



NAO 3.x



Theme 2: NAO programming in SDK





Table of Content

1. NAO Introduction
2. NAO Mechanic
3. NAO Hardware
4. NAO Software
 - i. Operating System on NAO
 - ii. Framework
 - iii. Monitor
 - iv. Choregraphe
 - v. Webots for NAO
 - vi. SDK (Software Development Kit)
 - vii. Structure of NAOqi
 - viii. SDK installation under MS Windows
 - ix. NAOqi API (Application Programming Interface)
 - x. Code Examples
 - xi. Status quo
5. NAO Sources





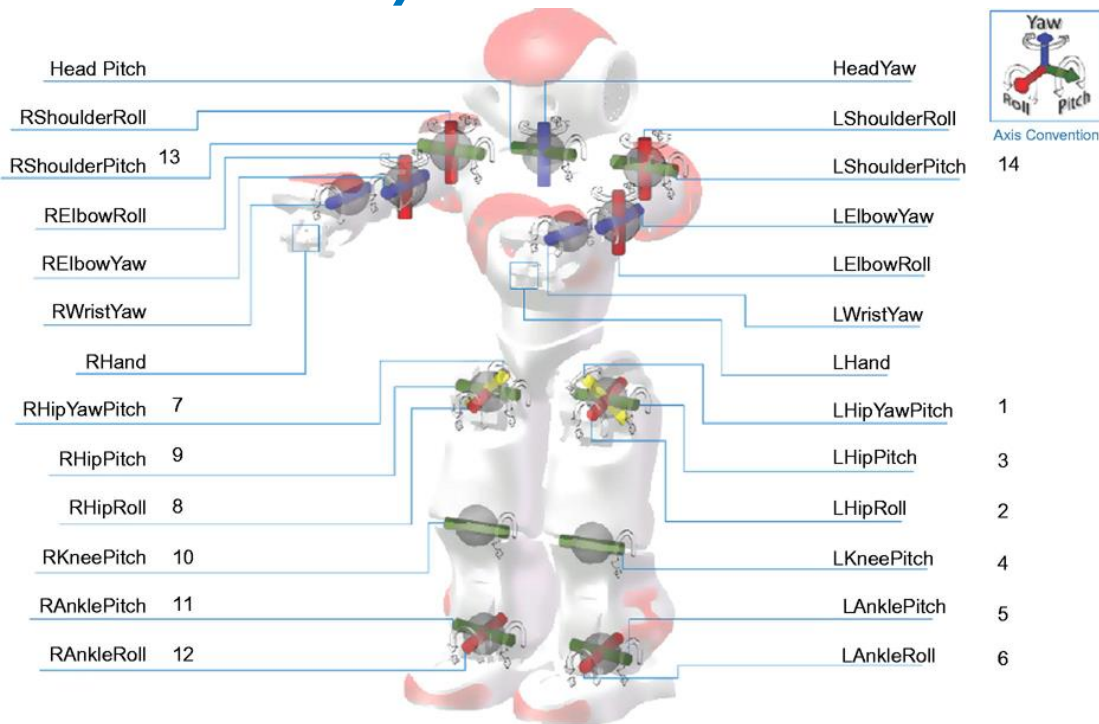
NAO Introduction

- Company name: Aldebaran Robotics
- Headquarter: Paris, France
- Foundet: 2005 by Bruno Maisonnier
- Type: Standardplatform for RoboCup
- License: Open source for Mechanical parts and Software modules, but not for electrical circuits
- NAO Model: Distributed since 2007 as a successor model of Sonys Aibo and still in development

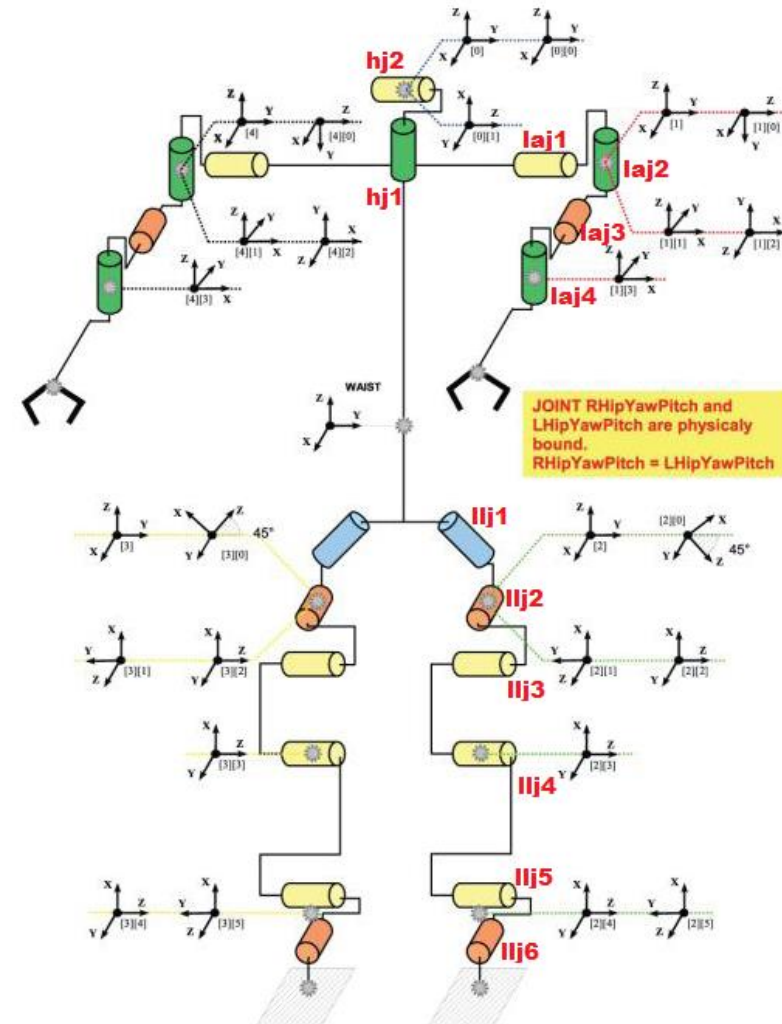




NAO Mechanic – Joints / Axis



- 25Axis = max. 6 degrees of freedom





NAO Mechanic – Modeling

- Aldebaran Robotics supports Mechanical design for NAO by offering, non functional (no transmission), surface models for SolidWorks. In use for Simulation and Shell/Case redesigne.



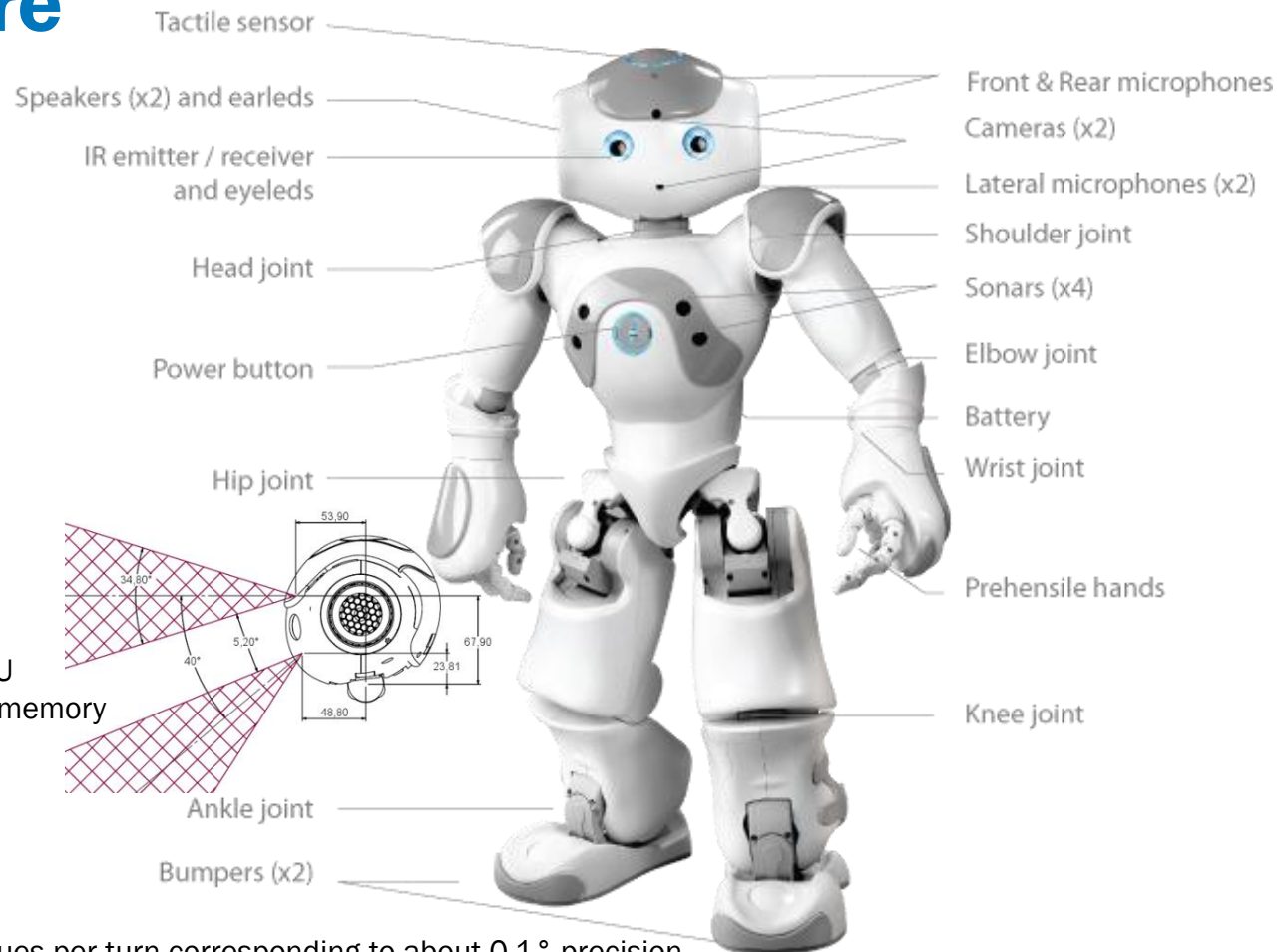
NAO Hardware

Motherboard Specification

- **NAO V. 4.0**
 - ATOM Z530 1.6GHz CPU
 - 1 GB RAM
 - 2 GB flash memory
 - 4 to 8 GB flash memory dedicated to user purposes
- **NAO V. 3.x**
 - x86 AMD GEODE 500MHz CPU
 - 256 MB SDRAM / 2 GB flash memory

Joint position sensors Specification

- 12 bit precision, $2^{12}=4096$ values per turn corresponding to about 0.1° precision.



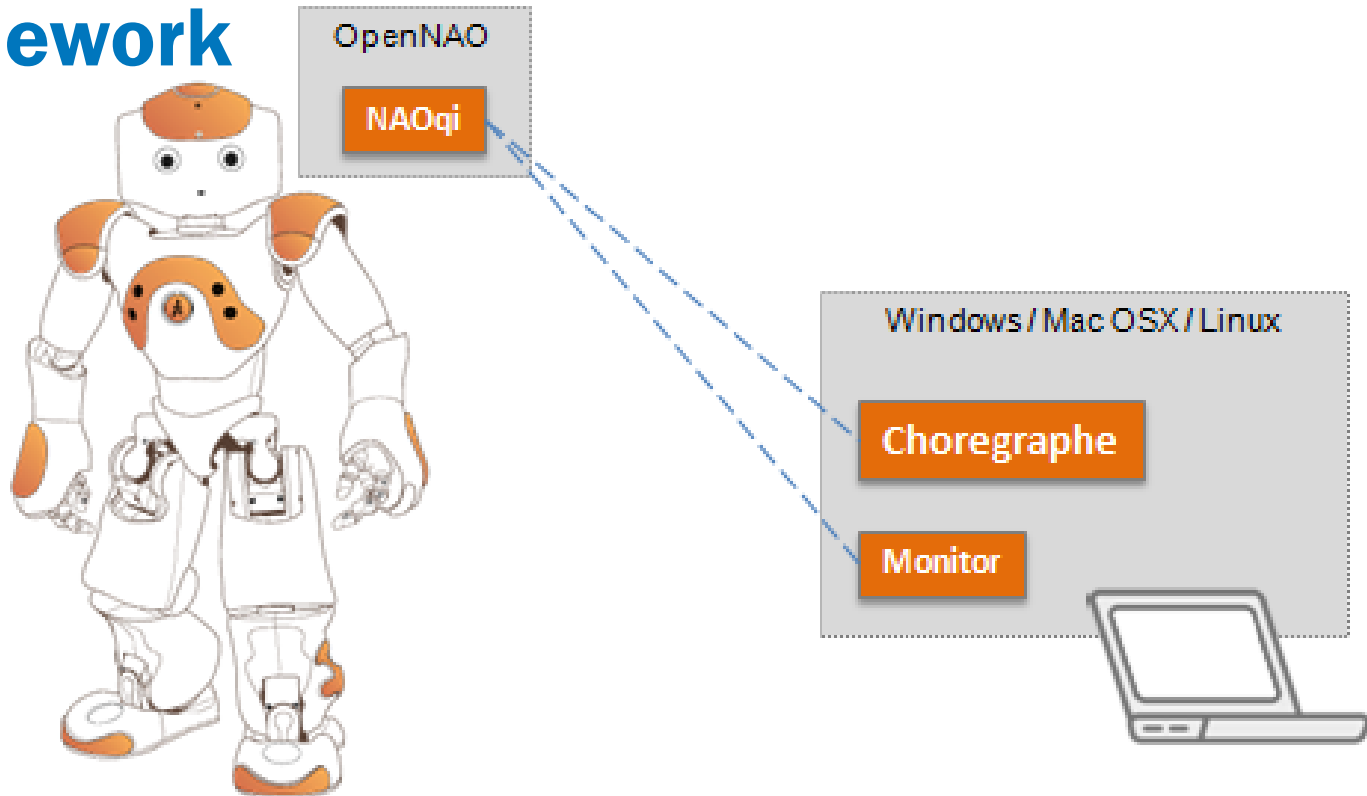


NAO Software – Operating System on NAO

- Name: OpenNao
- Background: It is adapted Gentoo Linux OS
- Update OS: NAOFlasher
- Connection: Connect to Nao over Putty



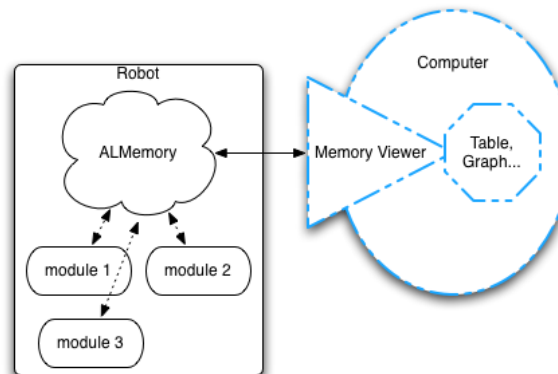
NAO Software – Framework



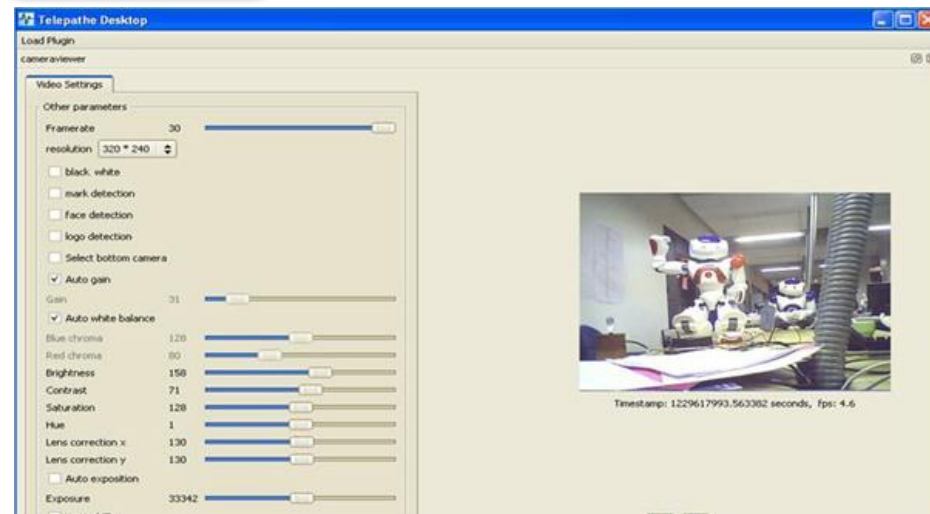
- Embedded Software, running on the motherboard located in the head of the robot, allowing autonomous behaviors.
- Desktop Software, running on your computer, allowing creation of new behaviors, and the remote control of the robot.

NAO Software – Monitor

- Memory Viewer

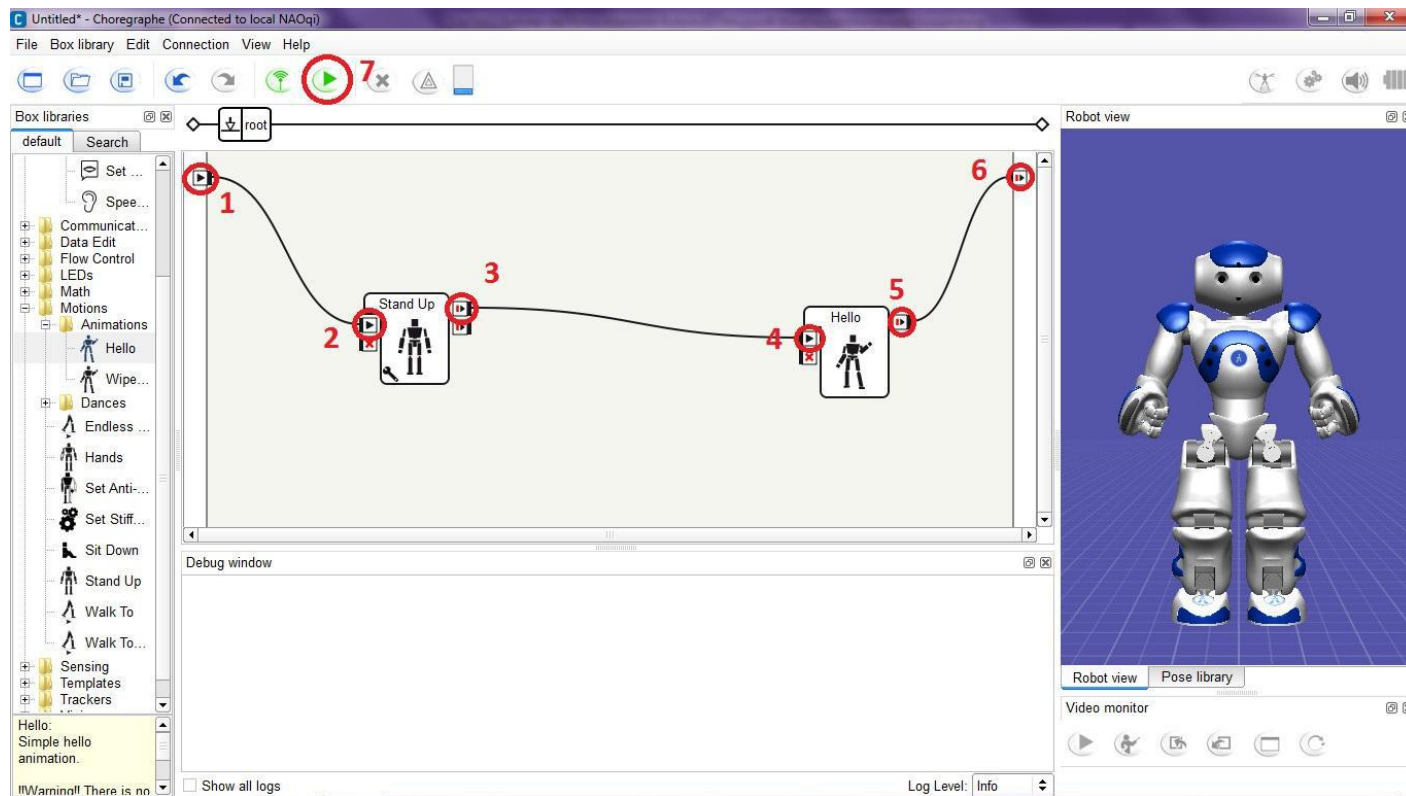


- Camera Viewer



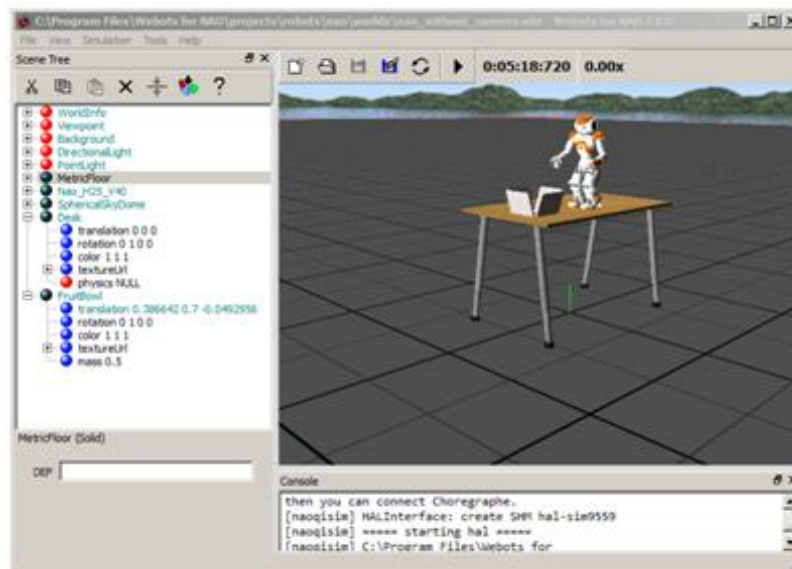


NAO Software – Choregraphe



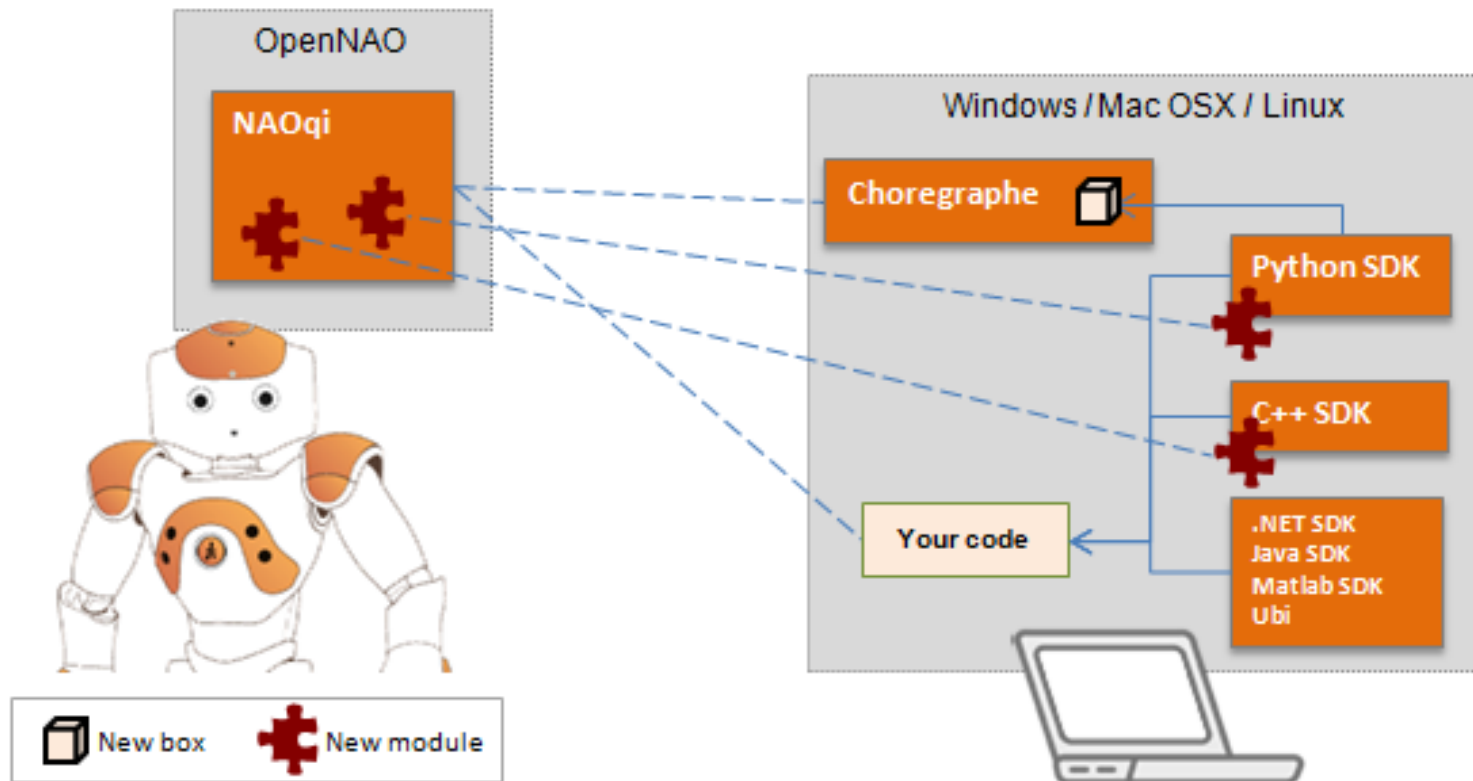


NAO Software – Webots for NAO



- In general to simulate robots interacting in their environment or für Rapid Prototyping
- Here only for Nao

NAO Software – SDK (Software Development Kit)





NAO Software – SDK (Software Development Kit)

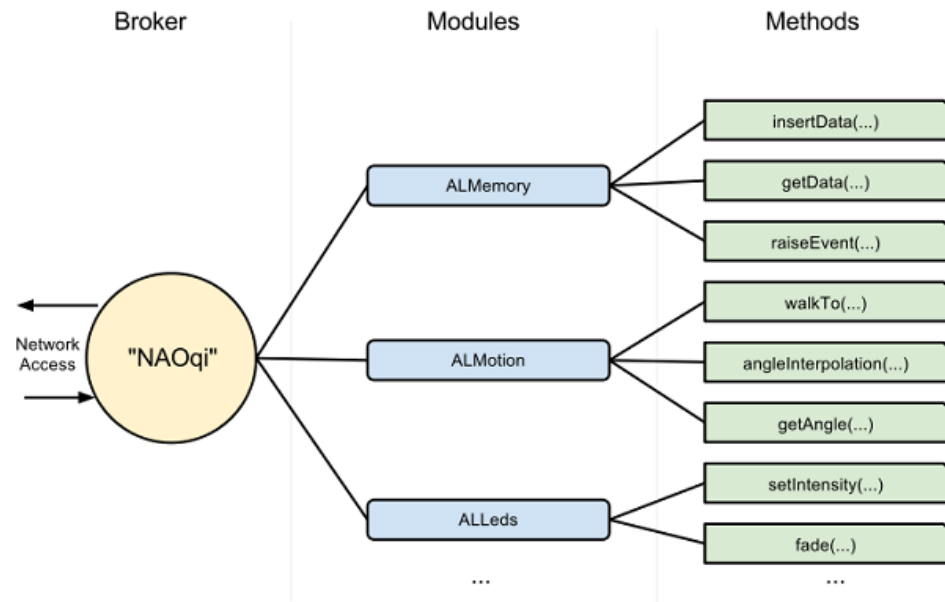
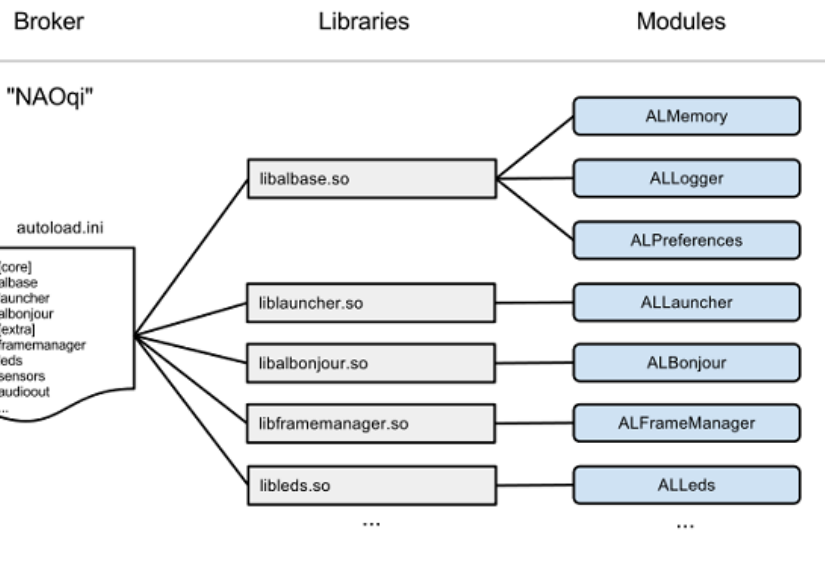
Supported programming languages:

- Fully supported:
 - C++ Main use to programm Services (Dienste) because it is fast
 - Python Main use to programm Tasks (Funktionen) because dynamic
- Partly supported:
 - .NET
 - C#
 - Visual Basic
 - F#
 - Java
 - Matlab
 - Urbi



NAO Software – Structure of NAOqi

- NAOqi is a Meta-Operating-System: Means it is platform independent
- After starting the NAOqi process it will initialize couples of C++ or Python Modules like ALMotion, ALSensors or ALMemory





NAO Software – NAOqi API (Application Programming Interface)

- **NAOqi Core** (like ALBehaviourManager)
- **NAOqi Motion** (like Stiffness controll of the joints)
- **NAOqi Audio** (like ALAudioRecorder)
- **NAOqi Vision** (like ALFaceDetection (human face) and ALVisionRecognition (objects))
- **NAOqi Sensors** (like ALSonar or ALBattery)
- **NAOqi Trackers** (like ALFaceTracker or ALRedBallTracker)
- **DCM** (Device Communication Manager) (like DCM/Time or DCM/RealtimePriority)
- **Types** (Type declaration according C++ to Python, Java & .Net (C#))

Source: [.../doc-release-1.14-public/aoqi/index.html](http://doc-release-1.14-public.aoqi/index.html)





NAO Software – Code Examples

- In general you can find some interesting example codes in C++ and Python under [.../doc-release-1.14-public/dev/example-index.html](#)
- It exist a python script for face recognission and tracking. See NAO Software documentation under [... /doc-release-1.14-public/dev/python/examples/vision/face_detection.html#python-example-vision-facedetection](#)
- In C++ you can controll his LED eyes under [.../doc-release-1.14-public/dev/cpp/examples/sensors/leds/leds.html#cpp-tutos-leds](#)
- Be aware that the more he has to calculate, the faster he gets headache !





NAO Software – SDK installation under MS Windows

1. Install „MS Visual Studio“ (Visual Studio 2010 oder 2008)
2. Install „CMake “ (CMake 2.8.3 or higher)
3. Install „Python “ (Python 2.7 or higher)
4. Add Cmake and Python folder into your Systemvariablebe Path if it is not done:
Startmenu rightclick auf Computer -> Eigenschaften -> Erweiterte Systemeinstellungen -> Erweitert -> Umgebungsvariablen -> Systemvariablen: Path
Doppelklick -> -Add at the end the two directories. Make sure that the directories are seperated with a semicolon.
Example: ...;C:\Program Files (x86)\CMake 2.8\bin;C:\python;C:\python\Scripts
5. Install „qiBuild“ (naoqi-sdk-1.14.5-win32-vs2010.zip) for C++ Crosscompiler





NAO Software – Status quo

- At first we have to be able to install all the SDK components properly
- Get connection to the virtual (simulator-sdk) or real NAO Robot
- Connect the Virtual Robot to Webot
- Evaluate the behaviour and optimize the NAO Robot (Debugging)



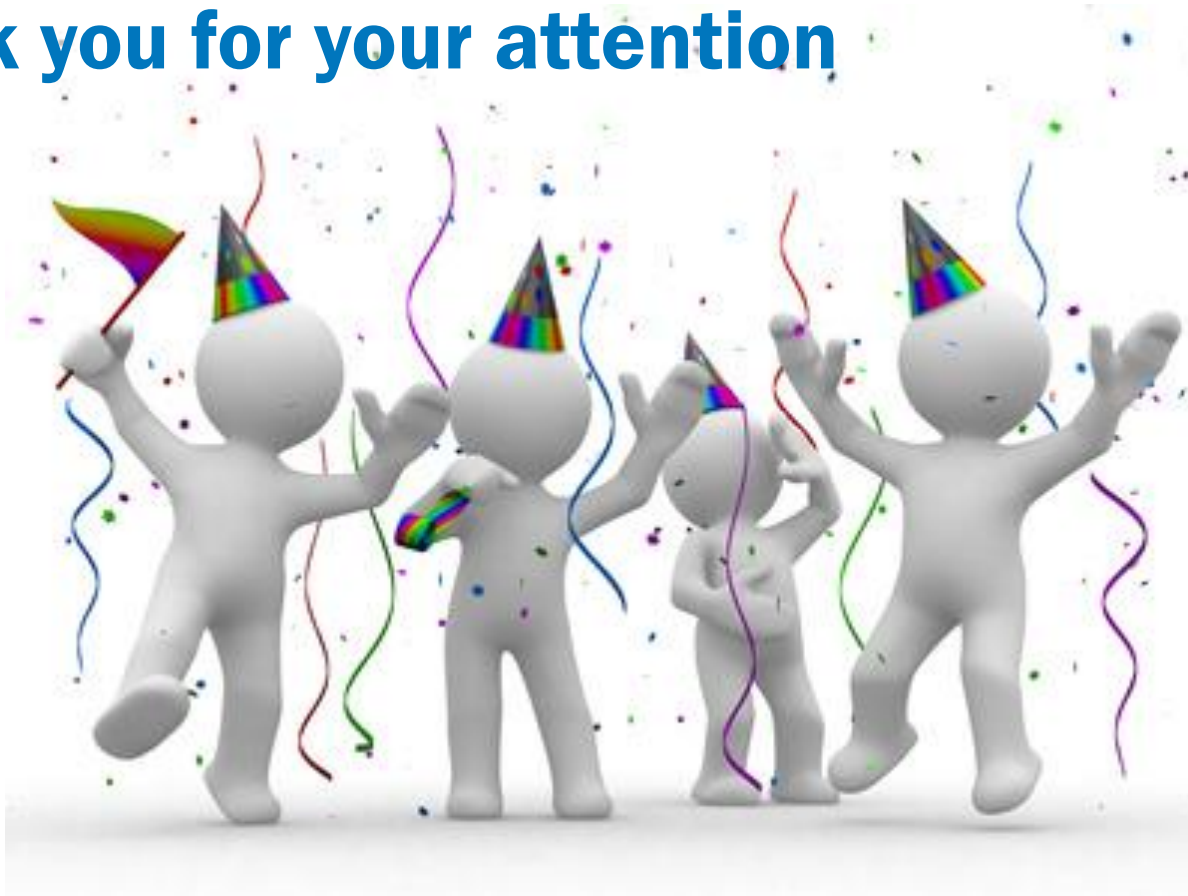


NAO – Are there any questions





NAO – Thank you for your attention





NAO Sources

- <https://www.google.de/>
- <http://translate.google.de/>
- <http://www.aldebaran-robotics.com/en/>
- NAO Software 1.14.5 documentation
- <http://de.wikipedia.org>
- <http://en.wikipedia.org>

