Robot Soccer Platform

Using NAO Robots

Group 6:

Jeroen Boonen

Jonas Robers

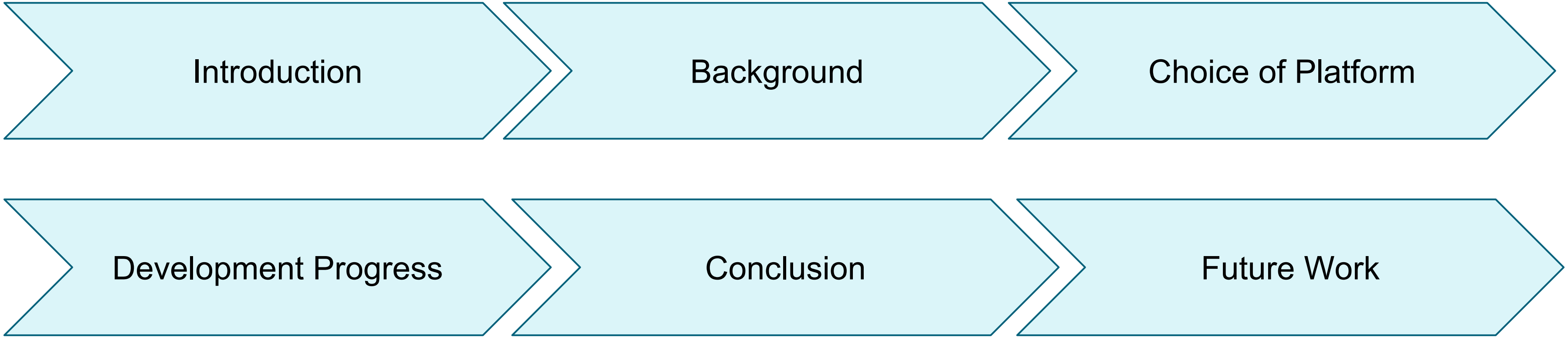
Guy Theunissen

Oliver Lehmacher

Mara Houbraken

Supervisor:
Nico Roos

Outline



Introduction

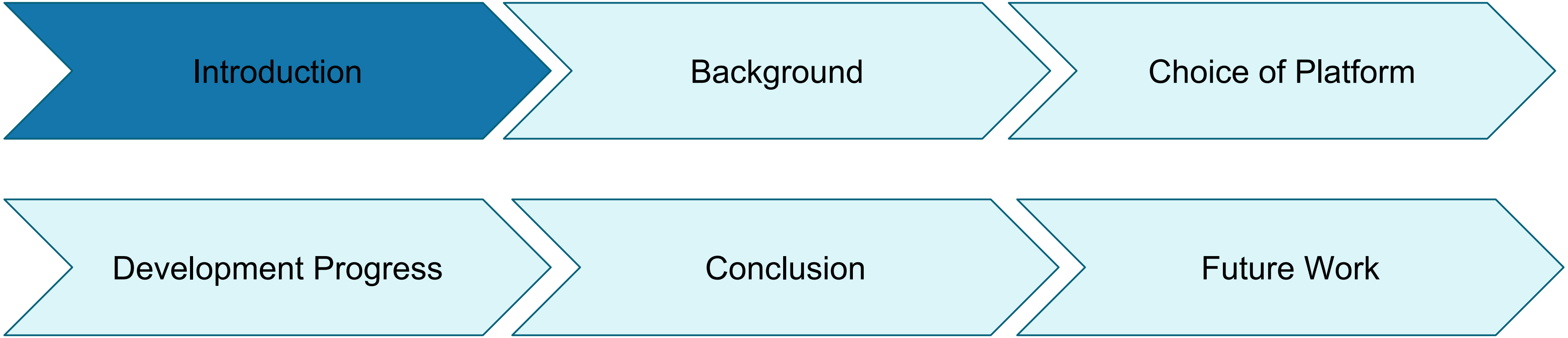
Background

Choice of Platform

Development Progress

Conclusion

Future Work



Introduction

Background

Choice of Platform

Development Progress

Conclusion

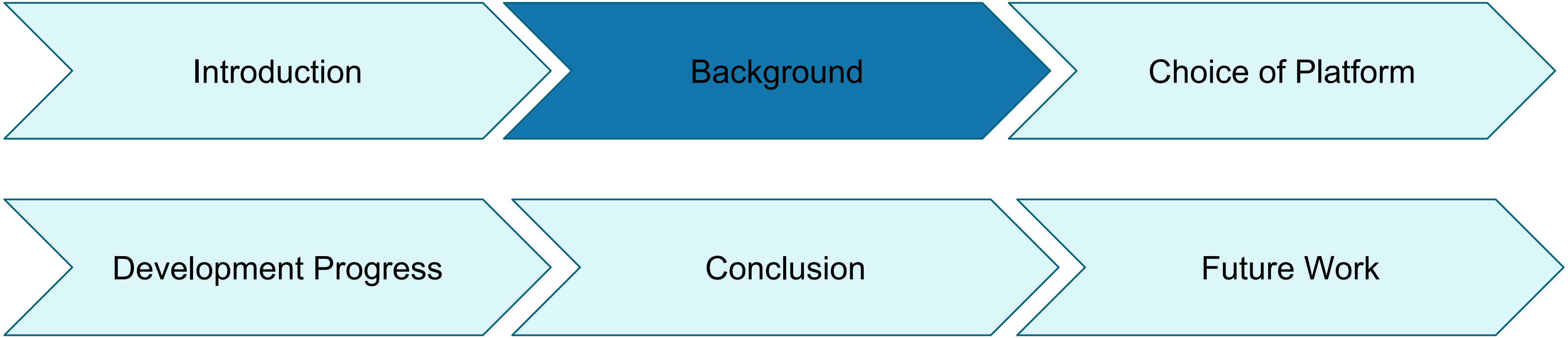
Future Work

Problem Statement

Find and develop a platform to be used by the NAO robots in the RoboCup SPL.

Research Questions

- What is the most appropriate platform for this application?
- What is needed additionally for this platform to work with the NAO Robots?



Introduction

Background

Choice of Platform

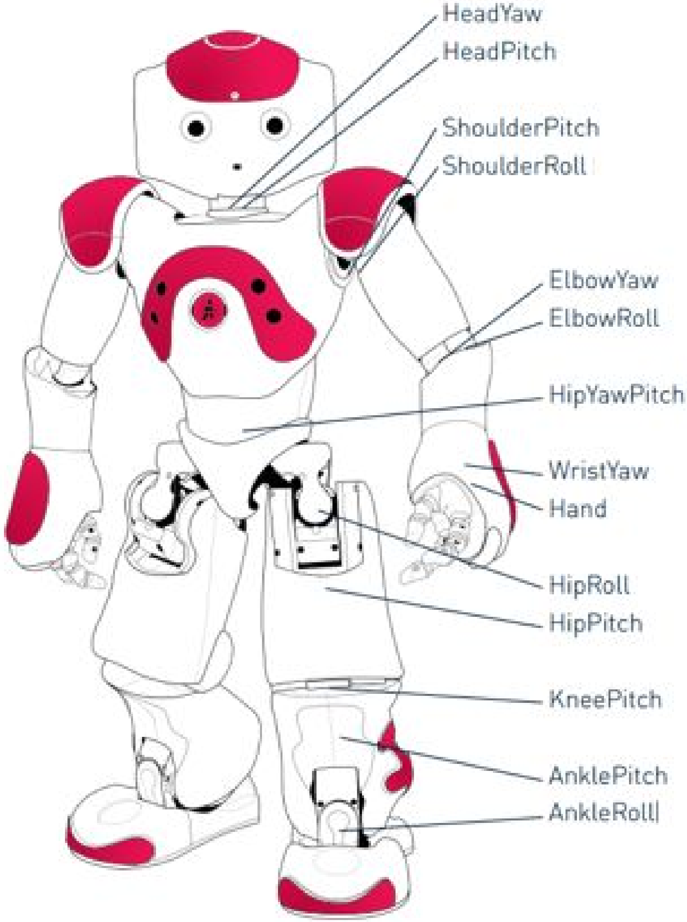
Development Progress

Conclusion

Future Work

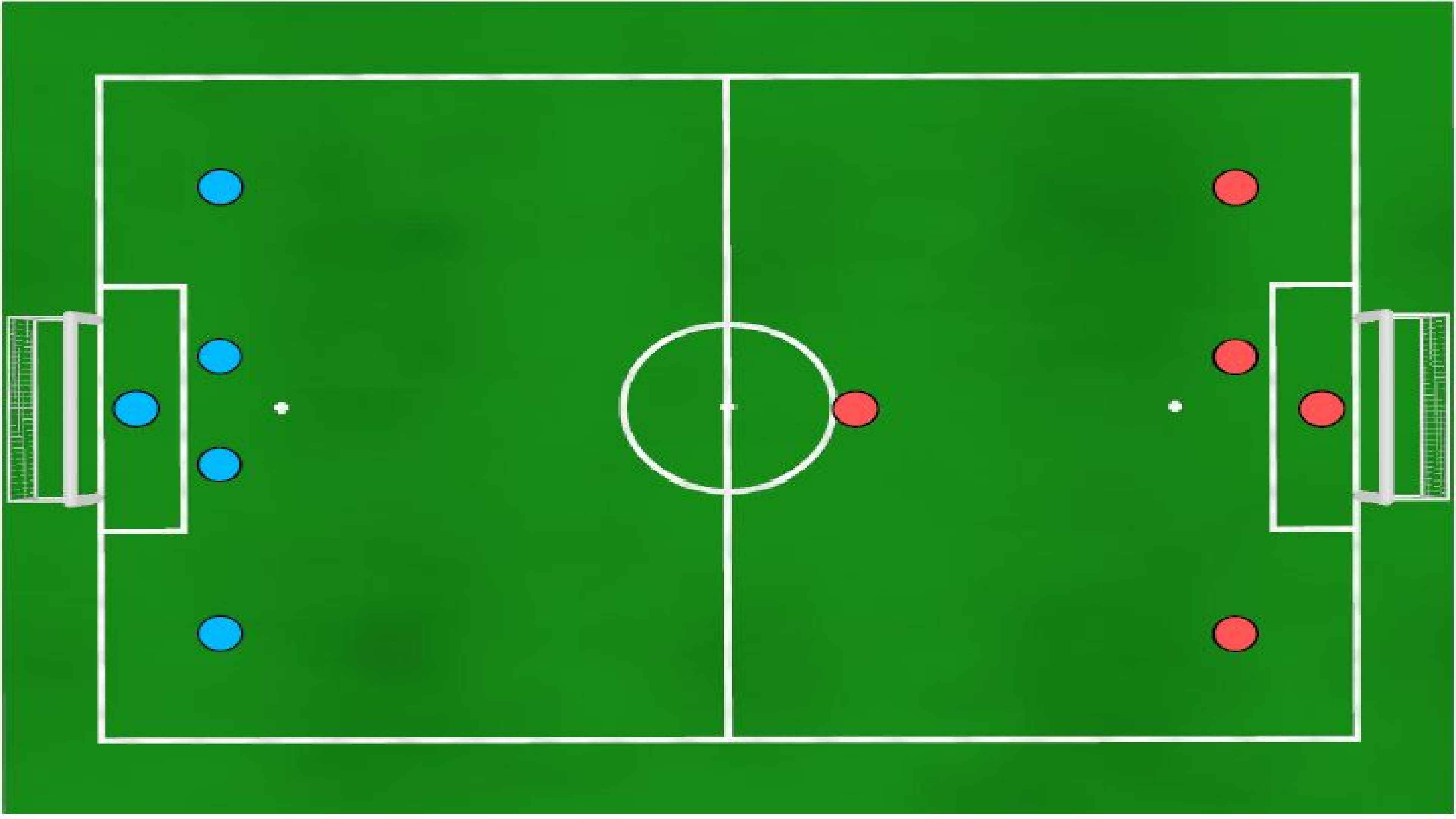
The Robot

- 57.4 x 31.1 x 27.5 cm
- 1.6 GHz CPU, 1 GB RAM
- WiFi, Ethernet & USB
- Speech Recognition
- Facial Recognition
- Touch sensors
- Distance Measurement



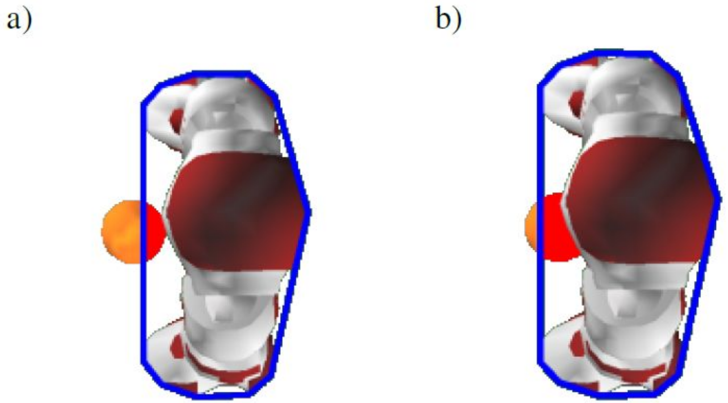
RoboCup

- RoboCup Soccer League
  - International competition
  - Goal: **"By the middle of the 21st century, a team of fully autonomous humanoid robot soccer players shall win a soccer game, complying with the official rules of FIFA, against the winner of the most recent World Cup"**
- Standard Platform League
  - Use NAOs
  - Focus only on Software

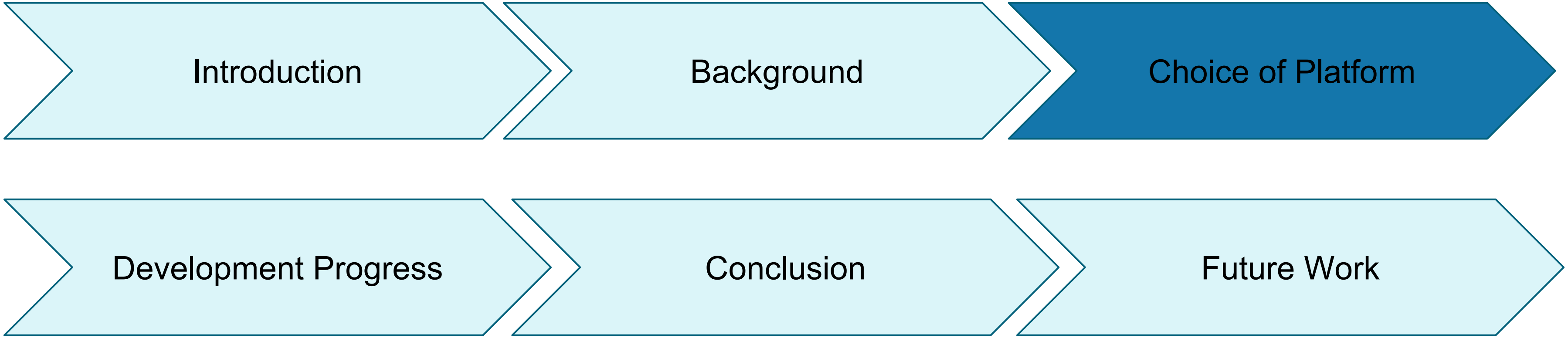


Most relevant Rules

- In case of a tie, there is a penalty shoot out
- Pass at least once after kick-off
- Forbidden actions
  - Pushing other robots
  - Not using bipedal movements (e.g. crawling)
  - Holding the ball
- Penalty for a foul is 45 seconds on the side of the field



ROHM
PHŒNIX CONTACT
UNIVERSITÄT MAGDEBURG



Introduction

Background

Choice of Platform

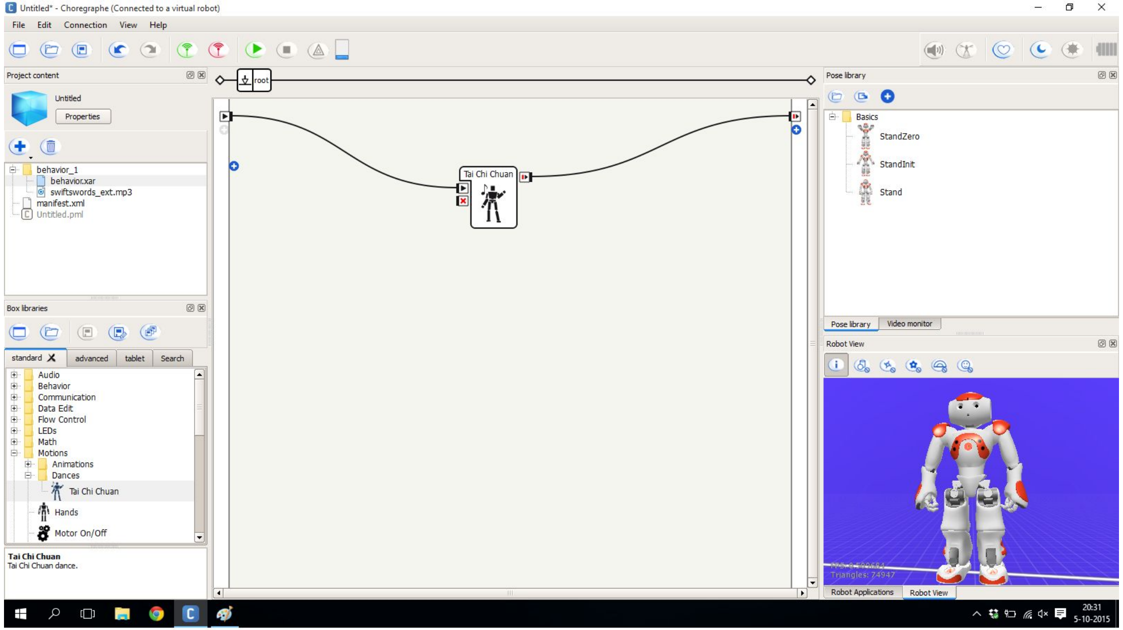
Development Progress

Conclusion

Future Work

Choregraphe

- Platform from Aldebaran
- Uses graphical Interface
- Drag & Drop
- Can include Python / C++



Tai Chi Chuan
Tai Chi Chuan dance.

Choregraphe

Advantages	Disadvantages
+ Easy to understand + Simple drag & drop UI + Easy start up + Compatible with NAO + Contains built-in Simulator	- Number of modules is limited - No specific modules for soccer - Need knowledge of Python or C++ to develop own modules

NAOqi-Framework

- NAOqi: Operating System
- Framework to program the NAO’s
- Provided by Aldebaran
- Python / C++

NAOqi

Advantages	Disadvantages
+ Easy start-up + Compatible with NAO	– Each task needs to be programmed from scratch – Need knowledge of Python or C++ to develop own tasks

B-Human

- Platform by University of Bremen
- Used by the Dutch Nao-Team
- Needs to be updated to support NAO 5

B-Human

RoboCup Standard Platform League

B-Human

Advantages	Disadvantages
+ Ready to use + Well documented + Active support	– Dependent on other teams in the league – Difficult to understand – Difficult to optimize factors

Robot **O**perating **S**ystem

- Open Source Platform
- Set of libraries
  - Messaging
  - Localization
  - Mapping
- C++ and Python

Robot Operating System

Advantages	Disadvantages
+ Open source + Large range of applications + Readily available useful libraries + Large community + Currently standard OS for robotics applications	− Need knowledge of C++ − Need to run Ubuntu or another version of Linux − Steep learning curve − Difficult start up − No compatibility with the NAO

Comparison Overview

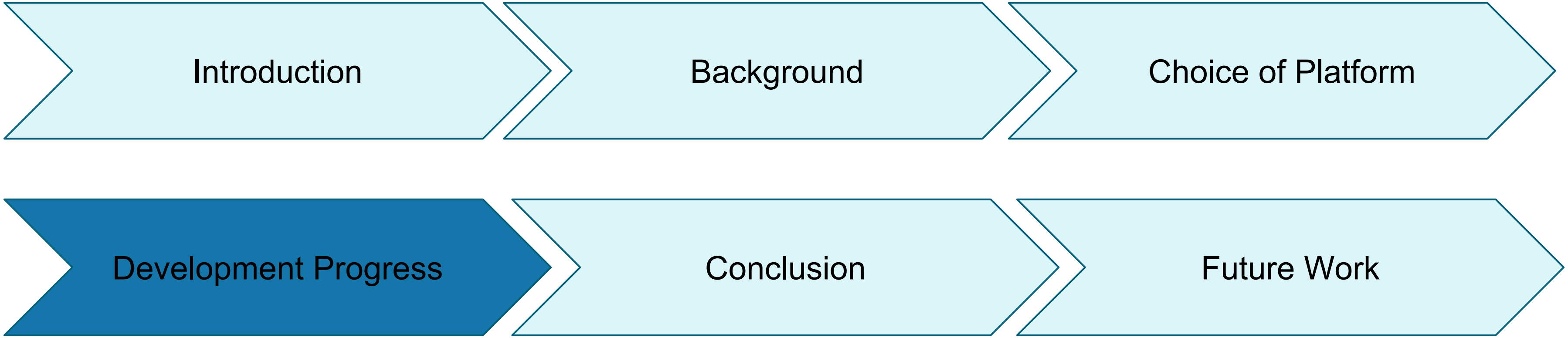
	Advantages	**Disadvantages**
Choregraphe	EASY	LIMITED
NAOqi	COMPATIBLE	LIMITED
B-Human	READY TO USE	DEPENDENT
ROS	SWISSKNIFE	DIFFICULT

Platform decision

- B-Human
  - Best-documented RoboCup SPL framework
  - Not clear how it works
  - Newest version (for NAOqi 2.0.x) released 02-12-2015
  - Dependance on University of Bremen
- ROS
  - Large community
  - Core is well documented
  - Libraries readily available

Migrate away from B-Human to ROS

Decision supported by Dutch Nao Team



Introduction

Background

Choice of Platform

Development Progress

Conclusion

Future Work

Workings of ROS

- Master and nodes
- Master → controls the system
- Node → function (module) within the system
- Master running on pc, nodes can be external
- Topics
- Publishers and subscribers

What is needed: Running ROS on NAO

- Special version of ROS for NAO (Indigo)
- Tutorials available
- RoboCup teams that participated before (UChile, Linköping Humanoids)
- UChile provides an installation script
- Development environment using VM

What is needed: Playing soccer

- Limited communication → run master on each NAO
- Two solutions:
  - Using dedicated package → not available on NAO
  - Using WiFi communication package

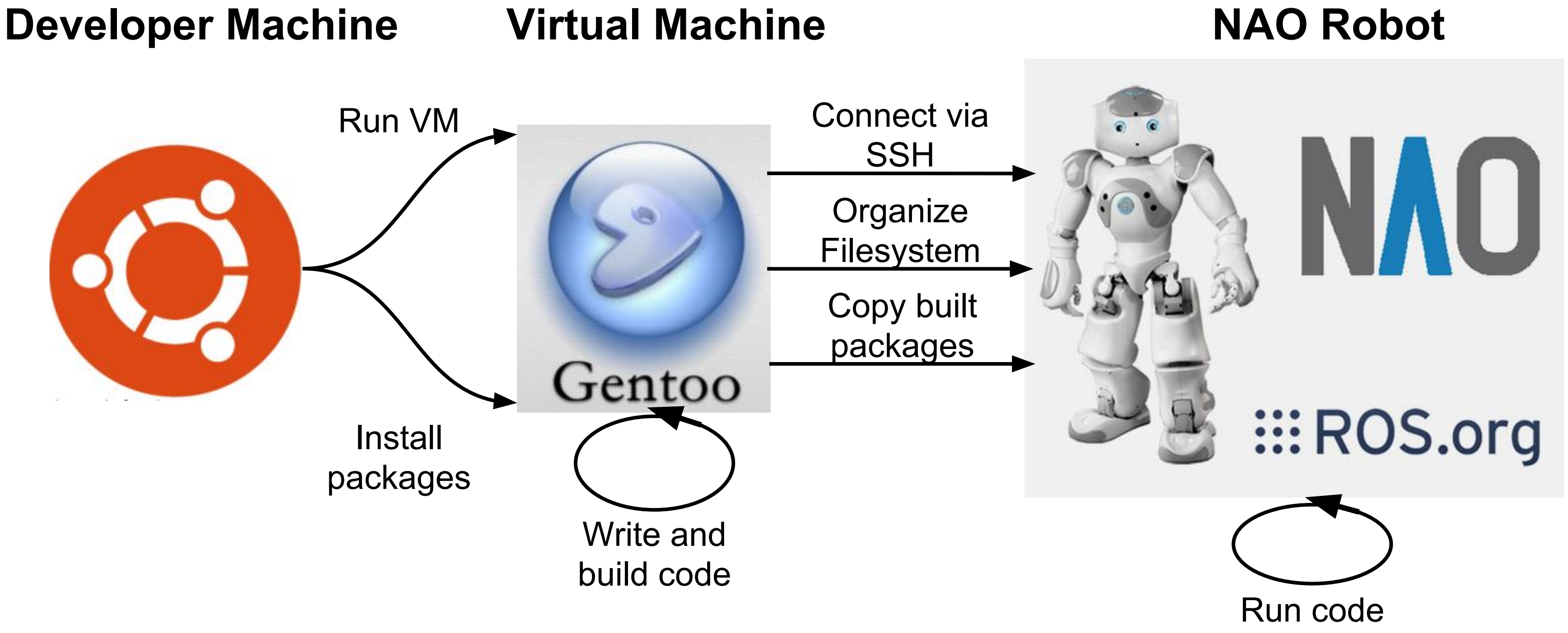
- Encapsulate parts of B-Human as ROS nodes
- Iterative process to move from B-Human code to own code
- Long term process → expected to take several years

- Focus on technical challenges first

Achievements

- ROS master is running on single NAO
- Development environment set up
- Running custom nodes
- Written tutorials for later use

Development process



Developer Machine

Virtual Machine

NAO Robot

Run VM

Install packages

Connect via SSH

Organize Filesystem

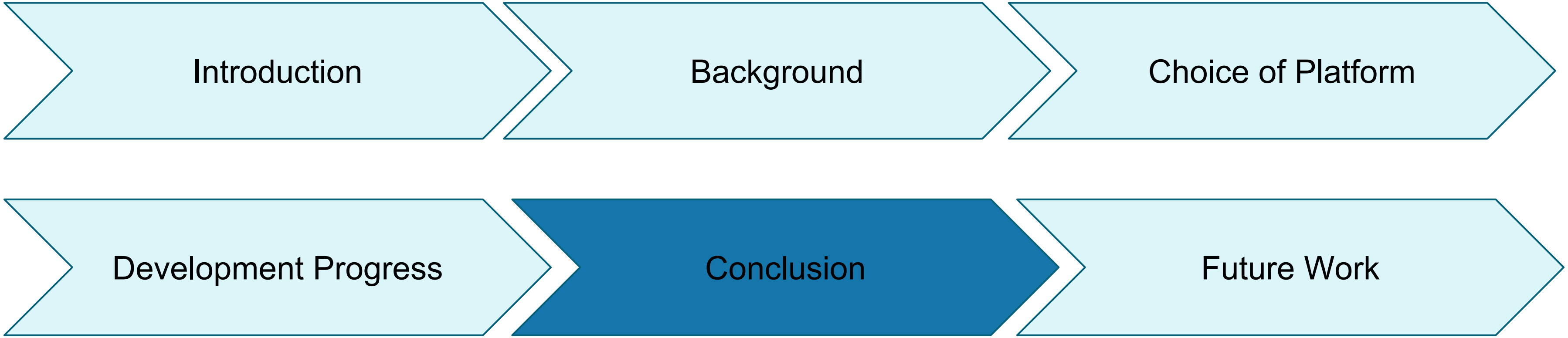
Copy built packages

Write and build code

Run code

Difficulties

- Dependency problems
  - dependencies missing on Gentoo
- Limited disk space on NAO
- Inconsistency of packages
  - some packages only for ROS-Hydro
  - others only for ROS-Indigo
  - newest ROS-Version: Jade
- Some packages not available for ROS on the NAO
- Outdated tutorials



Introduction

Background

Choice of Platform

Development Progress

Conclusion

Future Work

Conclusion: Theory

- NAOqi/Choregraphe
  - too much development effort needed
- B-Human
  - too dependent on another team
- ROS
  - big community
  - well documented

→ ROS is a good choice

Conclusion: Practice

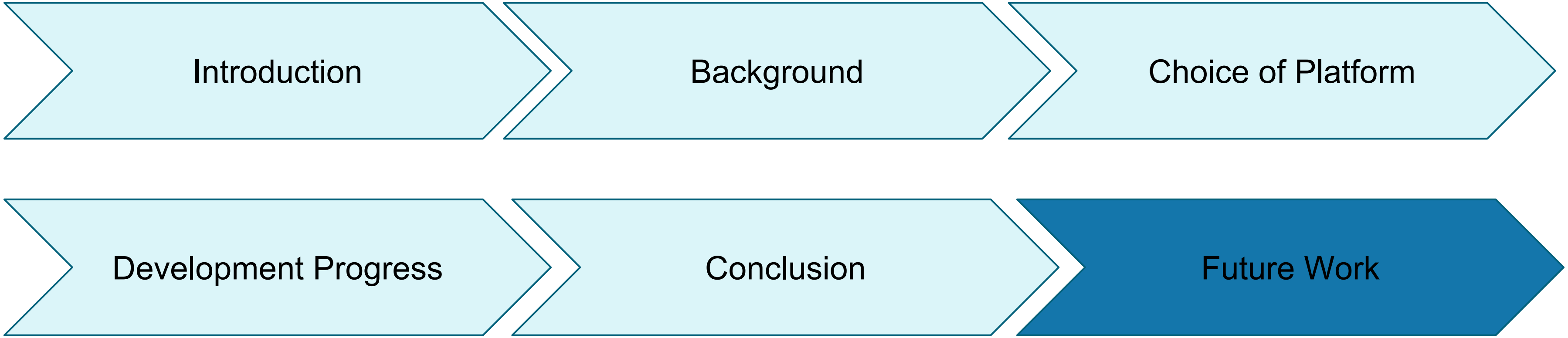
- ROS difficult to integrate to the NAO
- Achievements
    - Install ROS on the NAO
    - Setup development environment
    - Run custom code
- Tutorials mostly outdated
- More research is needed

→ ROS is not a short term solution

Conclusion

- RQ1: *What is the most appropriate platform for this application?*
  - ROS is the most appropriate platform

- RQ2: *What is needed additionally for this platform to work with the NAO robots?*
  - ROS installation
  - Development environment
  - Solution to the multi-master problem
  - Development of soccer-specific modules



Introduction

Background

Choice of Platform

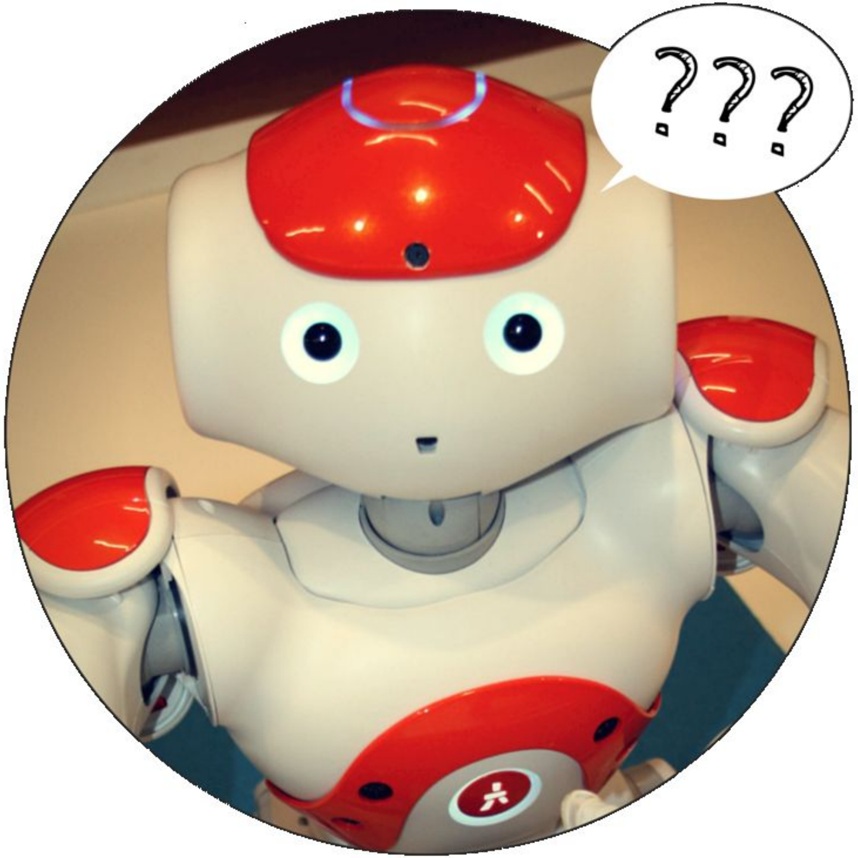
Development Progress

Conclusion

Future Work

Future Work

- Better integration of ROS into the NAO
- Update the tutorials
- Encapsulate modules from B-Human
- Solve multi-master communication
- Cooperate with Dutch Nao Team



Thank you for your attention!