

JAWABAN NOMOR 1 URO TEST PROGRAMMING

Nama : Aryo Wisanggeni

NIM : 13523100

Jurusan : Teknik Informatika

Jawaban 1.a.

Robot Operating System (ROS) adalah sekumpulan library perangkat lunak dan alat yang membantu developer membangun aplikasi robot. ROS berperan untuk memberi fasilitas untuk mengintegrasikan berbagai sensor robot sehingga menyederhanakan pengembangan algoritma robot dalam melakukan berbagai tugasnya.

ROS penting dapat diimplementasi untuk integrasi berbagai komponen karena memberi keunikan interaksi dengan environment melalui berbagai bahasa pemrograman. Ini memungkinkan node yang tertulis dengan bahasa pemrograman 1 (katakanlah Python) untuk berkomunikasi dengan node lain yang menggunakan bahasa pemrograman 2 (katakanlah C++) tanpa hambatan. Inti dari ROS adalah arsitektur modular yang membagi fungsionalitas robot menjadi komponen yang dapat digunakan lagi dan memudahkan proses pengembangan software robot, walaupun menggunakan komponen yang tidak seragam dalam satu robot itu sendiri.

Jawaban 1.b.

Terdapat tiga perbedaan utama antar ROS dan ROS2:

1. Architecture

ROS menggunakan Master-Slave Architecture dan XML-RPC middleware, sedangkan ROS2 menggunakan Data Distribution Service (DDS). Architecture ROS2 didesain untuk lebih efisien, latency rendah, scalability baik, dan Quality of Service (QoS) yang dapat dikonfigurasi dibanding architecture ROS.

Selain itu, dari sisi API, library ROS menggunakan dua library: roscpp untuk C++ dan rospy untuk Python. Sedangkan itu, ROS2 menggunakan library dasar berbahasa C atau RCL (ROS Client Library) dengan berbagai library dikembangkan dari library dasar tersebut. Ini menjamin fungsi dasar dapat digunakan oleh berbagai API berbeda dan memungkinkan ROS2 untuk mempunyai support lebih banyak bahasa pemrograman dari ROS biasa

2. Fitur

ROS2 dapat menjalankan beberapa node secara paralel sehingga dapat memanfaatkan prosesor multi-core dengan lebih baik dibanding ROS

ROS2 juga lebih unggul melakukan real-time processing karena komunikasi low-latency dibutuhkan

3. Ecosystem

ROS menggunakan Ubuntu, sedangkan ROS2 dapat di-run di MacOS, Windows, Ubuntu, dan OS lainnya.

Kesimpulan: Secara performa, ROS2 jelas lebih unggul dari ROS karena alasan yang sudah disebut di atas. Dari sisi keamanan, DDS ROS2 memberi keamanan lebih seperti enkripsi sehingga, walaupun robot melalui tempat dengan network yang tidak aman, informasi yang ditransfer dari robot ke robot tidak dapat diambil oleh pihak lain. Fitur keamanan tersebut termasuk kelemahan ROS yang keamanannya memang kurang. Dari sisi pemeliharaan jangka panjang, support untuk ROS akan berakhir tahun 2025, sedangkan ROS2 adalah sistem yang relatif lebih baru dan supportnya masih lama akan ada. Tidak hanya itu, setiap package ROS2 juga memiliki beberapa maintainer yang bertugas untuk memelihara sistem.

Jawaban 1.c.

Simulasi robotik dalam software sangat penting sebelum mengembangkan robot secara nyata. Pertama adalah dari sisi biaya, dengan adanya simulasi, dapat dilihat apakah setiap komponen bekerja satu sama lain dengan ideal sebelum membeli komponen-komponen tersebut. Ini menghindari salah beli ketika membeli komponen robot.

Dari development pun juga penting adanya simulasi, semisal stress-test suatu body robot terhadap berbagai kondisi yang dihadapi dan membantu developer untuk memilih material yang lebih baik untuk robot sehingga menghemat waktu: membangun robot yang tidak sesuai dan akhirnya perlu dibuat ulang karena masalahnya ada di “pondasi” robot tersebut.

Jadi, dengan adanya simulasi, memungkinkan developer untuk melakukan *try and error* secara virtual sehingga tidak buang-buang waktu dan biaya untuk membangun robot secara fisik dan gagal berkali-kali karena coba-cobanya secara virtual dulu.

Jawaban 1.d.

Gazebo adalah simulator robotika 2D dan 3D yang open-source. Alat ini dapat menyimulasi robot dalam berbagai kondisi seperti lingkungan indoor dan outdoor secara akurat dan efektif.

Dengan asumsi ROS sudah di install, Gazebo dapat di install dengan command:

```
sudo apt install ros-foxy-gazebo-ros-pkgs
```

Seterusnya dengan menggunakan command **gazebo** dengan **ROS launch file**, gazebo dan ROS dapat di-run secara bersamaan seperti berikut untuk loading .world tertentu:

```
ros2 launch gazebo_ros gazebo.launch.py world:=src/robot_testing/worlds/world6.world
```

Selanjutnya, setelah konfigurasi file URDF, `robot_state_publisher` dapat di publish ke `robot_description` dan ini akan spawn robot yang sesuai dengan URDF di `robot_description` ke world.

Dengan demikian, gazebo dan ROS terintegrasi sehingga robot dapat disimulasikan di gazebo.

Proses ini dapat dipersingkat lagi dengan membuat suatu launcher.

Jawaban 1.e.

Untuk robot dapat merencanakan rute dari satu poin ke satu poin lain, diperlukan peta dari lingkungannya. Simultaneous Localization and Mapping (SLAM) memungkinkan robot untuk membuat peta secara online dan sekaligus melakukan localization dirinya pada peta tersebut.

Dari itu, mapping adalah kemampuan robot untuk membuat peta lingkungannya untuk mempermudah navigasi dalam lingkungan tersebut. Sedangkan, localization adalah kemampuannya untuk memproyeksi posisi/menentukan posisi robotnya dalam peta lingkungan tersebut. Dalam SLAM, kedua hal ini dilakukan secara bersamaan dan cepat sehingga navigasi robot dapat berjalan dengan lancar.

Jawaban 1.f

Dalam robotika, frame merujuk pada system koordinat yang menggambarkan posisis dan orientasi sutatu objek, biasanya sepanjang sumbu x, y, dan z. Dalam satu scene, ada satu frame dasar yang biasanya dinamakan world atau map. Frame lain dalam satu scene, termasuk frame robotnya, disebut base_llink dan termasuk children dari frame dasar.

Transform mendefinisikan translasi dan rotasi yang perlu dilakukan untuk bergerak dari frame source hingga ke frame target. Satu set dari transform scene disebut sebagai transform tree, ini memungkinkan robot untuk menentukan posisi dan orientasi dari frame apa pun. Transform pun dibedakan menjadi dua: transform static dan transfor mobile. Transform static akan selalu konstan, seperti di antara frame base_link dan sensor_link. Transform mobile akan berubah tergantung pada lingkungannya, seperti di antara frame base_link dan tangan robot (yang bisa bergerak) atau objek yang terpisah dari robotnya.