

Robotics and
Embedded Systems



Applied Computer Vision for Robotics

29.10.2012

Philipp Heise

Brian Jensen

Sebastian Klose

Syllabus



- ✦ **Phase 1 - Assignments**
~4 sheets (until week before christmas)

- ✦ **Phase 2 - Projects**
weekly status presentations of project groups

GitHub & Teams



Robotics and
Embedded Systems



- ✦ We are going to use private github repositories
- ✦ Create an account if you don't have one yet (free)
 - ✦ use git-config to setup committer names and e-mail on your local machines!
- ✦ Team up - 3 people per team (choose a creative name)
- ✦ We need to know your github names to give you access to the repositories for the sheets!
- ✦ mailing list: rvc@mailknoll.informatik.tu-muenchen.de

Assignments

- ✦ Sheet will go online every second monday
- ✦ Current exercise sheet is available from now:
<http://www6.in.tum.de/Main/TeachingWs2012ComputerVisionRobotics>
- ✦ We'll give short tutorials about the topics here
(03.07.023)
- ✦ Afterwards, you can work in the lab (03.07.011), were
at least one of us will be present to help you, if you got
questions / problems

Submission

- ✦ Each group gets a private repository from us
- ✦ initially clone from our repository to have the same root!
- ✦ Workflow:
 - ✦ create a folder for each sheet
 - ✦ commit locally to your private clone

The Lab



Robotics and
Embedded Systems

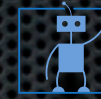


- ✦ currently 5 machines in the lab (dedicated to us)
 - ✦ Different hardware configuration
 - ✦ Ubuntu LTS 12.04
 - ✦ ROS Fuerte
- ✦ Who needs access?
- ✦ Who works with own Notebook?

Sheet 1

- ✦ Getting Familiar with ROS
- ✦ Working with images: Publishing, Subscription, Manipulation
- ✦ Simple Feature Detection: Harris Corners
- ✦ Nodelets
- ✦ Visualization with rViz
- ✦ Camera Calibration using ROS

ROS



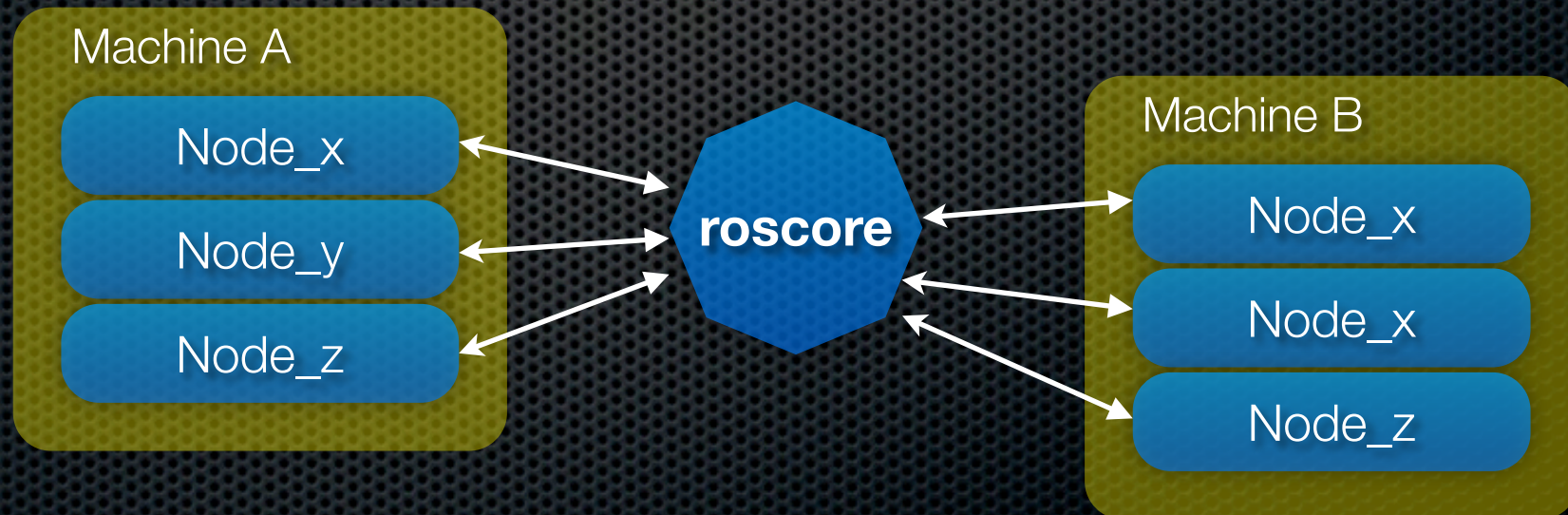
Robotics and
Embedded Systems



- ✦ Robot Operating System
 - ✦ core developed by Willow Garage
 - ✦ Extended by world-wide Open-Source Community
 - ✦ www.ros.org
 - ✦ www.willowgarage.org

ROS Overview

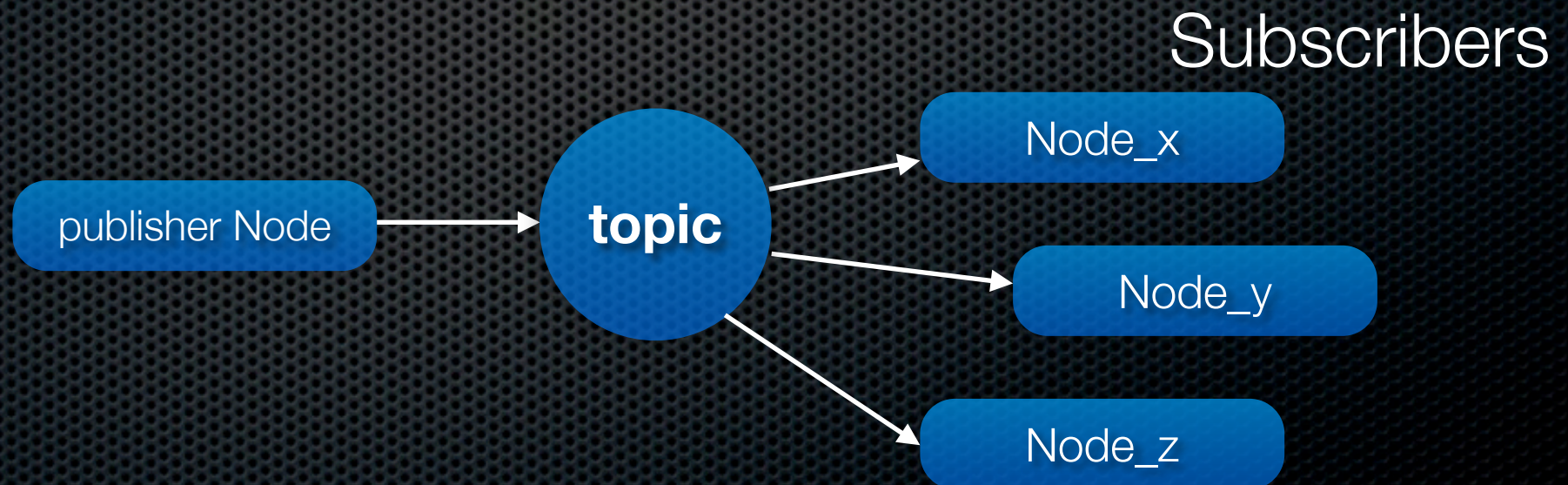
- Modularization via Nodes:
 - process that uses ROS framework
 - nodes may run distributed (different machines, robots, ...)
 - knowledge about others: `roscore` (= *nameserver*)
 - `$ROS_MASTER_URI` specifies which host is running **roscore**



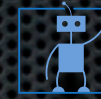
ROS

Internode Com.

- Topics:
 - mechanism to send messages between nodes
 - Publish - Subscribe principle
- Service
 - request / response principle (client/server)



Build System

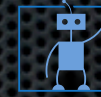


Robotics and
Embedded Systems



- **Package:** self-contained directory containing sources, makefiles, resources, ...
`$ roscreate-pkg foo_package`
 - creates basic build structure in folder *foo_package*
 - rosmake or make to build

Command Line Tools

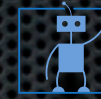


Robotics and
Embedded Systems



- *roscd x - cd to the folder of package x*
- *rostopic ... - information about topics*
- *rosservice ... - information about services*
- *rosviz ... - record / playback data*
- *roslaunch pkg node - run node of package*
- *roscore - start the core*
- *roscpp - parameter server handling*

Harris



Robotics and
Embedded Systems

