An Introduction to the Space Robot Operating System. S. Will Chambers, Blue Origin Advanced Development Programs, 20819 72nd Ave S.Kent, WA 98032, (Contact: schambers@blueorigin.com)

Introduction: A sustainable lunar presence requires robotic and autonomous space systems to perform tasks ranging from in-situ resource utilization, excavation and construction, inspection and repair, and power generation and distribution. Common to these systems is the need for a robust flight-quality software that is certifiable to mission and safety assurance standards. Although such software frameworks like Core Flight System [1] exist for spacecraft, there currently does not exist a comparable space-quality software framework designed for robotic and autonomous space system.

Space Robot Operating System. Blue Origin, in collaboration with NASA, is leading the technical maturation of Space Robot Operating System (Space ROS), a space-quality software framework designed specifically for autonomous and robotic space systems and missions. Space ROS software includes communication middleware, core software packages and application software packages that enable autonomous and robotic lunar systems to reliably execute their tasks. Application packages provide the operational functionality critical to lunar operations, and include robotic manipulation, mobility, autonomy and collaboration software kits. Space ROS will also define an agent communication language and space ontology that allows robotic space systems share information, allocate task, and form coalitions.

Space ROS is predicated on open-source ROS [2] and therefore inherits many of its useful qualities. Users of ROS, which comprises the bulk of roboticists, will find Space ROS readily adoptable. Its modularity and reusable packages will enable rapid software development, and an estimated 40% reduction in develop costs [3]. When released, the Space ROS open-source repositories will be hosted in GitHub, and therefore accessible to lunar roboticists and the space community worldwide.

There is precedent using ROS in space robotic experiments and missions, including Astrobee, Robotnaut and Valkyrie. The upcoming VIPER) Volatiles Investigating Polar Exploration Rover) mission is employing ROS on its ground node [4]. ROS, however, is not flight-quality software and is limited in its use on the lunar surface and other space missions. We are therefore taking special care to ensure the Space ROS software that lunar robotic and autonomous systems employ is flight-qualifiable. For instance, we are designing Space ROS to subscribe to strict memory management criteria, to be real-time and deterministic, and to be compatible with the processing platforms expected in robotic space systems.

Furthermore, the Space ROS team is defining a quality policy and code compliance rules to which Space ROS software will subscribe. The policy aligns to existing flight software standards including NPR 7150.2 [5]. And we are developing a continuous integration infrastructure and a suite of automated tools that enforces code quality by checking Space ROS software for compliance. The infrastructure detects and tracks non-compliance in an accessible database to allow an open community of software developers to contribute to the technical maturation of Space ROS. Moreover, the Space ROS continuous integration infrastructure catalogues software pedigree, artifacts and documentation to support the upstream qualification of a robotic mission's software.

Space ROS is a paradigm shift in how robotic flight software is maturated and maintained. Rather than customizing software per mission, Space ROS provides accessibility and reuse of quality software, continuous integration for persistent software maturation and maintenance, and an open-community of contributors and users. Our intent is that Space ROS will become the de facto software standard for robotic and autonomous systems on the moon, and beyond.

References:

[1] https://cfs.gsfc.nasa.gov [2] https://design.ros2.org [3] JPL D-26303, Handbook for Software Cost Estimation (2003) p 19. [4] N. V Patel, NASA's next lunar rover will run open-source software, April 12, 2021, MIT Tech Review [5] NASA Procedural Document NPR 7150.2C