Xacro can only take you so far

Go further with EmPy and ERB

Louise Poubel
@chapulinaBR

ROSCon 2022 - Kyoto

Outline

$ concepts
$ past projects
$ comparison #desplate

$ concepts

$ concept(‘Robot description formats’)

SDF URDF

$ concept('Robot

```
─┬─SCD½╢
 ├─ ½ÈÈ─ ´ Ş ─┼─ ½ÈÈ─
 ├─┬─SCD½─
 │ ├─ŵŵ─ ¸ Ş ¨ ¨ ´O─┼─ŵŵ─
 │ ├─ŵŶ─ ¸ Ş ─┼─ŵŶ─
 │ ├─ŵŴ ¸ Ş ─┼─ŵŴ
 │ ├─ŶŶ─ ¸ Ş ´YH¨ ─┼─ŶŶ─
 │ ├─ŶŴ ¸ Ş ─┼─ŶŴ
 │ ├─ŴŴ ¸ S ¸ ´E¨ ─┼─ŴŴ
 ├─┼─┬─SCD½─
─┼─┬─SCD½╢
```

$ concept(‘Templating’)

```
┬─┬ ︵ (°□°)︵ ┻━┻
  ├─┤ ╯°□°╯ ├─┤ ╯°□°╯ ┻━┻ ├─┤ ╯°□°╯
  ┬─┬ ︵ (°□°)︵
    ┬─┬ ﾉ( ゜-゜ﾉ) ├─┤ ╯°□°╯ ┻━┻ ┬─┬ ﾉ( ゜-゜ﾉ)
    ┬─┬ ﾉ( ゜-゜ﾉ) , ┻━┻ ┬─┬ ﾉ( ゜-゜ﾉ)
    ┬─┬ ﾉ( ゜-゜ﾉ) , ┻━┻ ┬─┬ ﾉ( ゜-゜ﾉ)
    ┬─┬ ﾉ( ゜-゜ﾉ) ├─┤ ╯°□°╯ ┻━┻ ┬─┬ ﾉ( ゜-゜ﾉ)
    ┬─┬ ﾉ( ゜-゜ﾉ) , ┻━┻ ┬─┬ ﾉ( ゜-゜ﾉ)
    ┬─┬ ﾉ( ゜-゜ﾉ) ├─┤ ╯°□°╯ ┻━┻ ┬─┬ ﾉ( ゜-゜ﾉ)
  ┻━┻ ┬─┬ ﾉ( ゜-゜ﾉ)
┻━┻ ┬─┬ ﾉ( ゜-゜ﾉ) ├─┤ ╯°□°╯
```

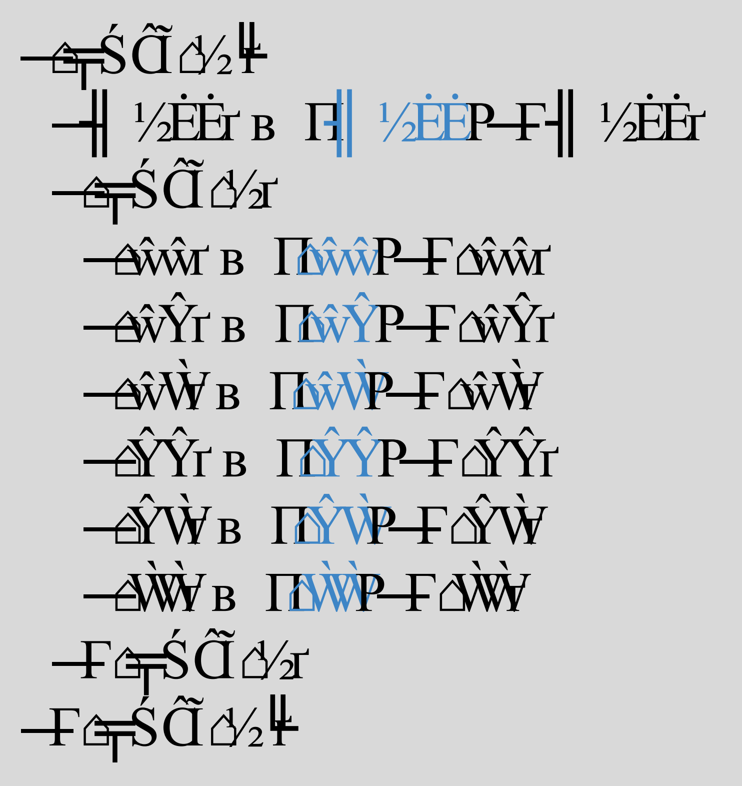
$ concept(‘Templating’)

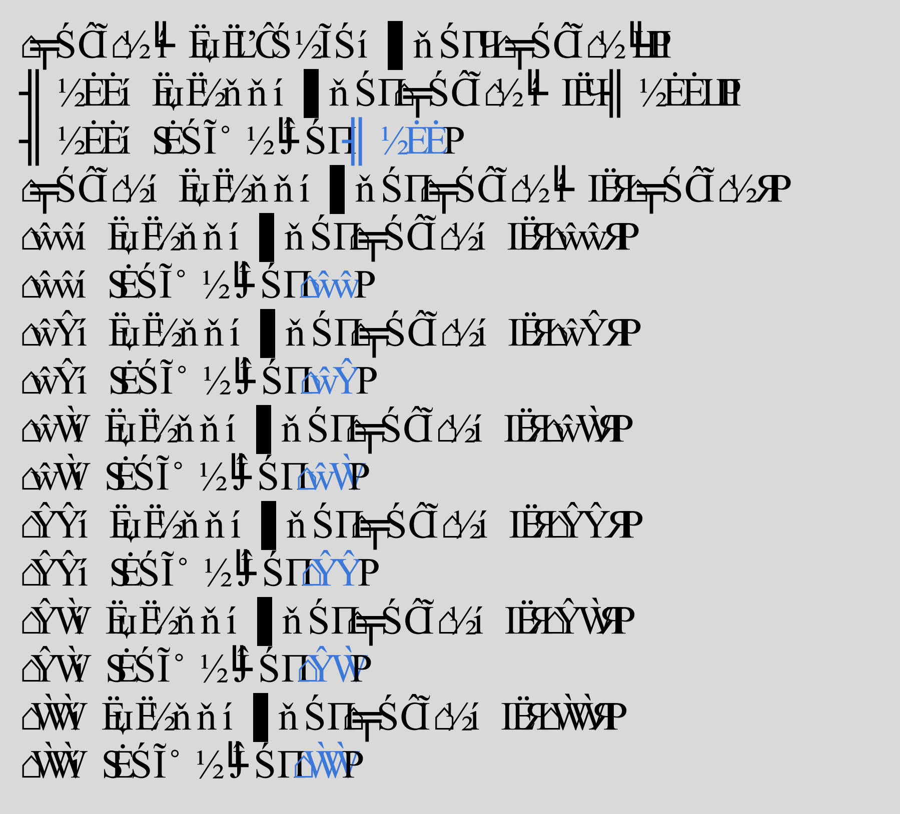
- Xacro (XML)
- EmPy (Python)
- ERB (Ruby)

$ concept(‘Templating’)

- Xacro (XML)
- EmPy (Python)
- ERB (Ruby)
- …

$ concept('Declarative vs Imperative')





$ concept(‘Generation stage’)

- Compile time
- Runtime (launch time)
- Separate step (manual)

$ projects

$ project(‘SRCSim’, 2017)

- ERB
- Separate step
- transformation matrix, randomization

github.com/osrf/srcsim/

$ project(‘CitySim’, 2017)

- ERB
- Separate step
- scripted animations

github.com/osrf/citysim/

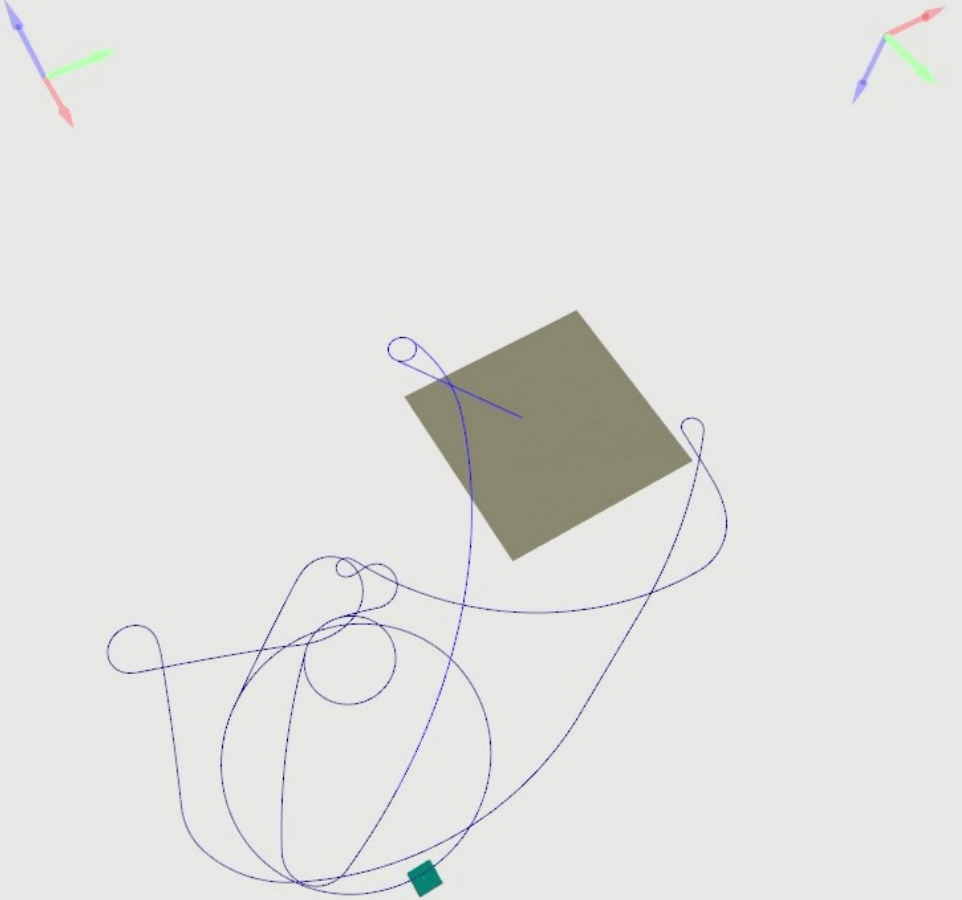
$ project(‘ServiceSim’, 2018)

- ERB
- Separate step
- modularity, YAML

github.com/osrf/servicesim

$ project(‘LRAUV’, 2022)

- EmPy
- Build step (CMake)
- spherical coordinates



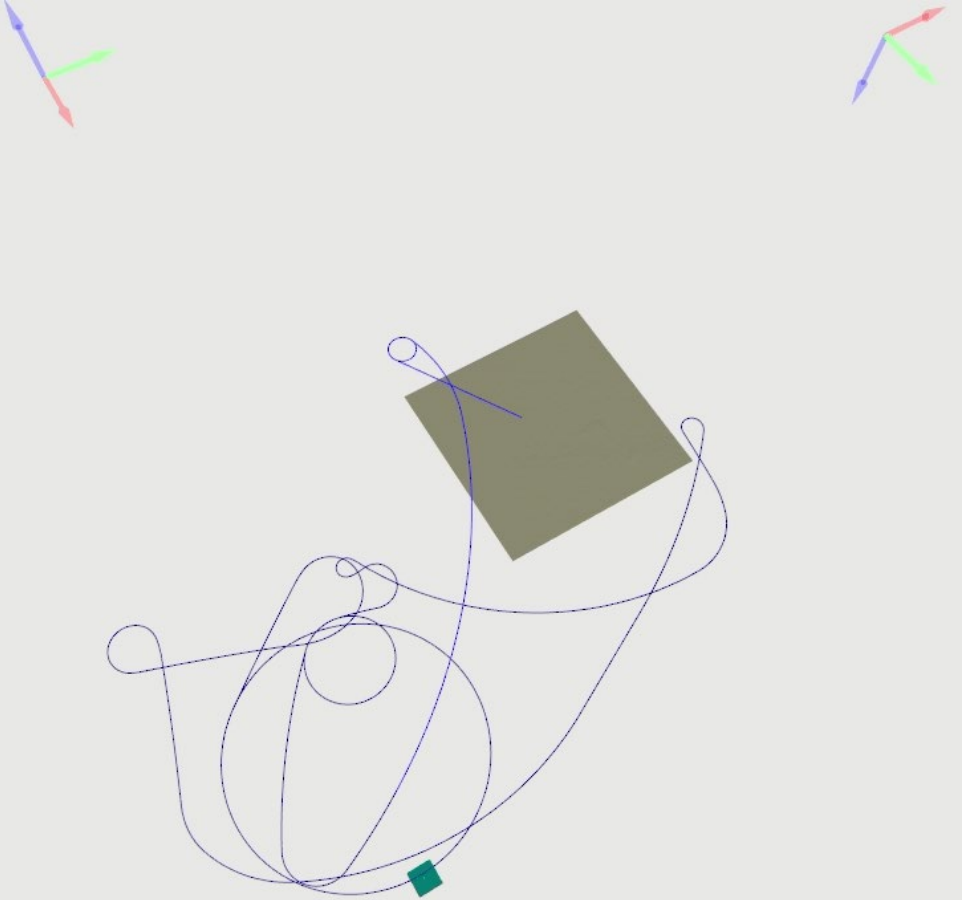
github.com/osrf/lrauv

$ project(‘LRAUV’, 2022)

github.com/osrf/lrauv

$ project(‘LRAUV’, 2022)

- EmPy
- Build step (CMake)
- spherical coordinates



github.com/osrf/lrauv

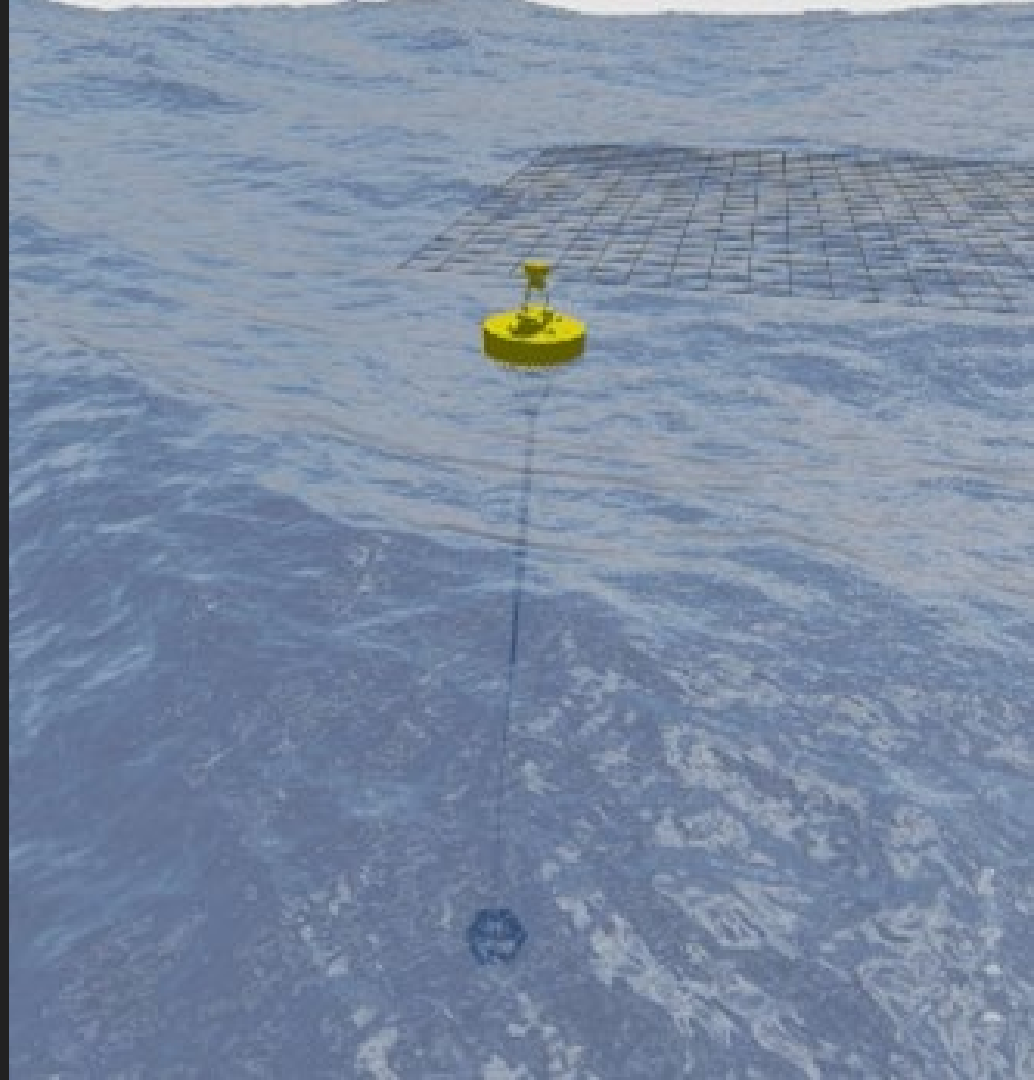
$ project(‘LRAUV’, 2022)

```
from gz.math7 import SphericalCoordinates, Vector3d, Angle
  ...
for tile in tiles:
    vec = Vector3d(math.radians(tile.lat_deg), math.radians(tile.lon_deg), 0)
    pos_enu = sc.position_transform(vec,
        SphericalCoordinates.SPHERICAL,
        SphericalCoordinates.LOCAL2)
    tile.pos_enu = pos_enu
```

github.com/osrf/lrauv

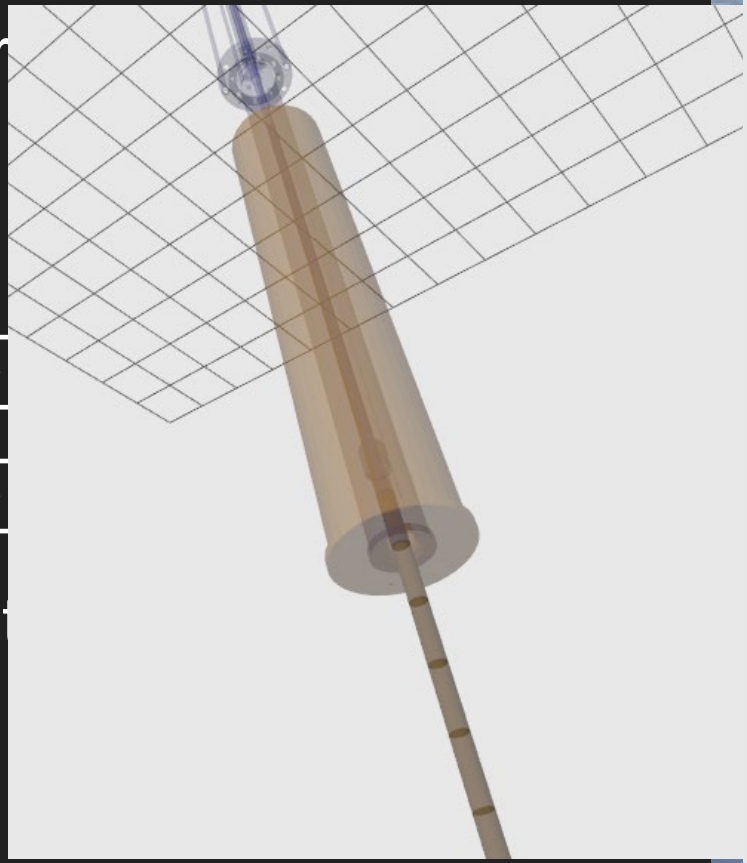
$ project(‘BuoySim’, 2022)

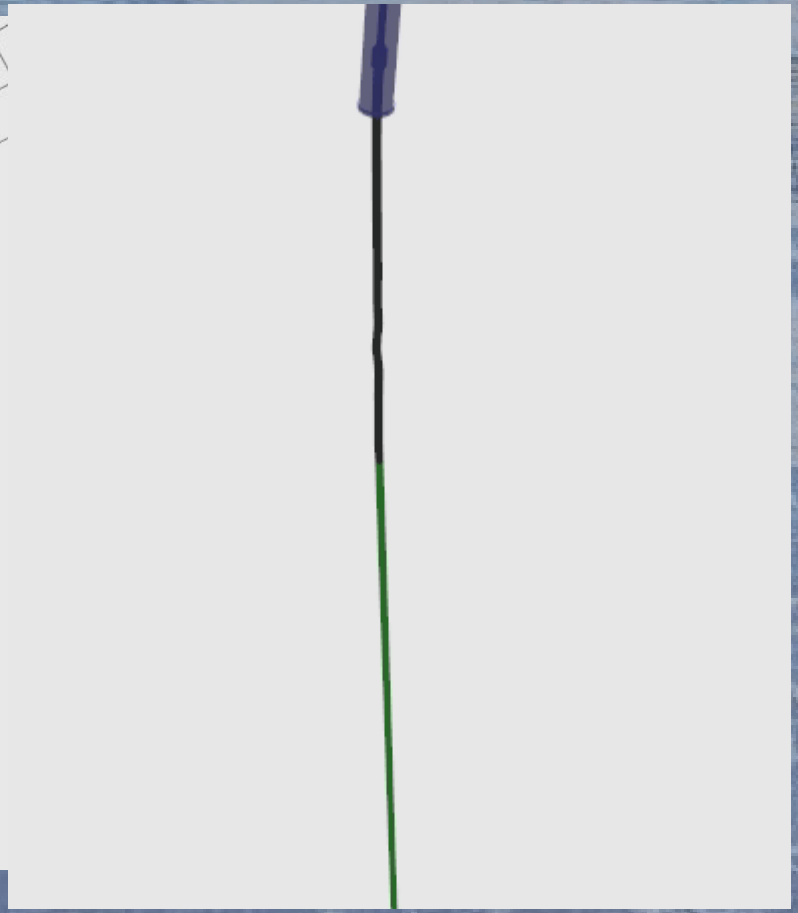
- EmPy
- Build step (CMake)
- tether



github.com/osrf/buoy_sim

$ pr





github.com/osrf/buoy_sim

$ comparison

github.com/chapulina/desplate

chapulina / desplate Public

Pin | Unwatch 1

Code | Issues | Pull requests | Actions | Projects | Wiki | Security

main | 1 branch | 0 tags | Go to file | Add file | Code

chapulina indentation, links on README 🧹 `ce07a9e` 2 hours ago **14** commits

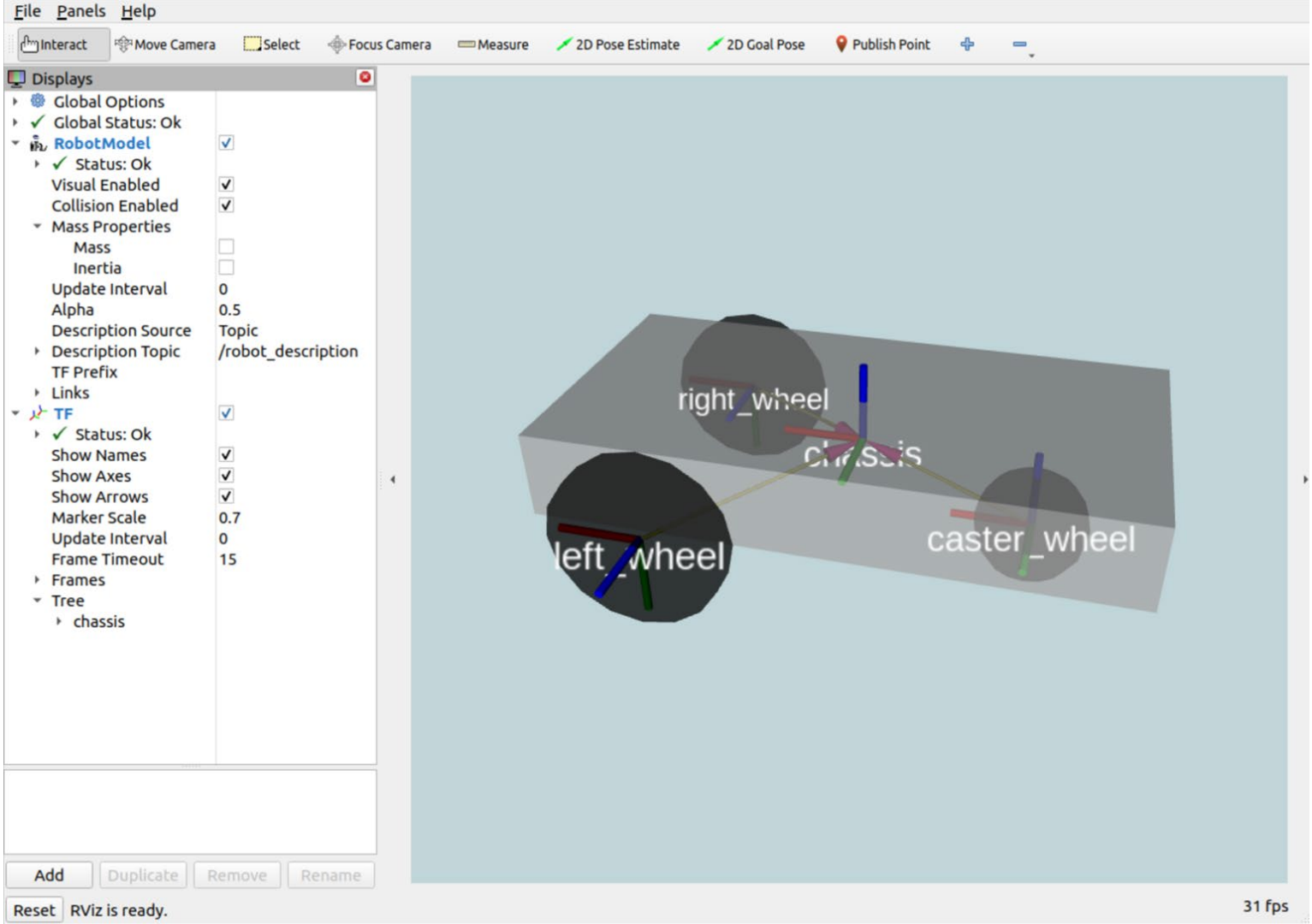
desplate_common	compile time, EmPy 🐍	2 days ago
desplate_empy	indentation, links on README 🧹	2 hours ago
desplate_erb	indentation, links on README 🧹	2 hours ago
desplate_xacro	indentation, links on README 🧹	2 hours ago
images	indentation, links on README 🧹	2 hours ago
LICENSE	Initial commit	4 days ago
README.md	indentation, links on README 🧹	2 hours ago

README.md

DEScription temPLATEs

This repository contains various examples for how to template description files in

デスペレート


File Panels Help
Interact
Move Camera
Select
Focus Camera
Measure
2D Pose Estimate
2D Goal Pose
Publish Point
Displays
Global Options
Global Status: Ok
RobotModel
Status: Ok
Visual Enabled
Collision Enabled
Mass Properties
Mass
Inertia
Update Interval 0
Alpha 0.5
Description Source Topic
Description Topic /robot_description
TF Prefix
Links
TF
Status: Ok
Show Names
Show Axes
Show Arrows
Marker Scale 0.7
Update Interval 0
Frame Timeout 15
Frames
Tree
chassis
right_wheel
chassis
left_wheel
caster_wheel
Add
Duplicate
Remove
Rename
Reset
RViz is ready.
31 fps

Try out all examples

Templating engine	Format	Stage	Command
EmPy	SDF	launch	`ros2 launch desplate_empy vehicle_sdf_generate.launch.py`
EmPy	URDF	launch	`ros2 launch desplate_empy vehicle_urdf_generate.launch.py`
EmPy	SDF	compilation	`ros2 launch desplate_empy vehicle_sdf_installed.launch.py`
EmPy	URDF	compilation	`ros2 launch desplate_empy vehicle_urdf_installed.launch.py`
ERB	SDF	launch	`ros2 launch desplate_erb vehicle_sdf_generate.launch.py`
ERB	URDF	launch	`ros2 launch desplate_erb vehicle_urdf_generate.launch.py`
ERB	SDF	compilation	`ros2 launch desplate_erb vehicle_sdf_installed.launch.py`
ERB	URDF	compilation	`ros2 launch desplate_erb vehicle_urdf_installed.launch.py`
Xacro	SDF	launch	`ros2 launch desplate_xacro vehicle_sdf_generate.launch.py`
Xacro	URDF	launch	`ros2 launch desplate_xacro vehicle_urdf_generate.launch.py`
Xacro	SDF	compilation	`ros2 launch desplate_xacro vehicle_sdf_installed.launch.py`
Xacro	URDF	compilation	`ros2 launch desplate_xacro vehicle_urdf_installed.launch.py`

Try out all examples

Templating engine	Format	Stage	Command
EmPy	SDF	launch	`ros2 launch desplate_empy vehicle_sdf_generate.launch.py`
EmPy	URDF	launch	`ros2 launch desplate_empy vehicle_urdf_generate.launch.py`
EmPy	SDF	compilation	`ros2 launch desplate_empy vehicle_sdf_installed.launch.py`
EmPy	URDF	compilation	`ros2 launch desplate_empy vehicle_u d.launch.py`
ERB	SDF	launch	`ros2 launch desplate_ nch.py`
ERB	URDF	launch	`ros2 lau generate.launch.py`
ERB	SDF	compilation	`ros2 laun vehicle_sdf_installed.launch.py`
ERB	URDF	compilation	`ros2 launch desplate_erb vehicle_urdf_installed.launch.py`
Xacro	SDF	launch	`ros2 launch desplate_xacro vehicle_sdf_generate.launch.py`
Xacro	URDF	launch	`ros2 launch desplate_xacro vehicle_urdf_generate.launch.py`
Xacro	SDF	compilation	`ros2 launch desplate_xacro vehicle_sdf_installed.launch.py`
Xacro	URDF	compilation	`ros2 launch desplate_xacro vehicle_urdf_installed.launch.py`

sdformat_urdf

Try out all examples

Templating engine	Format	Stage	Command
EmPy	SDF	launch	`ros2 launch desplate_empy vehicle_sdf_generate.launch.py`
EmPy	URDF	launch	`ros2 launch desplate_empy vehicle_urdf_generate.launch.py`
EmPy	SDF	compilation	`ros2 launch desplate_empy vehicle_sdf_installed.launch.py`
EmPy	URDF	compilation	`ros2 launch desplate_empy vehicle_urdf_installed.launch.py`
ERB	SDF	launch	`ros2 launch desplate_erb vehicle_sdf_generate.launch.py`
ERB	URDF	launch	`ros2 launch desplate_erb vehicle_urdf_generate.launch.py`
ERB	SDF	compilation	`ros2 launch desplate_erb vehicle_sdf_installed.launch.py`
ERB	URDF	compilation	`ros2 launch desplate_erb vehicle_urdf_installed.launch.py`
Xacro	SDF	launch	`ros2 launch desplate_xacro vehicle_sdf_generate.launch.py`
Xacro	URDF	launch	`ros2 launch desplate_xacro vehicle_urdf_generate.launch.py`
Xacro	SDF	compilation	`ros2 launch desplate_xacro vehicle_sdf_installed.launch.py`
Xacro	URDF	compilation	`ros2 launch desplate_xacro vehicle_urdf_installed.launch.py`

EmPy

Add these lines to a launch file to generate a description from an EmPy template:

desplate/desplate_empy/launch/vehicle_sdf_generate.launch.py
Lines 33 to 36 in 86b7960

```
33    # These 3 lines generate a description file from an EmPy template
34    with open(template_path) as template_file:
35        template = template_file.read()
36    description_str = em.expand(template, {})
```

ERB

Add this line to a launch file to generate a description from an ERB template:

desplate/desplate_erb/launch/vehicle_sdf_generate.launch.py
Lines 32 to 33 in 86b7960

```
32    # This line generates a description file from an ERB template
33    description_str = subprocess.run(['erb', template_path], capture_output=True).stdout.decode()
```

Xacro

Add this line to a launch file to generate a description from a Xacro template:

desplate/desplate_xacro/launch/vehicle_sdf_generate.launch.py
Lines 33 to 34 in 86b7960

```
33    # This line generates a description file from a Xacro template
34    description_str = xacro.process_file(template_path).toxml()
```

$ compare(‘Xacro’, ‘EmPy’)

```
<?xml version="1.0"?>
<sdf xmlns:xacro="http://www.ros.org/wiki/xacro">
  <xacro:macro name="inertial_cylinder" params="mass radius length">
    <xacro:property
      name="xy"
      value="${mass * (3 * radius * radius + length * length) / 12.0}"/>
    <xacro:property
      name="z"
      value="${mass * radius * radius * 0.5}"/>
    <inertial>
      <mass>${mass}</mass>
      <inertia>
        <ixx>${xy}</ixx>
        <ixy>0.0</ixy>
        <ixz>0.0</ixz>
        <iyy>${xy}</iyy>
        <iyz>0.0</iyz>
        <izz>${z}</izz>
      </inertia>
    </inertial>
  </xacro:macro>
</sdf>
```

```
@{
xy = mass * (3 * radius * radius + length * length) / 12.0
z = mass * radius * radius * 0.5
}@
<inertial>
  <mass>@(mass)</mass>
  <inertia>
    <ixx>@(xy)</ixx>
    <ixy>0.0</ixy>
    <ixz>0.0</ixz>
    <iyy>@(xy)</iyy>
    <iyz>0.0</iyz>
    <izz>@(z)</izz>
  </inertia>
</inertial>
```

$ compare(‘EmPy’, ‘ERB’)

```
@{
xy = mass * (3 * radius * radius + length * length) / 12.0
z = mass * radius * radius * 0.5
}@
<inertial>
  <mass>@(mass)</mass>
  <inertia>
    <ixx>@(xy)</ixx>
    <ixy>0.0</ixy>
    <ixz>0.0</ixz>
    <iyy>@(xy)</iyy>
    <iyz>0.0</iyz>
    <izz>@(z)</izz>
  </inertia>
</inertial>
```

```
<%
xy = $mass * (3 * $radius * $radius + $length * $length) / 12.0
z = $mass * $radius * $radius * 0.5
%>
<inertial>
  <mass><%= $mass %></mass>
  <inertia>
    <ixx><%= xy %></ixx>
    <ixy>0.0</ixy>
    <ixz>0.0</ixz>
    <iyy><%= xy %></iyy>
    <iyz>0.0</iyz>
    <izz><%= z %></izz>
  </inertia>
</inertial>
```

$ compare(‘URDF’, ‘SDF’)

```
<%
inertia = (2 * $mass * $radius * $radius) / 5.0
%>
<inertial>
  <mass value="<%= $mass %>" />
  <inertia
    ixx="<%= inertia %>"
    ixy="0.0"
    ixz="0.0"
    iyy="<%= inertia %>"
    iyz="0.0"
    izz="<%= inertia %>" />
</inertial>
```

```
<%
inertia = (2 * $mass * $radius * $radius) / 5.0
%>
<inertial>
  <mass><%= $mass %></mass>
  <inertia>
    <ixx><%= inertia %></ixx>
    <ixy>0.0</ixy>
    <ixz>0.0</ixz>
    <iyy><%= inertia %></iyy>
    <iyz>0.0</iyz>
    <izz><%= inertia %></izz>
  </inertia>
</inertial>
```

$ questions