

CONCEPT Open-source coding involves developing software for robots using open-source tools, platforms, and methodologies. This approach promotes collaboration, transparency, and accessibility by allowing anyone to view, modify, and redistribute the software's source code freely. It enables the creation of more capable and affordable robots by leveraging the collective intelligence and expertise of the community and fostering innovation.



EXAMPLES

OPERATING SYSTEM: Software infrastructure and services that manage the hardware resources, schedule tasks, and enable communication and coordination among different software components.

PROGRAMMING LANGUAGES: High-level languages such as Python, C++, and Java, that are used to write the software code that controls the behavior and functionality of the robot.

LIBRARIES AND FRAMEWORKS: Provides pre-built code and tools that enable developers to create software components quickly and efficiently, such as ROS (Robot Operating System), OpenCV (Open Source Computer Vision Library), and TensorFlow (Machine Learning Framework).

BACKGROUND

Open-source code has its roots in the 1980s and 1990s. The emergence of personal computers and the Internet enabled the creation of open source software communities, such as the Free Software Foundation and the Linux operating system. These communities promoted the principles of collaboration, transparency, and accessibility in software development, which inspired the creation of open-source robotics projects, such as Player/Stage and ROS (Robot Operating System) in the 2000s.

ROS, which was launched in 2007 by Willow Garage, quickly became the de facto standard for open-source robotics software, due to its modular architecture, robustness, and community support. ROS enabled the development of a wide range of applications, from small hobbyist robots to large industrial robots, and facilitated the integration of different hardware and software components through standardized interfaces and protocols.