ROS2 + WASM

Bringing the Robot Operating System to the Web

Tobias Fischer[1], Isabel Paredes[2,3], Michael Batchelor[1], Thorsten Beier[3], Jesse Haviland[1], Silvio Traversaro[4], Wolf Vollprecht[5], Markus Schmitz[2] & Michael Milford[1]

[1] QUT Centre for Robotics
[2] RWTH Aachen University
[3] QuantStack Scientific Computing
[4] iit Istituto Italiano di Tecnologia
[5] Prefix·dev

Using ROS 2

docs.ros.org

robostack.github.io

conda install ros-jazzy-desktop

ros2wasm.dev

Motivation

- ROS is the de facto standard middleware in robotics, but platform support is very limited ☹ ➔ accessibility and reproducibility challenges in industry, research & teaching!
- **RoboStack** alleviated some of these challenges by facilitating installation across Linux, MacOS & Windows via the Conda package manager:

conda install ros-humble-desktop

- What if we could design a solution that requires ***no installation whatsoever*** and provides a standardised execution platform?

Background + Glossary

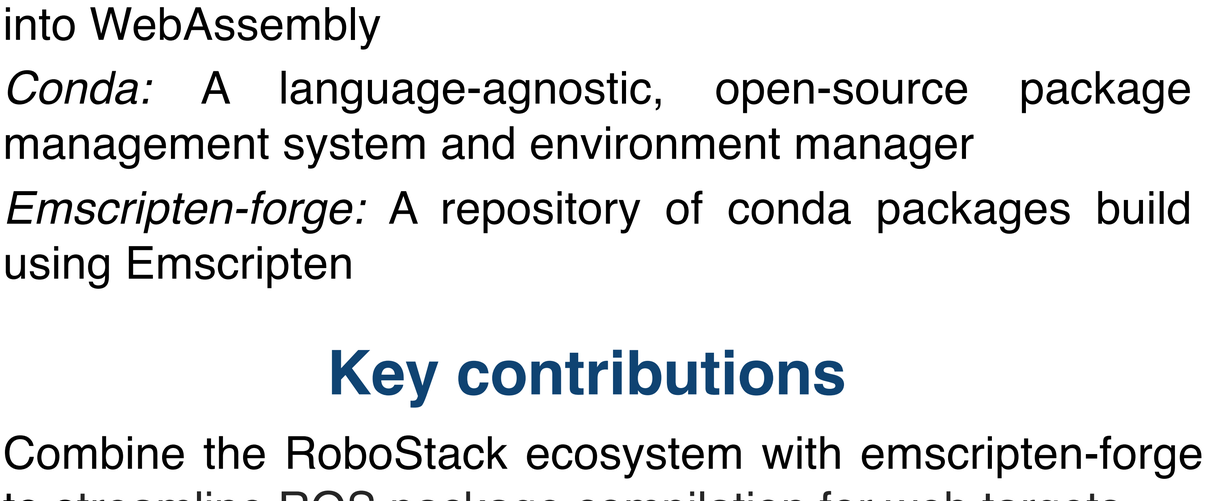
- *WebAssembly (WASM):* Binary executable format designed for execution on modern web browsers
- *Emscripten:* Compiler toolchain to convert C/C++ code into WebAssembly
- *Conda:* A language-agnostic, open-source package management system and environment manager
- *Emscripten-forge:* A repository of conda packages build using Emscripten

Key contributions

- Combine the RoboStack ecosystem with emscripten-forge to streamline ROS package compilation for web targets

RoboStack + Emscripten Forge

- Cross-compiled ROS 2 to WebAssembly, ***enabling execution directly in web browsers***
- Developed a custom ROS 2 middleware for browser communications (***rmw-wasm***)



- Created **www.ros2wasm.dev** as a demonstration platform
- Extended support to Peter Corke's Robotics Toolbox for Python with Swift simulator adaptation

Applications: What does it do for you?

- Execute ROS 2 nodes in browser environments
- Same code across all devices: Mobile & Desktop
- ***Share complete ROS environments via web links!***
- Communicate between ROS nodes within browsers

- ***Control of physical robots through web interfaces:*** Demonstrations with LEGO BOOST Vernie

- ***Reproducible environments*** for education, research papers, and collaboration

Future work

- Extension to ROS2 Actions, QoS, and Parameters
- Include graphical interfaces (RViz, Gazebo, MoveIt)

Acknowledgements: This research was partially supported by funding from ARC DECRA Fellowship DE240100149 to TF, ARC Laureate Fellowship FL210100156 to MM, and the QUT Centre for Robotics.