

Avoimen lähdekoodin ROS2 nopeuttaa autonomisten koneiden kehittämistä

16.5.2024

Töysän Säästöpankkisäätiön rahoittaman Autonomiset ajoneuvot -selvityshankkeen toimenpiteenä hankittu TurtleBot 4 mobiilirobotti (kuva 1) on ollut alkuvuoden 2024 ajan SeAMKin Robolabran käytössä. TurtleBot on alkujaan vuonna 2010 Willow Carage (Ackerman, 2013) laboratorion kehittämä ja sittemmin jo neljänteen sukupolveensa ehtinyt tuotesarja, joka on tarkoitettu mobiilirobotiikan ”sisäänheittotuotteeksi” opetus- ja tuotekehityskäyttöön. Viimeisimmän version TurtleBotista on kehittänyt kanadalainen Clearpath Robotics, ja se tuli myyntiin vuonna 2022 (Clearpath Robotics, i.a.).

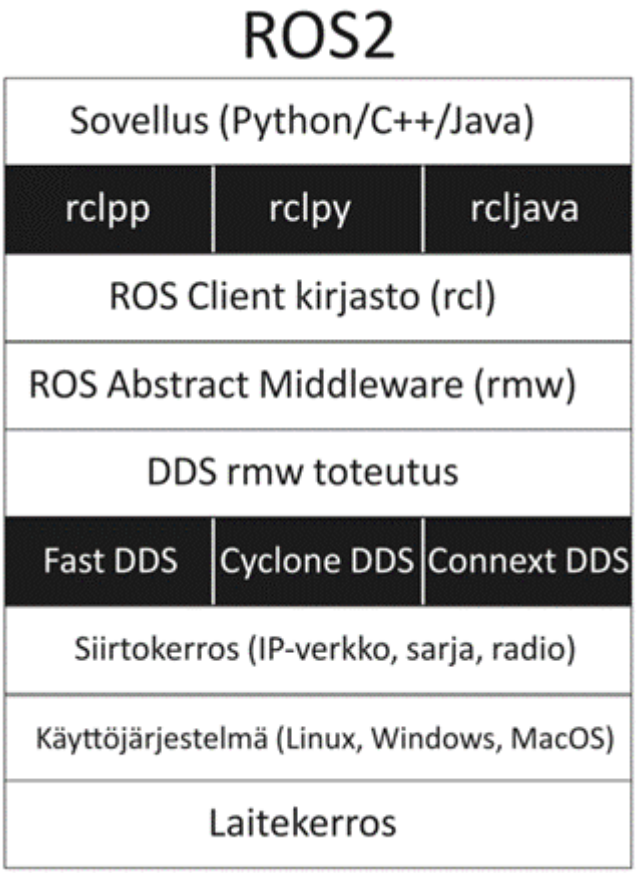


Kuva 1. TKI-asiantuntija Hannu Hakalahti käskyttää TurtleBot 4 mobiilirobottia ROS2-verkon viesteillä (Kuva: Tommi Ylimäki).

Autonomiset ajoneuvot -hankkeen kirjallisuus- ja markkinakatsausosuudessa kävi varsin nopeasti selväksi, että avoimen lähdekoodin Robot Operating System 2 (ROS2) väliohjelmisto on kokonaisuus, johon SeAMKin on syytä perehtyä niin TKI-työn kuin erityisesti opetuksenkin kehittämisen näkökulmista: automaattinen liikkuminen ja työsuoritteiden tekeminen vaativat laitteen kokoluokasta riippumatta samantapaisia teknologiaratkaisuja ympäristön havainnoimisesta koneen paikoitukseen ja reitinsuunnitteluun. Näin ollen tuotteiden tekninen kehitystyö sujuvoituu hyödyntämällä toimivaksi havaittuja arkkitehtuurillisia rakenteita (Macenski ym., 2022). Vuonna 2007 käynnistetyn, avoimen lähdekoodiin perustuvan Robot Operating System -projektin tavoitteena oli välttää jatkuvaa samantapaisten ohjausohjelmistojen uudestaan kirjoittamista mobiilirobotteja kehitettäessä (Machado ym., 2021). ROS2 on ROSin käytön myötä kertyneiden kokemusten pohjalta uudestaan kehitetty kokonaisuus, jossa on huomioitu monia tuotannollisen käytön kannalta tähdellisiä seikkoja, muun muassa reaaliaikavaatimusten, tietoturvan, luotettavuuden ja verkon skaalautuvuuden suhteen. ROS2:ta kehitettäessä näkökulma on ollut siinä, että kehikko tarjoaisi kehitettäville roboteille mahdollisimman hyvän valmiuden markkinoille tuomiseen. ROS2 projektin ympärille on kehittynyt aktiivinen kansainvälinen yhteisö (ROS, i.a.), joka on viime vuosina saanut tuekseen myös globaaleja teollisuustoimijoita.

Itsessään ROS2 on robottisovellusten kehittämiseen soveltuva kokoelma C++- ja Python-kielisiä avoimen lähdekoodin ohjelmakirjastoja ja -työkaluja, jotka hyödyntävät C-kielellä kirjoitettua rcl-nimistä rajapintaa (Elmofty, 2022) (kuva 2). ROS2:n varaan rakennetussa järjestelmässäeri tehtäviä –vaikkapa ympäristön

skannaamista laseretäisyysmittarilla, erilaisten paikoitusmenetelmien tuottaman paikkadatan yhteensovittamista tai lopulta ajokäskyjen syöttäminen robotille – toteuttavat niille omistetut ohjelmistoprosessit (nodet).Nämä keskustelevat keskenään ROS2-verkossa aihealueiden (topic) kautta julkaisija–tilaaja periaatteella.Topicien ohella prosessit voivat kutsua toistensa toimintoja (action) ja palveluita (service).Hajautetun järjestelmästä tekee se, että yksittäisten nodejen ei tarvitse huolehtia siitä, missä tai minkälaisen verkkoyhteyden päässä jokin toinen node toimii: ROS2 on rakennettu niin kutsutun Data Distribution Service (DDS) väliohjelmiston päälle, jonka tehtävänä on pitää huoli siitä, että kukin datapaketti toimitetaan kyseistä aihealuetta kuunteleville prosesseille samassa ROS2 verkossasaman tietokoneen sisällä tai koneiden välillä. DDS:ssä ja näin ollenROS2:ssa ei siis ole mitään keskitettyä palvelinta, jonka kautta kaikki dataliikenne soljuisi. Tämä osaltaan tekee järjestelmästä skaalautuvan ja virhesietoisen(Elmofty, 2022). Kaikkiaan ROS2ohjaa rakentamaan rajapintapohjaista arkkitehtuuria, jolloin järjestelmistä tulee modulaarisia



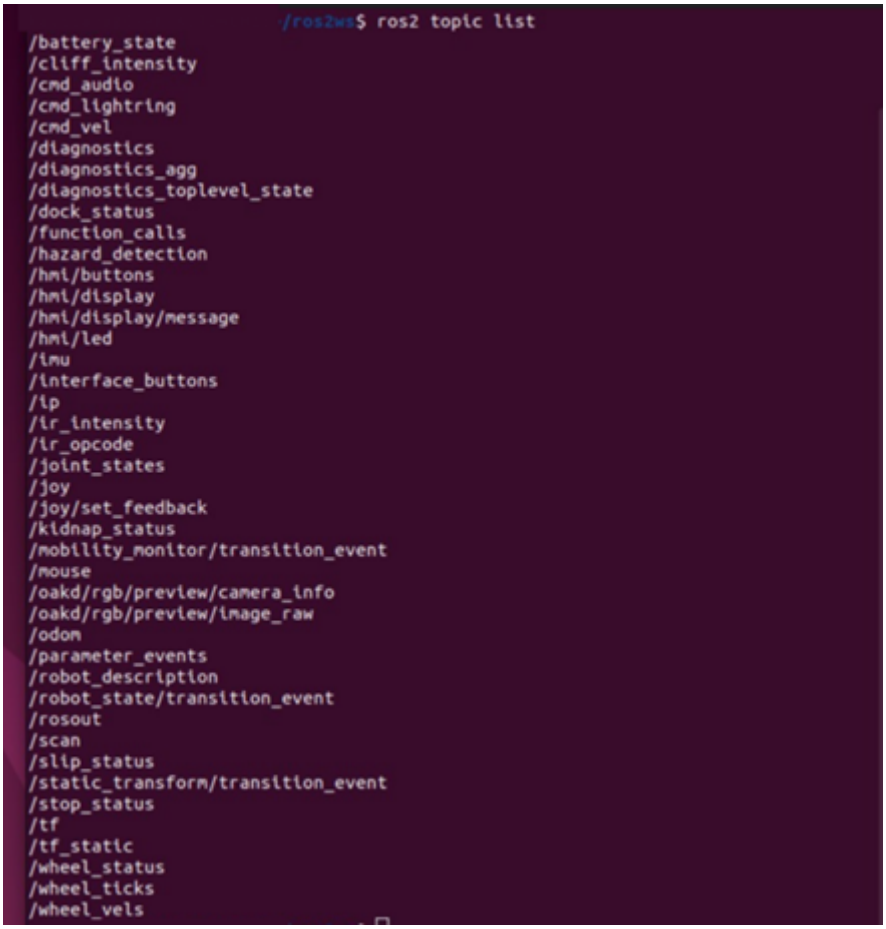
ja uudelleenkäytettäviä.(Shanmugam, 2022b)
Kuva 2. ROS2 ohjelmistokerrokset (soveltaen Shanmugam, 2022a).

ROS2:n yhteydessä hyödynnetään usein Gazebo-nimistä simulaattoria, jolla voidaan mallintaa kehitettävän laitteen fyysistä toimintaa ROS2:n järjestelmän ohjaamana jo ennen kuin laitetta on olemassa. Näin voidaan testata vaikkapa uuden anturin toimivuutta ja kehittää sille tarvittavaa ohjelmistoa huomattavasti suoran fyysisen laitteen kehittämistä resurssitehokkaammin (Schierenebck, 2020).

Alkeelliseenkaan itsenäiseen toimintaan kykenevän mobiilirobotin rakentaminen vaatii teknistä osaamista monella eri abstraktiotasolla fyysisen alustan moottorikäytöistä tehtävän suunnitteluun ja parvitoiminnan ohjaamiseen saakka. ROS2 tarjoaa kehikon, jonka puitteissa tällaista kokonaisuutta on mahdollista kehittää jossain määrin valmiista palasista rakentaen. Tampereen yliopiston projektipäällikkö Jukka Yrjänäisen sanoin ”ROS2 on hyvä kehikko, mutta monimutkaiset asiat ovat silti monimutkaisia” (Ylimäki, 2024), joten myös

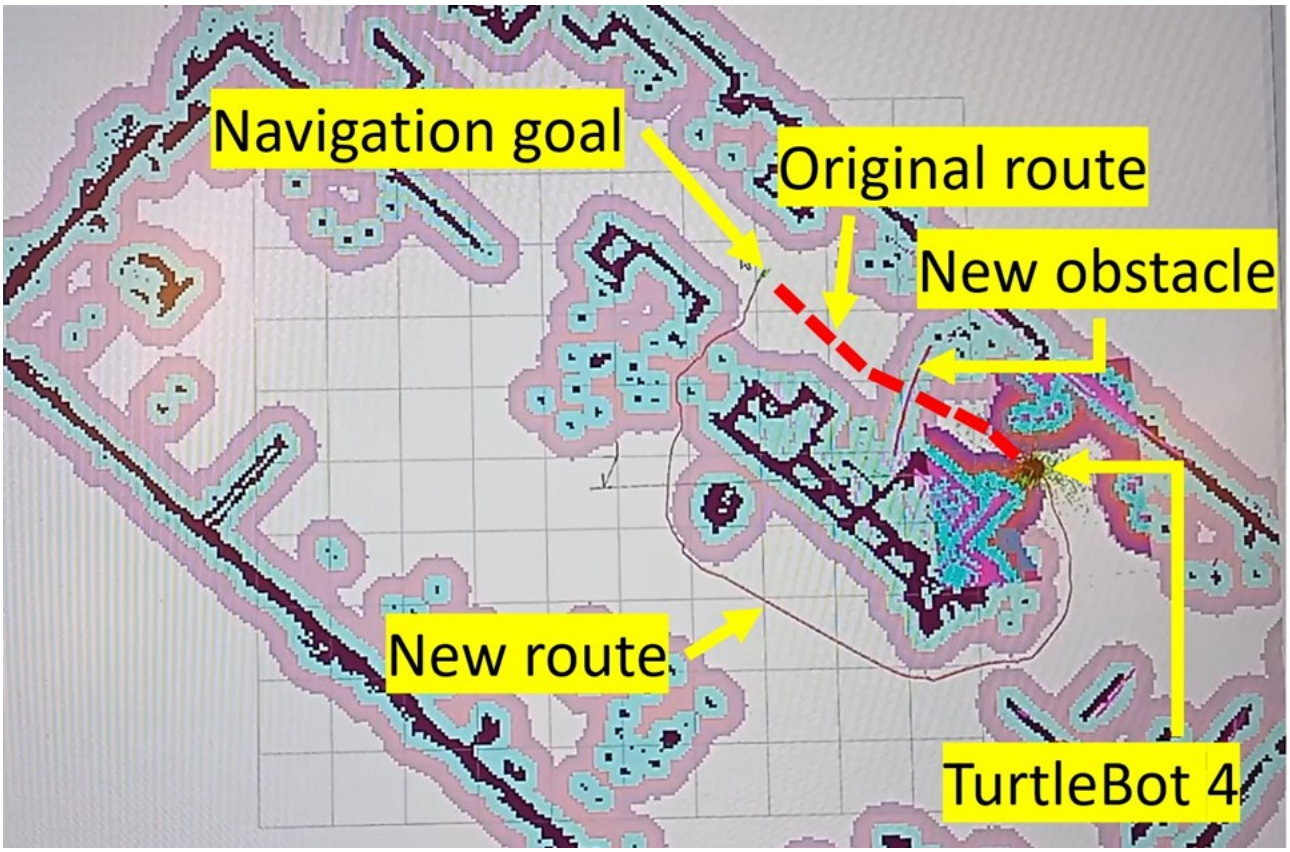
ROS2:n haltuunotto vaatii askellettua etenemistä. Tähän oppimiskäyrän loiventamiseen TurtleBotin kaltainen, laajan käyttäjäkunnan omaksuma ja hyvin dokumentoitu, vaikka ulkonaisesti vaatimatonkin laite,soveltuu mainiosti.

TurtleBot 4:n arkkitehtuuri rakentuu kolmeen fyysiseen kerrokseen: varsinainen ”ajoalusta” on automaattisen Roomba-pölynimurin runko, jota ohjataan opetuskäyttöön kehitetyllä iRobot Create 3 järjestelmällä (iRobot Education, i.a.). Tämän kokonaisuuden päälle Clearpath on lisännyt oman, autonomisen ajamisen vaatimilla antureilla täydennetyn, Raspberry Pi 4 -korttikoneelle rakennetun ROS2-kerroksen (Clearpath Robotics, 2022). TurtleBot pystyy etenemään paitsi tasaisella lattialla, myös matalahkojen kynnysten yli. Robotti liikkuu differentiaaliohjatusti kahden vetävän pyörän voimin eli se pystyy kääntymään umpikujaan päätyessään myös paikallaan. TurtleBotin autonominen liikkumiskyky perustuu robotin paikoitukseen pyöräenkoodereiden, inertiaaliyksikön (IMU) ja pyörivän 2D-LiDARin avulla. Tätä dataa hyödynnetään ROS2-ohjelmistossa siten, että TurtleBot kykenee siirtymään sille syötetyn ohjelman mukaisesti karttapohjalla (kuva 3)itsenäisesti koordinaattipisteestä toiseen. Nämä koordinaattipisteet voidaan määritellä itse kehitetyssä sovelluksessa tai RViz-käyttöliittymässä, jotka ovat samassa ROS2-verkossa.TurtleBotiin on integroitu myös rajakytkinpohjaiset törmäystunnistimet, infrapuna-antureihin perustuva jyrkän reunan tunnistussekä RGB-syvyyskamera. TurtleBot osaa myös hakeutua itsenäisesti lataustelakkaansa.Laitetta on mahdollista ohjata”käsinkin”, joko ulkoisen tietokoneen näppäimistöllä tai erillisellä Bluetooth-ohjaimella. ROS2 kokonaisuuden opiskelemisen ja pienen jatkokehittämisen näkökulmasta TurtleBotin etuna on Clearpathinjatkokehittämät ROS2-yhteisön avoimet lähdekoodit,tutoriaalit ja varsin laaja käyttäjäyhteisö.TurtleBotin tuotteistuksen ideana on tuoda kaikki anturidata ja alustan ohjauskomennot ROS2-rajapinnalle (kuva 4), jolloin mobiilirobotin jatkokehitys on verrattain helppoa ja nopeaa.



Kuva 3. TurtleBot 4:n ROS2 topic -listaus. Anturidata ja ajokomennot on tuotu käyttäjän saataville.

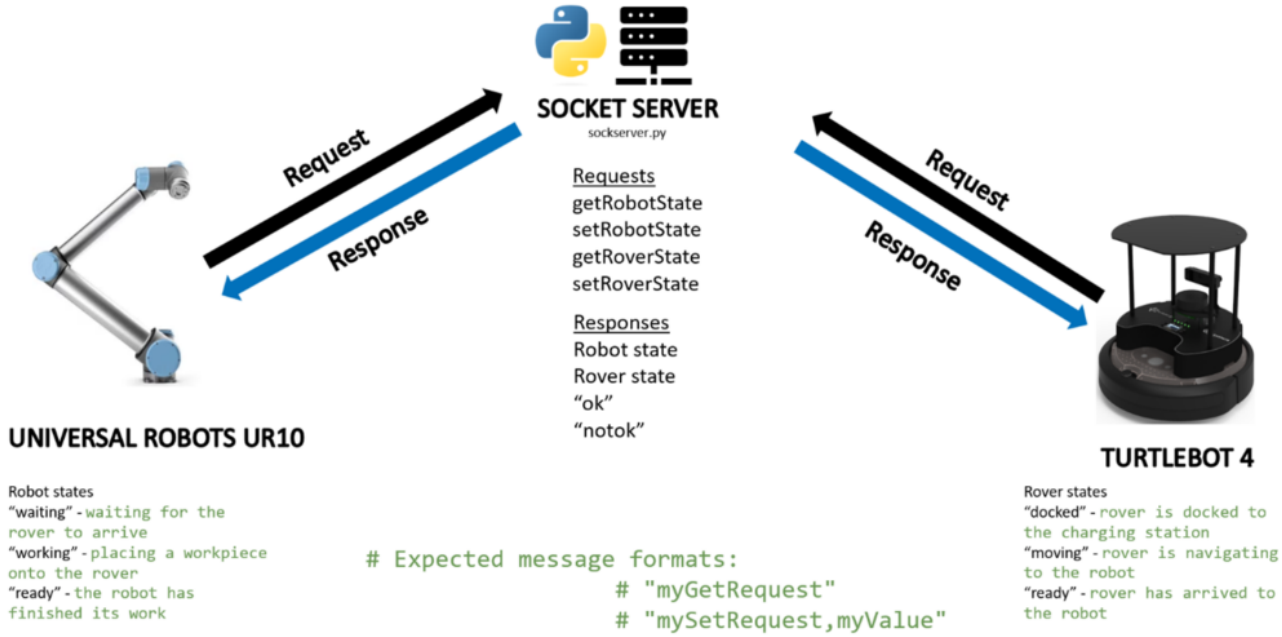
Autonomiset ajoneuvot -hankkeessa käytiin läpi TurtleBotin käyttöönottoprosessi ja todennettiin sen navigaatiokerroksen kyvykkyys toimia sekä staattisessa että dynaamisesti muuttuvassa ympäristössä (kuva 5). Hankkeessa kehitettiin demo-luonteinen, kevyt graafinen käyttöliittymä (kuva 6) sekä työsykli, jossa TurtleBot toimii kahden Universal Robots -yhteistyörobotin kanssa kappaleenkuljetustehtävässä. Jälkimmäinen todentaa ROS2-ohjatun mobiilirobotin helppoa integroitavuutta muihin ohjelmoitaviin laitteisiin: kommunikointi yhteistyörobottien kanssa toteutettiin socket-viesteillä siten, että mobiilirobotti pystyy kertomaan valmiutensa luovuttaa tai ottaa kuorma vastaan yhteistyörobotilta (kuva 7).



Kuva 4. Rviz-käyttöliittymän näkymä TurtleBotin dynaamisesta reitinsuunnittelusta havaitun uuden esteen kiertämiseksi (Kuva: Hannu Hakalahti 2024).



Kuva 5. Autonomiset ajoneuvot hankkeessa kehitetty kevyt graafinen käyttöliittymä TurtleBot-mobiilirobotille. Käyttöliittymä näyttää reaaliaikaisen videokuvan robotilta (Kuva: Hannu Hakalahti 2024).



Kuva 6. Hankkeen demona kehitetty socket-viestintä yhteistyörobottien ja TurtleBotin välillä (Kuva: Janne Kapela 2024).

Autonomiset ajoneuvot -selvityshanke päättyy kesäkuussa 2024. Hankkeen rahoituksella hankittu TurtleBot 4

-mobiilirobotti otetaan SeAMKissa käyttöön ROS2-ohjelmiston opetusalustana ja sitä tullaan hyödyntämään myös autonomisen toiminnan perusteiden opetuksessa. Hankkeen myötä SeAMKiin kertynyt osaamispääoma saa jatkokäyttöä elokuussa 2024 alkavassa, AKKE-rahoitteisessa OpenUGV-hankkeessa (SeAMK, i.a.-a). Kyseisen hankkeen tarkoituksena on tuoda esille ROS2:n ja yleisemmin mobiilirobotiikan ja autonomisten työkoneiden mahdollisuuksia yritysten tuotekehitys- ja tuotantotyössä. Hankkeessa tullaan hyödyntämään ja jatkokehittämään SeAMKin Sähköiset voimalinjat -investointihankkeessa (SeAMK, i.a.-b) hankittavaa ulkokäyttöistä mobiilirobotti-alustaa. Tämä on osa Autonomiset ajoneuvot -hankkeessa alkanutta kehityskaarta pienistä alustoista kohti täyskokoisia työkoneita.

Tommi Ylimäki
SeAMK

Ylimäki toimii TKI-painotteisena lehtorina SeAMKissa. Hän vetää Autonomiset ajoneuvot -selvityshanketta.

Petteri Mäkelä
SeAMK

Mäkelä toimii yliopettajana SeAMK Automaatio ja tietotekniikassa.

Hannu Hakalahti
SeAMK

Hakalahti toimii TKI-asiantuntijana Digitalisaatio ja älykkäät teknologiat -tiimissä.

Lähteet

Ackerman, E. (26.3.2013). *TurtleBot Inventors Tell Us Everything About the Robot*. https://spectrum.ieee.org/interview-turtlebot-inventors-tell-us-everything-about-the-robot

Clearpath Robotics. (i.a.). *TurtleBot 4.* Haettu 3.5.2024, https://clearpathrobotics.com/turtlebot-4/

Clearpath Robotics. (8.2.2022). *User Manual · Turtlebot4 User Manual*. https://turtlebot.github.io/turtlebot4-user-manual/

Elmofty, O. (4.8.2022). ROS2—How is it better than ROS1. Medium. https://medium.com/@oelmofty/ros2-how-is-it-better-than-ros1-881632e1979a

iRobot Education. (i.a.). *Create 3 Robot*. Haettu 8.5.2024, https://edu.irobot.com/what-we-offer/create3

Macenski, S., Foote, T., Gerkey, B., Lalancette, C., & Woodall, W. (2022). Robot Operating System 2: Design, architecture, and uses in the wild. *Science Robotics*, *7*(66), eabm6074. https://doi.org/10.1126/scirobotics.abm6074

Machado, T., Ahonen, A., & Ghabcheloo, R. (2021). Towards a Standard Taxonomy for Levels of Automation

in Heavy-Duty Mobile Machinery. *Proceedings of ASME/BATH 2021 Symposium on Fluid Power and Motion Control*. https://doi.org/10.1115/FPMC2021-70251

ROS. (i.a.). *Welcome to the ROS Community*. Haettu 3.5.2024, https://www.ros.org/blog/community/

SeAMK. (i.a.-a). *OpenUGV.* Haettu 8.5.2024, https://projektit.app-staging.seamk.hion.dev/openugv/

SeAMK. (i.a.-b). *Sähköiset voimalinjat*. Haettu 8.5.204, https://projektit.app-staging.seamk.hion.dev/alykkaat-teknologiat/sahkoiset-voimalinjat/

Schierenebck, T. (14.10.2020). Creating a Gazebo with ROS2 for you own robot. *Medium*. https://medium.com/creating-a-gazebo-simulation-with-ros2-for-your/introduction-8daf6efa12f4

Shanmugam, J. (22.12.2022a). ROS2 from the Ground Up: Part 1 — An Introduction to the Robot Operating System. *Medium*. https://medium.com/@nullbyte.in/ros2-from-the-ground-up-part-1-an-introduction-to-the-robot-operating-system-4c2065c5e032

Shanmugam, J. (23.12.2022b). ROS2 from the Ground Up: Part 2 — Publishers and Subscribers. *Medium*. https://medium.com/@nullbyte.in/ros2-from-the-ground-up-part-2-publishers-and-subscribers-8deda54927c7

Ylimäki, T. (12.4.2024). *Työkoneet kohti autonomista toimintakykyä*. https://urn.fi/URN:NBN:fi-fe2024041216885