: (d, d, d, d)
          - transparency: d
        - {geometry}:
          - type: str ("box" | "sphere" | "cylinder" | "plane" | "me
          - radius: d #sphere
          - [size]: (d, d, d) #box
          - radius, height: d, d #cylinder
          - [size]: (d, d, d) #mesh
```

Github Pages: Sphinx

Phobos 0.5 documentation »

Quick search

Enter search terms or a module, class or function name.

Index

[A](#) | [B](#) | [C](#) | [D](#) | [E](#) | [F](#) | [G](#) | [H](#) | [I](#) | [J](#) | [L](#) | [M](#) | [N](#) | [O](#) | [P](#) | [R](#) | [S](#) | [T](#) | [U](#) | [V](#) | [W](#) | [X](#)

A

- [add_link](#) (phobos.sensors.AddSensorOperator attribute)
- [AddChainOperator](#) (class in phobos.misctools)
- [AddControllerOperator](#) (class in phobos.controllers)
- [AddGravityVector](#) (class in phobos.misctools)
- [AddLegacyControllerOperator](#) (class in phobos.controllers)
- [AddSensorOperator](#) (class in phobos.sensors)

B

- [bands](#) (phobos.sensors.AddSensorOperator attribute)
- [BatchEditPropertyOperator](#) (class in phobos.misctools)
- [bl_category](#) (phobos.phobosgui.PhobosExportPanel attribute)
 - [\(phobos.phobosgui.PhobosModelPanel attribute\)](#)
 - [\(phobos.phobosgui.PhobosObjectPanel attribute\)](#)
 - [\(phobos.phobosgui.PhobosPanel attribute\)](#)
 - [\(phobos.phobosgui.PhobosSenConPanel attribute\)](#)
 - [\(phobos.phobosgui.PhobosSettingsPanel attribute\)](#)
- [bl_context](#) (phobos.phobosgui.PhobosObjectPanel attribute)
- [bl_idname](#) (phobos.collision.CreateCollisionObjects attribute)
 - [\(phobos.collision.CreateCollisionObjects.SetCollisionGroupOperator attribute\)](#)
 - [\(phobos.controllers.AddControllerOperator attribute\)](#)
 - [\(phobos.controllers.AddLegacyControllerOperator attribute\)](#)
 - [\(phobos.exporter.ExportModelOperator attribute\)](#)

Github Wiki

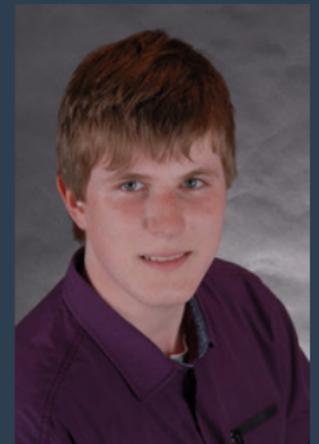
The road ahead...

- URDF not yet supported 100%
 - e.g.: mimic joints
- dictionary and custom properties not fully specified (depend on SMURF's current status)
- code refactoring and debugging

(contributors welcome)

Acknowledgements

- Contributors:
 - Stefan Rahms
 - Ole Schwiegert



Phobos

SMURF



Models and scenes

- VRML scenes
- Blender model, camera position, light, etc.
- URDF (Robot URDF)

Production pipeline is based

- on Blender
- on ROS (Robot Operating System)
- on Gazebo (Simulation)
- on ROS (Robot Operating System)

SMURF

- ROS
- Gazebo
- Blender
- URDF

Robot dictionary

- ROS
- Gazebo
- Blender
- URDF

Importing from CAD



URDF vs. Blender



Development



Documentation

- ROS
- Gazebo
- Blender
- URDF



What is Phobos?

- ROS
- Gazebo
- Blender
- URDF



Robot Model Development on Steroids

by Kai von Szadkowski



Acknowledgements

- Contributors
- Stefan Roth
- Ole Schwegert



The road ahead...

- ROS
- Gazebo
- Blender
- URDF

Contributors