



Széchenyi István Egyetem

ROS (NEM HIVATALOS) DOKUMENTÁCIÓ – NEOBOTIX MP500

2012. december 16.

Ros (nem hivatalos) Dokumentáció – Neobotix MP500	1
A Ros bemutatása	3
A Robotok bemutatása	3
ROS telepítés	3
ROS kiadások	3
Telepítési hibák	4
Parancsok	5
Parancssori eszközök	5
ROS "szemantika" - Elnevezések	6
Neobotix ROS stack-ek	7
Neobotix stack-ek telepítése	8
Kinect	11
RVIZ	11
Ssh	11
Rosbridge	11
Egyéb	12
Irodalomjegyzék, olvasnivaló, linkek	13

A ROS BEMUTATÁSA

- A ROS (Robot Operating System) egy meta-operációsrendszer, amely könyvtárai és eszközei segítségével teszi lehetővé a szoftver-fejlesztők számára robot alkalmazások fejlesztését.
- Többek közt hardver absztrakciót, hardver-illesztőket, könyvtárakat, megjelenítőket, üzenetküldési funkciókat, package managementet biztosít. A ROS open source, BSD licenc alá esik.

A ROBOTOK BEMUTATÁSA

Az UT111-ben megtalálható 4 Neobotix robot (Neo1, Neo2, Neo3, Neo4) mind az MP-500-as szériába tartozik, egész pontosan MPS-500 v 2.3.1. A labor az Automatizálási tanszék Intelligens és Autonóm robotok laborja, a 4 Neobotix mobil roboton kívül több robot tartozik ide.

A Neobotix honlapja: www.neobotix-roboter.de/mobile-roboter-mp-500.html

ROS TELEPÍTÉS

Robot Operating System

<http://nootrix.com/2012/05/ros-installation/>
<http://www.ros.org/wiki/fuerte/Installation/Ubuntu>

Ez egy régebbi (Electric) telepítését mutatja. A Neobotix Gmbh ajánlotta ezt a verziót, ugyanis a stack-ek csak Ubuntu 10.04 + ROS Electric-en lettek tesztelve. A Neo1-es roboton is ez a konfiguráció van gyárilag, azonban sikerült a 12.04-re frissítés.

Ubuntu 10.04 (Lucid)

1. Ubuntu konfigurálása "restricted"-re
2. `sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu lucid main" > /etc/apt/sources.list.d/ros-latest.list'`
3. `wget http://packages.ros.org/ros.key -O - | sudo apt-key add -`
4. `sudo apt-get update`
5. `sudo apt-get install ros-electric-desktop-full`
6. `echo "source /opt/ros/electric/setup.bash" >> ~/.bashrc`
`. ~/.bashrc`

Sudo password: neobotix

ROS KIADÁSOK

- Box Turtle (2010 Február)
- C Turtle (2010 Augusztus)
- Diamondback (2011 Február)
- Electric Emys (2011 Augusztus)
- Fuerte Turtle (2012 Március)
- Groovy Galapagos (2012 Október)

TELEPÍTÉSI HIBÁK

Missing Dependency: std_msgs

```
Warning: Current working directory is not on ROS_PACKAGE_PATH
```

Ellenőrizzük echo kiíratással, milyen környezeti változónk van:

```
echo $ROS_PACKAGE_PATH
```

Megoldás: ROS_PACKAGE_PATH enviroment variable-be hozzáadni kettősponttal (:) azt mappát, ami majd tartalmazni fogja a stackjainket jelen esetben a /home/neobotix

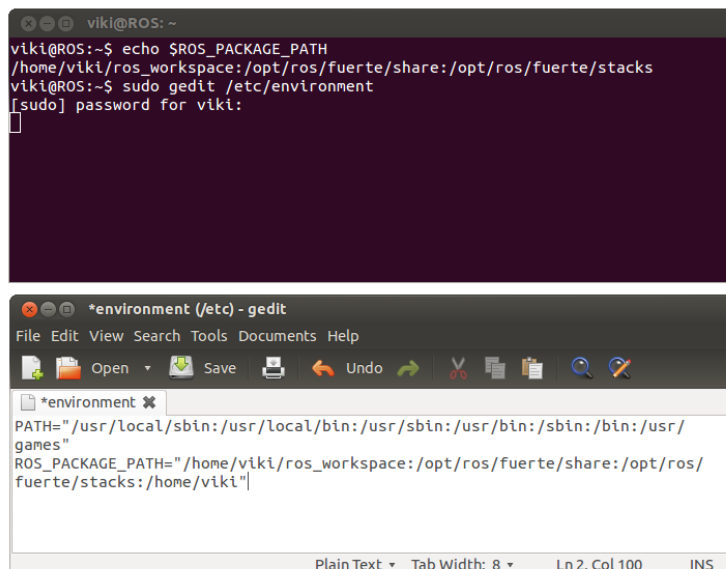
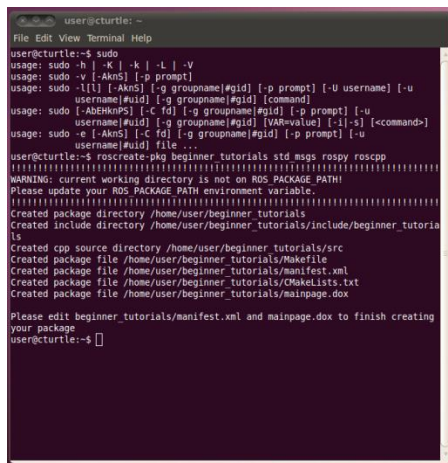
```
export ROS_PACKAGE_PATH=/home/neobotix:$ROS_PACKAGE_PATH
```

Egyéb példák:

```
ROS_PACKAGE_PATH=/opt/ros/electric/stacks:/home/neobotixroswtf
```

```
ROS_PACKAGE_PATH=/opt/ros/crurtle/stacks:/home/user
```

```
ROS_PACKAGE_PATH=/opt/ros/fuerte/stacks:/home/wiki
```



Localhost beállítása (pl. tutorialok alatt)

```
$ export ROS_HOSTNAME=localhost
$ export ROS_MASTER_URI=http://localhost:11311
```

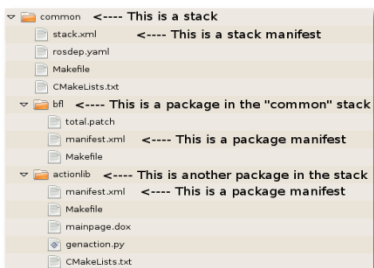
PARANCSONOK

Parancs	Jelenése
echo \$ROS_PACKAGE_PATH echo \$ROBOT	a ROS környezeti változók (Részletesen a Telepítési hibák részben) <i>/home/your_user_name/forte_workspace/sandbox:/opt/ros/forte/share:/opt/ros/forte/stacks</i>
export ROS_PACKAGE_PATH= export ROBOT=mp_500_2_1	az előző környezeti változók beállítására használatos
rospack profile	Kilistázza az összes létező stacket
rospack find stacknév	Megkeresi az útvonalát a stacknek
roscd stacknév	Oda ugrik
rospack depends stacknév	Függőségek
roscrcat-pkg [package_név] [depend1] [depend2] [depend3]	Package készítése, név után a létező package-tól való függés megadása
roscrcat Beginner_Tutorials	elkészíti a 'Beginner_Tutorials' stack-et
rospack find stacknév	Megkeresi az útvonalát a stacknek
roscrcat	Roscrcat = Master + rosout + parameter server (ezután új terminálban futtatni: roscrcat list)
roscrcat list	listázza a futó node-okat (a roscrcat , ami mindig fut, ez gyűjti össze a többi node debug kimenetét)
roscrcat info /rosout	a rosout node-ről ad infókat (publikációk, feliratkozás /subscribe/, szervízek)
roscrcat turtlesim	buildeli a 'turtlesim'-et
roscrcat turtlesim turtlesim_node	futtatja a turtlesimet run ctrl+c leállít
turtlesim/bin	>> futtatható állományok
roscrcat turtlesim draw_square	négyzetet rajzol ki
rxgraph	dinamikus grafikon, arról, ami épp a ROS rendszerében folyik
rostopic list	a létező topicokat listázza
rostopic echo [topic]	Mutatja a topic által publikált adatokat pl. rostopic echo /turtle1/command_velocity

PARANCSSORI ESZKÖZÖK

roscrcat: ezzel lehet "visszajátszani" mi történt, ha egy szituációban valami baj volt

- Package
 - a legalacsonyabb szintű ROS csoportosítás
 - Manifest: a Package leírására szolgál, legfontosabb feladata a Package függőségek leírása
- Stack
 - több Package összessége
 - Stack Manifest: a Stack leírására szolgál, legfontosabb feladata a Stack függőségek leírása
- Nodes: olyan futtatható állomány, amely a ROS-on keresztül kommunikál másik node-okkal
- Messages: üzenetek, ROS adattípusok, amelyeket topic feliratkozásra (subscribing) vagy publikációra (publishing) használunk.
- Topics: A node-ok üzeneteket (messages) publikálhatnak egy topic-nak valamint feliratkozhatnak (subscribe) egy topic-ra, hogy üzeneteket kapjanak.
- Master: Névszolgálat a ROS-hoz (pl. segít a node-oknak egymás megtalálásában)
- roscore: ROS-os megfelelője az stdout/stderr-nek ez mindig fut, ez gyűjti össze a többi node debug kimenetét (a roscore maga is egy node)
- roscore: Master + roscore + parameter server



NEOBOTIX ROS STACK-EK

Stack	Jelenése
neo_driver	Ez a stack harver drivereket tartalmaz Neobotix robotok számára
neo_common	Neobotix robotok leírásai
neo_apps	Tool-okat tartalmaz, amelyekkel lehet vezérelni lehet a robotot
neo_navigation	A navigáció konfigurálására Ezen belül Package: neo_2dnav_mp_500 : példa konfiguráció az mp-500 2D navigálására neo_2dnav_tools : a move_base-nek küld célokat
neo_simulation	Példa térképek, szimulációk konfigurálása

ROBOT környezeti változó beállítása:

```
export ROBOT=mp_500_0_9 inkább:  
export ROBOT=mp_500_2_1
```

```
roscore
```

Új terminálban (scriptek futtatása):

```
roscd neo_apps  
rosmake neo_bring_up  
roslaunch neo_bring_up mp_500.launch
```

Vagy ezt új terminálban (billentyűzetről irányítás):

```
roscd neo_apps  
rosmake neo_teleop  
roslaunch neo_teleop teleop_keyboard.launch
```

Új terminálban:

```
rostopic list  
rosnode list
```

Esetleg ellenőrzésül, hogy minden települt-e:

```
rosstack list
```

akkor a neo_ kezdetű stackjeinknek listázva kell lenni.

```
rxgraph  
rostopic list  
rostopic echo /joy joystick gombok kiírása  
rostopic echo /base_controller/command
```

/home/neobotix/ros_workspace/neo_driver/neo_serrelayboard/ros/launch/mp_500_2_1.yaml -> ComPort: /dev/ttyUSB0
helyett ComPort: /dev/ttyUSB1

float64[2] angularVelocity
bool[2] driveActive
bool[2] quickStop
bool[2] disableBrake

```
rostopic pub /DriveCommands neo_serrelayboard/DriveCommas  
'{angularVelocity:[0.1,0.1],driveActive:[1,1],quickStop:[0,0],disableBrake:[1,1]}'  
  
rostopic pub -1 /DriveCommands neo_serrelayboard/DriveCommas --  
'{angularVelocity:[0.1,0.1],driveActive:[1,1],quickStop:[0,0],disableBrake:[1,1]}'
```

angularVelocity: rad/s az egység

A lentiek pedig milliméterben értendők.

test "sugár" (középponttól a kerék közepéig):	592/2 - 85/2
teljes szélesség:	592
kerék teljes szélesség:	85
kerék futófelület szélessége:	70
kerék sugár:	130
két kerék távolsága (két kerék futófelületének a távolsága):	592-85 = 507 HELYETT 515!!!!!!

http://www.pr2-remotelab.com/doku.php?id=using_the_remote_lab_infrastructure_using_the_ros_simulator

NEOBOTIX STACK-EK TELEPÍTÉSE

Udpate: egyszerűbb a teljes ros_workspace-t felrakni, ami a sze.hu/~hern oldalon megtalálható. (A függőségeket természetesen ebben az esetben is telepíteni kell)

A következő leírás a klasszikus telepítést mutatja:

Bizonyosodjunk meg róla, hogy a ROS is telepítve van és a git is:

```
sudo apt-get install git-core curl
```

```

viki@ROS: ~
[630 kB]
Get:4 http://ca.archive.ubuntu.com/ubuntu/ precise/main git i386 1:1.7.9.5-1 [5,
963 kB]
Get:5 http://ca.archive.ubuntu.com/ubuntu/ precise/main git-core all 1:1.7.9.5-1
[1,384 B]
Fetched 6,756 kB in 5s (1,250 kB/s)
Selecting previously unselected package curl.
(Reading database ... 197713 files and directories currently installed.)
Unpacking curl (from .../curl_7.22.0-3ubuntu4_i386.deb) ...
Selecting previously unselected package liberror-perl.
Unpacking liberror-perl (from .../liberror-perl_0.17-1_all.deb) ...
Selecting previously unselected package git-man.
Unpacking git-man (from .../git-man_1%3ai.7.9.5-1_all.deb) ...
Selecting previously unselected package git.
Unpacking git (from .../git_1%3ai.7.9.5-1_i386.deb) ...
Selecting previously unselected package git-core.
Unpacking git-core (from .../git-core_1%3ai.7.9.5-1_all.deb) ...
Processing triggers for man-db ...
Setting up curl (7.22.0-3ubuntu4) ...
Setting up liberror-perl (0.17-1) ...
Setting up git-man (1:1.7.9.5-1) ...
Setting up git (1:1.7.9.5-1) ...
Setting up git-core (1:1.7.9.5-1) ...
viki@ROS:~$

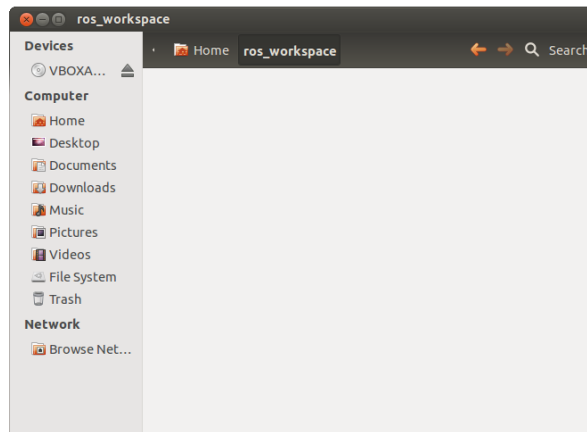
```

ROS workspace készítése:

```

mkdir ~/ros_workspace
export ROS_PACKAGE_PATH=~/ros_workspace:$ROS_PACKAGE_PATH
export ROS_PACKAGE_PATH=/home/user/ros_workspace:$ROS_PACKAGE_PATH

```



Ideiglenesen adjuk hozzá workspace-t a bashrc típushoz:

```

echo ROS_PACKAGE_PATH=~/ros_workspace:$ROS_PACKAGE_PATH >> ~/.bashrc

```

Ha a stack-eket szeretnénk telepíteni, klónozzhatjuk a [github](https://github.com) -ról.

```

cd ~/ros_workspace
git clone https://github.com/neobotix/neo_driver.git

```

```

viki@ROS: ~/ros_workspace
viki@ROS:~/ros_workspace$ git clone https://github.com/neobotix/neo_driver.git
Cloning into 'neo_driver'...
remote: Counting objects: 685, done.
remote: Compressing objects: 100% (432/432), done.
remote: Total 685 (delta 342), reused 524 (delta 181)
Receiving objects: 100% (685/685), 165.16 KiB | 147 KiB/s, done.
Resolving deltas: 100% (342/342), done.
viki@ROS:~/ros_workspace$

```

Először a függő stack-eket kell feltenni: [cob_driver](#), [cob_common](#), [joy](#), [erratic_gazebo_plugins](#) és [pr2_gazebo_plugins](#), amik **nem** az alapértelmezett ROS telepítés részei. Így:

```

sudo apt-get install ros-electric-erratic-robot ros-electric-cob-driver ros-electric-cob-common ros-electric-pr2-simulator ros-electric-joystick-drivers

```

Vagy így:

```
git clone git://github.com/ipa320/cob_driver.git
git clone git://github.com/ipa320/cob_common.git
svn co https://code.ros.org/svn/ros-pkg/stacks/joystick_drivers/trunk/joy
svn co https://ua-ros-
pkg.googlecode.com/svn/stacks/erratic_robot/trunk/erratic_gazebo_plugins
svn co https://code.ros.org/svn/wg-ros-pkg/stacks/pr2_simulator/trunk/pr2_gazebo_plugins
```

A teljes telepítéshez kell: [neo_driver](#), [neo_apps](#), [neo_common](#), [neo_navigation](#) és [neo_simulation](#).

```
git clone https://github.com/neobotix/neo_driver.git
git clone https://github.com/neobotix/neo_common.git
git clone https://github.com/neobotix/neo_navigation.git
git clone https://github.com/neobotix/neo_simulation.git
```

Amíg a neo_driver and cob_driver nincs teljesen egyesítve és az új cob_driver nem lesz apt cob_driver addig manuálisan telepíthető [innen](#).

Mielőtt a Stackeket fordítanánk, érdemes a

- neo_common mappából a:
 - neo_driver_description_mpo_500 és a
 - neo_driver_description_mpo_700 valamint a
- neo_driver mappából a:
 - neo_driver_description_mpo_500 és a
 - neo_driver_description_mpo_700

kimásolni, oda ahol a telepítő nem látja (a PATH-on kívülre). Ezek a fajta robotok úgysem kellenek.

Szintén érdemes kikommentezni a neo_driver/neo_watchdogs/CMakeList.txt-ből ezt a sort:

rosbuild add_executable(watch_temp_node ros/src/watch_temp.cpp). Ezáltal nem lesznek elérhetőek a hőmérséklet adatok, viszont nem lesz fordítási hibánk se.

Neobotix stack-ek fordításához:

```
roscd ${neobotix_stack}
rosmake ${neobotix_stack}

roscd neo_common
rosmake neo_common
roscd neo_driver
rosmake neo_driver
roscd neo_apps
rosmake neo_apps roscd
neo_navigation
rosmake neo_navigation
roscd neo_simulation
rosmake neo_simulation
```

Végül ellenőrizzük, hogy a neo_ kezdetű stackek-et látja-e a ros:

```
rostack list
```

KINECT

http://www.ros.org/wiki/openni_kinect

RVIZ

1. Terminálba: `roscore`
2. Terminálba: `export ROBOT=mp_500_2_1`
`roslaunch neo_description_mp_500 mp_500_empty_world.launch`
3. Terminálba: `roslaunch rviz rviz`

Fixed frame-t /base_link-re állítani, beimportálni (Add) a RobotModel-t.

SSH

Bejelentkezés SSH-val a robotra:

```
ssh neobotix@192.168.0.151
```

Ezután egy megerősítést kér, majd a jelszót, ami "neobotix".

ROSBRIDGE

Telepítés v1.0

```
sudo apt-get install ros-fuerte-brown-remotelab
```

Telepítés v2.0

```
http://kforge.ros.org/rosbridge/git
```

Futtatás v1.0

```
roslaunch rosbridge rosbridge.py
```

Futtatás v2.0

```
roslaunch rosbridge_server rosbridge.py
```

```
rostopic pub /turtle1/command_velocity turtlesim/Velocity -- 1 0  
'/turtle1/command_velocity', 'turtlesim/Velocity', '{"linear":1, "angular":0}'
```

```
rostopic pub /DriveCommands neo_serrelayboard/DriveCommas  
'{angularVelocity:[0.1,0.1],driveActive:[1,1],quickStop:[0,0],disableBrake:[1,1]}'  
  
'/DriveCommands', 'neo_serrelayboard/DriveCommas',  
'{"angularVelocity":[0.1,0.1],"driveActive":[1,1],"quickStop":[0,0],"disableBrake":[1,1]}'
```

```
$ rosmmsg show turtlesim/Velocity  
float32 linear  
float32 angular
```

EGYÉB

```
ssh neobotix@192.168.0.151
sudo apt-get install synaptic
viki@ROS:~/ros_workspace$ svn co http://rovis.unitbv.ro/rovis_ros
env | grep ROS
sudo apt-get install eclipse-cdt
svn up
svn co http://rovis.unitbv.ro/rovis_ros
sudo apt-get install synaptic
sudo apt-get install cmake-qt-gui
sudo apt-get install grap
rostopic list | grep move_base/goal
env | grep ros
cmake-gui
wget http://packages.ros.org/ros.key -O - | sudo apt-key add -
sudo shutdown -h now
```

A parancs nem blokkolja a terminált ha a végére space után egy "&" jelet írunk, így lehet a roscore után futtatni a bring_up-ot, stb.

- ros-sze.tumblr.com
- www.ros.org/wiki
- www.ros.org/wiki/ROS/Tutorials
- www.ros.org/wiki/neo_driver
- ros-sze.tumblr.com
- www.sze.hu/~hernon/NGB_IN039_1/docs
- nootrix.com