Best Practice in Robotics (BRICS)

Grant Agreement Number: 231940

01.03.2009 – 28.02.2013

Instrument: Collaborative Project (IP)

Specification of required hardware

Tim Guhl

Deliverable D1.1 – public version

Lead contractor for this deliverable:	KUKA Roboter GmbH
Due date of deliverable:	March 01, 2010
Actual submission date:	April 07, 2010
Dissemination level:	Public
Revision:	8

Contents

1. Introduction 3
2. Analysis of the EURON questionnaire 3
   2.1. The questionnaire 3
   2.2. Conclusions drawn from this questionnaire 11
3. Analysis of the Fast Research Interface Requirements 11
   3.1. The questionnaire 12
   3.2. Developing the FRI based on the findings of the questionnaire 17
4. Hardware requirements of the showcases 21
5. Functional requirements of the community 21
   5.1. Developing the third questionnaire 22
   5.2. The questionnaire 22
6. Conclusions 29

Executive summary

In the Description of Work (DoW) it was detailed that the needs of the community will be assessed during the first six months of the project (Task 1.1). During the next three months the consortium was to come up with a selection of hardware fulfilling a maximum of the requested requirements with the resources available (Task 1.2). It was detailed that this deliverable (D1.1) would contain the outcome of these tasks - it would specify hardware for Best Practise in Robotics addressing modularity and interoperability aspects.

The approach to Task 1.1 was to first evaluate a questionnaire previously conducted among the EURON community as described in Section 2. The outcome of this questionnaire, described in detail in Section 2.2, was that the community would see a real benefit in a common research platform as long as it is modular enough to allow individual research topics to be followed and the pricing is such that it would reduced costs overall. If the "kit" included all components required for mobile manipulation, this would allow the majority of research interests to be followed. The consortium could use these first results and their own experiences to make decisions regarding, for example, the showcases. Experiences made during the conduction and the evaluation of this first questionnaire will also be valuable for later work in this work package.

Having developed a broad overview of the needs of the community, the consortium then wanted to obtain in depth feedback for a specific piece of hardware. The KUKA lightweight robot (LWR) was chosen as a use case. A second questionnaire, described in Section 3, was used to find out more about the requirements concerning an interface to this robotic arm. The results of the questionnaire were analysed and some of the conclusions can be found in Sections 3.1.1 to 3.1.9. Then the Fast Research Interface (FRI) for the LWR was designed based on this knowledge and the experiences and preferences of KUKA as the supplier of the interface. The FRI is described in some detail in Section 3.2. The results will also be useful when developing interfaces for other hardware.

While the results of these two questionnaires were in line with the DoW, the consortium decided that it could serve the community better if a further questionnaire was conducted. Based on the results of the former two activities the consortium developed this third questionnaire to explore the community's needs with respect to functionality in more detail. Details are given in Section 5.

All available questionnaire responses were combined with the collective experience of the principal investigators. The hardware requirements of the showcases were also an important aspect to be considered (Section 4). The conclusions are that the community has a great need for common research platforms and that it is possible to provide platforms which will satisfy the needs of a broad spectrum of research interests. Most important are usability, modularity, extendibility, and price. These conclusions are detailed in Section 6.

1. Introduction

This deliverable describes the approach taken, progress made and results developed with respect to Tasks 1.1 and 1.2. In the introduction the two tasks will be introduced briefly and the overall approach will be described. The following sections will detail the steps taken towards the completion of the tasks and will highlight some of the shortcomings. Some conclusions are drawn in the final chapter.

Task 1.1 is concerned with determining the needs of the community. Several steps are described to be to ensure the involvement of the community:

- The results of a questionnaire conducted within the EURON community before the start of BRICS are to be analysed, mainly as input to the third questionnaire to be developed
- A second questionnaire targets the community's needs with respect to the hardware interfaces.
- A third questionnaire covers a broader range of the communities needs with respect to functionality.

Task 1.2 uses the results of Task 1.1 and, by considering the resources available to the consortium, determines how existing platforms and components can be put to best possible use for the community. The goal here is not to develop new hardware but to ensure that BRICS maximises the usefulness to the community of the provided systems, interfaces and models.

During the first few months of the project, KUKA conducted the second questionnaire within the BRICS consortium and during a workshop at the EURON Annual Meeting 2009. Here the Fast Research Interface (FRI) for the KUKA lightweight robot (LWR) was used as an example. This questionnaire and the results obtained are described in Section 3.

The definition of the showcases was also driven forward. It was important to determine the hardware needs of these early to ensure all necessary components would be available within BRICS. During the process the needs within the consortium were also considered. The results of this work can be found in Section 4.

Finally, the third questionnaire was developed to determine the needs of the community in more detail than already done. Very much to our regret the results of this effort are not yet available. The questionnaire has been designed and is described in Section 5. In the last section the conclusions are drawn.

2. Analysis of the EURON questionnaire

2.1. The questionnaire

The questions of the conducted questionnaire and the replies are detailed below.

2.1.1. Question 1: Would you be in favour of having one or several commercially available platforms which could be used as research platforms for future robotics research?

Statistically the answers to this question can be represented as shown in this graph. Many of the replies were more detailed and can be summarised by the following bullets:

- The cost needs to be within reason.
- The participants would enjoy the comfort of having working systems available.
- There are some concerns whether these systems would be flexible enough to allow for novel research. A modular approach is considered a must.
- The platform needs an open architecture and good documentation.

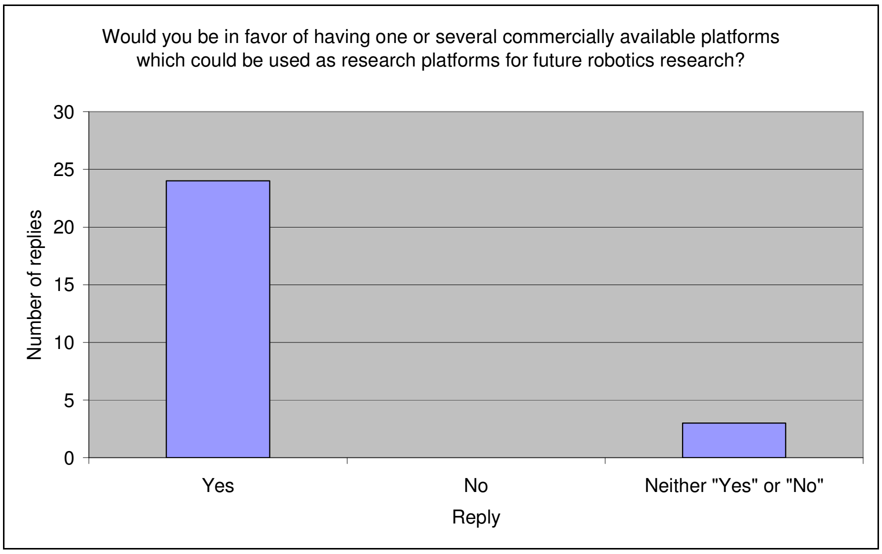

Figure 1: Replies to question 1

2.1.2. Question 2: For which research areas would you like to have a research platform and what would be an ideal configuration of such a platform?

A choice of several research areas was given as can be seen from the values stated in the graph below. Multiple answers were possible. The more detailed replies were analysed and the most important points made were:

- A major concern, besides the cost of the robotic system itself, are the costs due to the environment or the task (e.g. in medical robotics)
- Tasks / applications / research goals mentioned:
    - Force-controlled deburring
    - Robot assisted surgery
    - Platform with 2, 4 or 6 wheels / legs
    - Robot assistant in manufacturing
    - Mobile manipulation in human environments
    - Robot perception platform with a collection of algorithms for calibration, sensor fusion and 3D representations
    - Modular architecture that will allow the integration of further sensors
    - Differential and omni-directional drives for indoor and outdoor applications
    - Doubts that "universal" platforms can be designed for some of these purposes
    - Haptics
    - Manipulators with more than 6 degrees of freedom
    - Mobile platform for heavy payloads
    - Mid-sized humanoid platforms
    - Parallel robots (cooperating?)
    - Cognitive robotics
    - Flying robots
- I/O devices and sensors mentioned:
    - Vision (incl. omni-directional and 3D)
    - Sonar / Laser scanners

- Other Time-of-flight (ToF) sensors
- Sound (input and output)
- Touch screen.
- E-nose
- Inertial sensors, rate gyros, inclinometers
- GPS (mostly outdoor but also indoor)

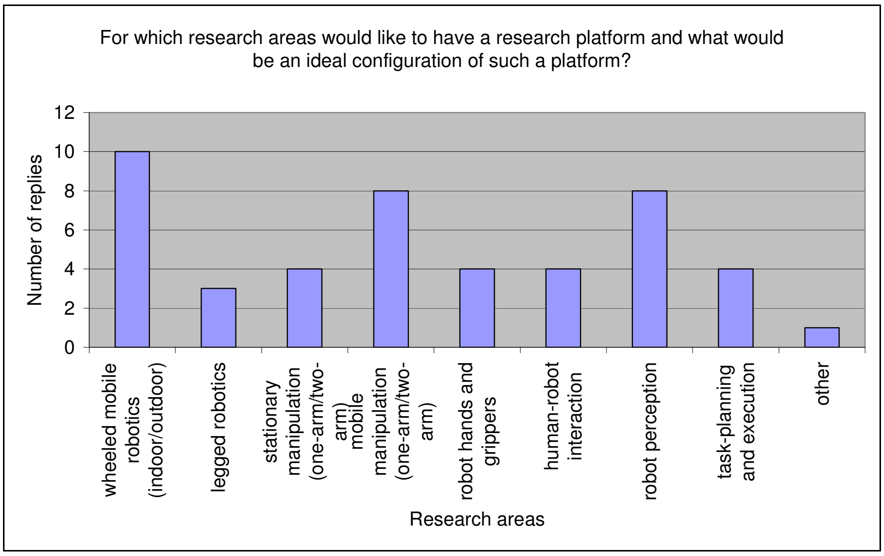

Figure 2: Replies to question 2

2.1.3. Question 3: In which potential application areas would you like to have a research platform

Here a set of predefined application areas was to be chosen from. The numerical evaluation can be viewed in the included graph. One of the more detailed replies mentioned “animats”. Two of those responding mentioned a concern that concentrating on the application area may distract from the real issues. Research efforts should ensure a generality, which makes the results reusable for many domains.

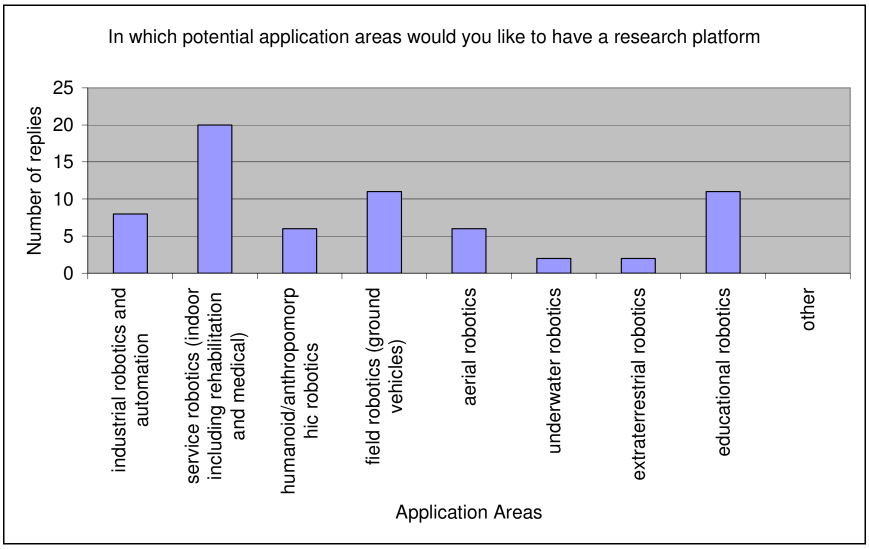

Figure 3: Replies to question 3

2.1.4. Question 4: Which requirements for a research platform does your research impose on the following aspects

Some of the respondents voiced some confusion regarding the exact meaning of this question. Most replies were in line with what was expected though. The numerical presentation of the results can be viewed below. The assumption is that the respondents selected the options if they had relevant special requirements. The more detailed replies can be summarised as follows:

- Locomotion (wheeled, legged, flying, swimming/floating)
  - Capability to move on unstructured / all terrain environment
  - Articulated wheeled
  - Long life batteries
  - Predominantly wheeled, some legged and flying
  - Omni-directional
  - Holonomic walking
  - Cheap wheeled platform for education
- Manipulation (one-/two-armed)
  - Torque controlled two-arm manipulation
  - Compliant joints with stiffness control
- Grasping (gripper vs. anthropomorphic hands)
  - Should be able to handle everyday objects
  - If flexible enough, grippers are preferred as easier controllable and more robust
- Sensor equipment
  - Please refer to question 2 for a list of types of sensors
  - Localisation, terrain reconstruction and obstacle avoidance capabilities
  - Drivers for hardware must be open source
  - All types of proprioceptors
- Operating system (OS)
  - Linux
  - Windows CE
  - similar to POSIX
  - Real-time
  - Open source if possible to avoid patent or copyright issues
  - The OS of the build host should not be restricted by the OS of the controller
- Programming and development environment
  - “Matlab for robotics”
  - JIT compiled languages preferred for security reasons
  - Low-level languages (Fortran, C, C++…) increase development time (but another user requested C++)
  - High-level languages (Prolog, Ruby, Matlab) are too domain specific
  - Ready to use modules / library
  - Open source
- Middleware and communication software

- TCP/IP
- ASEBA
- OROCOS

- Other
  - Scale should be compatible with human centred environments
  - Modular mechanical hardware
  - Consider white papers (EURON, CLAWAR…)
  - Simulation software

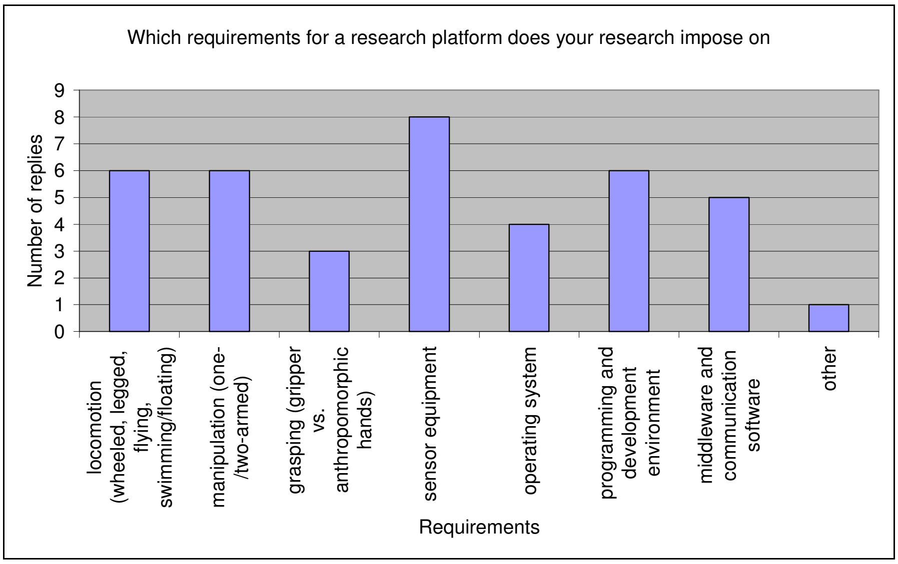

Figure 4: Replies to question 4

2.1.5. Question 5: Which other demands would you make on a research platform in terms of the following aspects

Again the numerical representation below indicates how many responses were received for each of the given aspects. In some cases, however, the respondents just ticked all or most boxes meaning that they require these aspects without stating their specific requirements. Those stated are summarised here:

- Access to low-level hardware and software components
  - Mentioned as most important several times
  - Needs to be open
  - Documentation and source code need to be available
  - Good APIs with direct access into logically coherent operations will render low-level access unnecessary
  - Should allow for torque control
- Availability of basic capabilities for navigation and manipulation (via APIs and software libraries)
  - Mentioned as second most important several times
  - Odometry, vision, and collision avoidance
  - Implement by supporting e.g. Player/Stage?
- Architectural concepts and structures
  - Mentioned as important but not critical

  - Distributed system components
  - Wireless communication
- Task level programming
  - Mentioned as useful but not critical
  - Up to the programmer, hard to find a standard here
- Autonomy
  - Independent of external power source for 2-4 hours
  - Independent of external computing power?
- Modularity
  - It is most important to develop modules not robots
  - Modularity in mechanics, hardware and software
  - Define standards
  - Look at CLAWAR Network's and ISO Standardisation Committee's work
  - Also with respect to sensor integration
  - Modules to be usable for both, wheeled and legged robots
- Configurability
  - Also with respect to sensor integration
  - Important w.r.t. software
- Extensibility
  - Need to be able to integrate own work
  - USB Ports
  - Digital and analogue I/Os
  - Robust D/C power supply with several voltages
  - Fixtures allowing to add hardware
- Simulation
  - Important
  - Matlab like environment
  - Open to extensions from community
  - Implement through e.g. Player/Stage or Webots (Cyberobotics)
  - High-fidelity, realistic, 3D simulation
  - Enable simulation of sensor and motor properties
  - Lack of open source or free software
- Power supply
  - Batteries are mentioned
  - Capacity/size ratio
  - Battery management systems should be incorporated to reduce costs due to wasted batteries
- Other
- Comments on all aspects
  - Several respondents mentioned that all aspects should be open source

- Open source with standardised APIs
- Open source approach would allow libraries to develop
- Has to be economical (absolute and as a cost-benefit ratio)
- The risk of 'basic capabilities' libraries is that they have a tendency to become very high-level, preventing the development of alternative approaches (as one using a specific algorithm is already available)

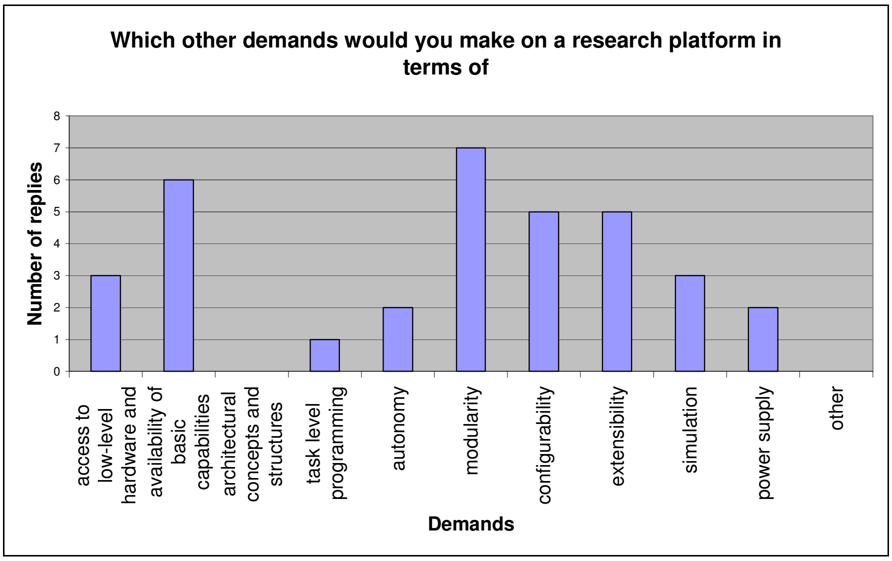

Figure 5: Replies to question 5

2.1.6. Question 6: On which level would you want to access the platform

The pretty much unanimous opinion here was that access is required at all three levels, depending on the research interests of the user. The platform should provide levels of abstraction allowing users to ignore levels that are not important to them (e.g. during teaching), but the possibility to modify low levels whenever necessary.

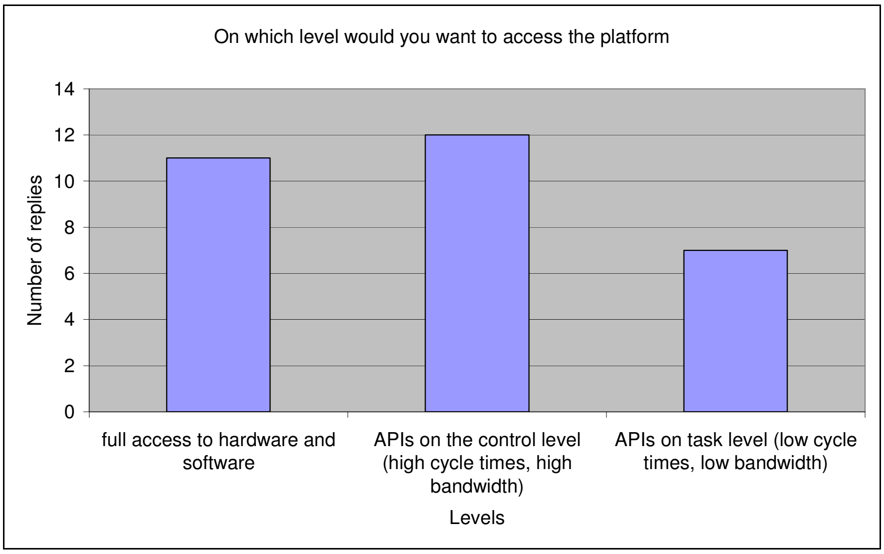

Figure 6: Replies to question 6

2.1.7. Question 7: Which budget do you have available for a platform as specified above

As the users had not always concentrated on a specific platform above, they found it hard to reply to this question. The following detailed replies were made:

- One respondent mentioned that one of the most important factors for him when deciding how much to pay would be the quality of the documentation and the source code.
- Some indicated they would not trust a product for less than €15k to be very useful.
- Researchers working on swarm robotics may think in terms of one price for a set of X robots.
- For teaching platforms a price tag of €1000/robot was mentioned to allow the lecturer to purchase enough robots for the whole class.
- Another important factor is how many researchers can access the platform simultaneously.

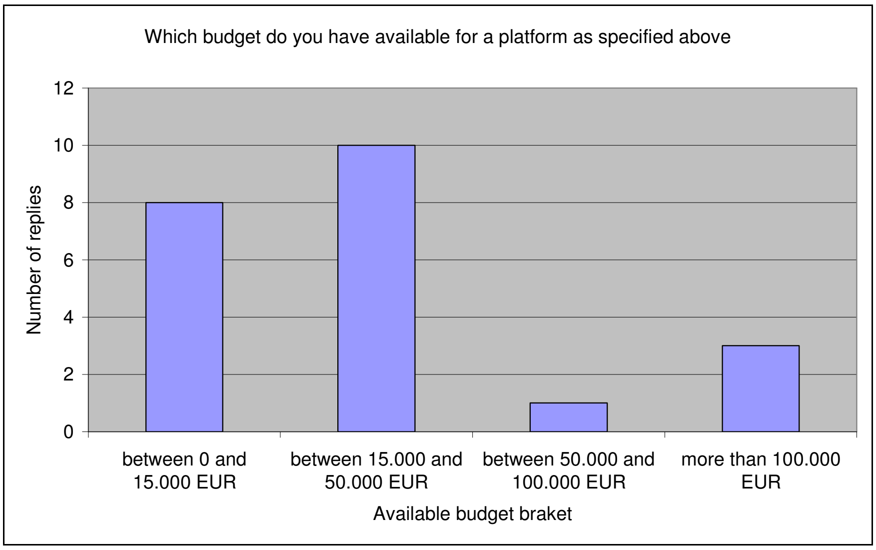

Figure 7: Replies to question 7

2.1.8. Question 8: If the cost for the platform, which suits your needs best, would exceed your budget by far, which of the scenarios below would you prefer to still get access to and work on a high-tech robotics platform

This question is not relevant to this deliverable, but to the last task of this work package (Access models). However, for completeness a short review of the results are included here.

- No agreement was found between those that stated an order of preference. Marginally the most common answer was “Option 3, Option 1, Option 2 in decreasing order of preference”.
- A leased platform should cost around €1000/month
- The second option is seen as a good option if several institutes at the same location work on the same project, but not ideal if they are located physically far apart.

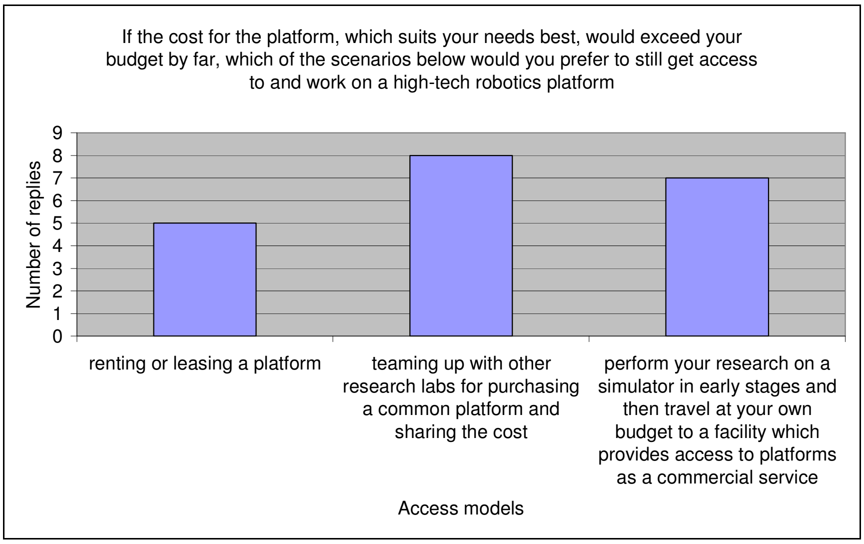

Figure 8: Replies to question 8

2.2. Conclusions drawn from this questionnaire

The main concern of this questionnaire was to determine the need for a common research platform and to determine how the usefulness of such a platform could be maximised. It can be concluded that such platforms are desperately needed, but some conditions need to be met for them to be embraced.

The main advantage is seen to have a system available to reduce set up costs and time requirements. However, this is only the case if the system allows the researcher to adjust it to his or her needs. Furthermore, the cost of the system and the necessary alterations must be low enough to justify the expense. While no clear picture of the appropriate cost emerged due to the wide spread of requirements, there was some indication, that systems costing more than €50k would be to expensive for some research institutions. It was clear that leasing or joint ownership of such platforms may be a good way of spreading the load over more shoulders. The researchers hope that good documentation and a developing community would allow them to make use of the system quickly.

If all modules required for wheeled mobile manipulation were available, then a large proportion of the suggested applications and research areas could be met. Ideally the system would be modular for two reasons. Firstly, this would reduce the costs for researchers only interested in a certain aspect such as mobility. Secondly, if a modular approach is used it can be considered a given that research teams may configure the systems to meet their individual needs (e.g., special configuration of sensors or grippers). The requirements, for example, with regards to configuration or physical properties of the robot, were detailed and varied.

While researchers may be more concerned with fundamental issues, these are often explored in the context of an application. A significant majority of the respondents were interested in ground based service robotic tasks. Other significant areas were industrial and educational robotics. Depending on the focus of their research the level at which the teams wanted to be able to control the robots varied significantly. A modular hardware approach would best be complimented by an interface allowing higher level access to make programming the system easy, but also making low level control possible if this is the area of interest. Some researchers seem to be asking for full control just because they do not trust the supplied controllers or because they think fewer resources have to be committed to implementing or applying their own controller than to learn how to use a controller supplied with the system.

One last conclusion from this questionnaire was that a slightly different approach had to be developed for future questionnaires to either give the respondents more flexibility for their answers (and to thereby obtain more detail) or to focus on comparable answers and replies.

3. Analysis of the Fast Research Interface Requirements

The first questionnaire conducted in this work package concerned the interfaces to robotic components to be developed in BRICS. The questionnaire concentrated on mobile manipulators in generally and the KUKA lightweight robot (LWR) specifically. This information had to be collected early as it would serve as input for other devices as well and the manipulator had to become usable by the consortium early to allow for other work to be conducted. It was also essential for the development of the use cases. In this section the questionnaire and the collected replies are described and analysed.

The following fundamental questions were to be answered by the questionnaire:

- What use case are you thinking of?
- What kind of interface is required, in your eyes, for this use case?
- Which information do you want to exchange and at what rate?
- How will the user try to access the interface? What kind of protocols would the user like?
- Which operating modes should the interface support?

The answers to these questions will result in questions the hardware supplier needs to ask himself:

- (How) will it be possible to write an interface which will satisfy the requests above that will work with his basic system?
- How should the interface deal with requests that could break the robot or could result in dangerous situations (e.g. communication failure).

3.1. The questionnaire

These and further considerations were kept in mind when designing the questionnaire. The headings of the following subsections are the questions that were asked and the answers given are summarised in each subsection. The conclusions are partially contained within these summaries. As a final result the interface developed in part based on the findings of this questionnaire is presented in Section 3.2.

3.1.1. Question 1: What is the use case of your intended application in the context of an open interface?

This question was asked to find out what kind of work and research people wanted to do with the robot and to understand better why they are requesting certain functionalities. The following use cases were named.

- Picking up things in an industrial setting using a novel hand design
- Pick and place tasks in a home or office setting including simple manipulation
- “Peg in hole” tasks
    - Challenging peg-hole clearances
    - Insert a key in a key hole and take it out again
- Advanced assembly and manufacturing tasks where the robot is adapted in real time based on sensor / process model information
- Visual servoing tasks
- Use robot to carry an instrument where the redundancy is controlled and using 10Hz mechanical bandwidth and a sensor communicating at 1kHz.
- Calligraphy
- Vibration damping (→ the robot already compensates for vibrations)
- Research in control algorithms
- 10 Hz multi-TCP „force“ control (or other sensor)

It can be seen that the robot will be used for a variety of diverse use cases. As these use cases are to different to be unified, the interface can not be unified either. However, the use cases could be divided into two groups. The first group consisted of various applications which could be solved using the standard KRC lr controller if this was to be equipped with an additional fast interface. This solution was realisable relatively fast and was therefore implemented resulting in the Fast Research Interface (FRI).

The second group of applications are based, to some extend on mobility. This requires a small controller, it means the robot and controller have to run on battery power, the controller must operate without the control panel (KCP), and several safety issues need to be overcome. This problem can only be solved in the mid or long term.

3.1.2. Question 2: Which information should be exchanged on a cyclical / noncyclical basis?

The following lists all replies given as an answer to this question:

- The robot should report is pose as well as the velocity and force vectors, all in Joint and Cartesian space.
- It should be possible to set the TCP in Joint and Cartesian space.
- It should be possible to find out the planned trajectory.
- The gripper position and sensor data.
- It should be possible to find out about interrupt events and where in the program we were.
- Low level interfacing is required so that the robotics hand and arm can used as one cinematic chain through an impedance controller.

3.1.3. Question 3: What cyclical frequencies are necessary? How much data would you like to transfer?

This question received a variety of answers:

- The amount of date to be transferred would be relatively little for the robot (about 1KB per cycle) but much larger for the sensors (>1MB).
- Outputs should be delivered as fast as possible (1kHz?)
- It should be possible to command the robot, depending on the application, at a mechanical rate of 10 to 30 Hz.
- If the robot can be considered stiff, then 250Hz will be sufficient.
- If it is necessary to compensate for vibrations or bending, 1 to 5kHz are required.

The wish list of Questions 2 and 3 was combined with the list of data that would have to be transferred to use all features of the robot. It was concluded that a frequency of 1kHz would (more than) satisfy most use cases. This resulted in the following list of inputs and outputs:

The interface should provide the following outputs:

- Measurements (at 1 kHz)
    - Joint position
    - Cartesian position of TCP
    - External torque
    - Estimated external force at TCP
- Trajectory generation (currently valid commanded values)
    - Desired joint position
    - Desired cartesian position
- A set of configurable variables from KRL (KUKK Robot Language)
- Current control mode

The input should also allow for the following inputs:

- Position control
    - Joint position
    - Joint speed
    - Cartesian position
    - Cartesian speed
    - Nullspace configuration

- Compliance issues: To use all features of the lightweight robots one has to consider three factors:
  - Stiffness
  - Desired commanded position
  - Forces or torques that can be superposed
  - Wave variables

3.1.4. Question 4: What kind of medium would you prefer (select from the list below)?

a) (TCPIP/UDP) Socket connecting an external computer. What (RT)OS will you use within the context of this interface?

b) Fieldbus, e.g. CAN, EtherCAT, ProfiNet connecting an external computer

c) Shared Memory inside VxWorks (Do you have experience with such a setup?)

d) Other

The following replies were given to this question:

- a) using Windows, Linux or MacOS.
- a) with RT Linux
- a) UDP with Linux
- a) and b) using EtherCAT and QNet.
- b) Ethernet
- b) CAN or EtherCAT
- c) Some mention experience with VxWorks, QMX and RTAI
- d) USB as compatible with “normal” PC networks
- d) low level digital I/O

3.1.5. Question 5: Concerning Question 2, what additional information should be provided, to use the sketched interface?

The following was mentioned in the context of this question:

- Good documentation
- Implementation examples with explanations → tutorials?
- To increase robustness each degree of freedom should have an additional line
- Configuration information such as the number of connections allowed
- Information about the internal state the robot is in (e.g., warnings, failure modes, saturation of drives…)
- Information about real time performance of VxWorks threat (e.g., jitter latency, memory usages, stack overflow…)

3.1.6. Question 6: Concerning Question 3, how should the Robot Controller react to irregular communication or lost packages

Here the questionnaire asked to consider a situation where, for example, the cyclic time had been set to 1msec, but the incoming telegrams were jittering between 0.5 and 4msec. Those interviewed offered the following replies:

- Provide a log of what happend
- Go to low impedance mode
- Generate error Message + put control to a “safety-controller“

- Interpolate
- It should be configurable how the robot reacts:
    - Robot stops and stays at the last position.
    - Robot continues until next packet arrives

It can be seen that the replies here were not very clear at times. Does the first reply mean "go to a save mode" similar to the second reply listed here? If yes, it is not clear what such a safe mode should be. We suspect the person asking to "interpolate" meant the system should "extrapolate". But even if that is correct it is not clear on what data one should extrapolate. This is not clear what the robot should do if it "continues" and how long it should keep on doing that.

3.1.7. Question 7: Is safety an issue to you?

With this question we tried to find out how concerned the user was with regards to "machine safety" (i.e. keeping the material safe from harm) and "man-machine safety" (i.e. preventing the machine from harming people). What kind of measures implemented directly in the controller would the user be willing to accept in order to ensure one or both of these aspects of safety? Those that completed the questionnaire replied as follows:

- The robot should adhere to the 80W limit stated in ISO 10218.
- Impedance should be controllable
- It should be possible to adjust or limit the power consumption and to set limits which would activate a safety controller.
- It should be impossible to send commands which would damage the robot via the interface (excluding external collisions of course).
- Humans should not be in danger.
    - The robot should stop if high external forces occur.
    - It should be possible to set velocity limits.

Again, these replies are not precise enough to make decisions without further knowledge (but of course it helped us to make the "right" choices). These choices concerned, for example, which frequencies should be allowed and whether (and how) to monitor the input signals to prevent mechanical harm as far as possible. A question may be what happens if the user changes from low to high impedance at a moment when the robot is quite far from its desired position? Mathematically the controller would command infinite forces, but this needs to be prevented. What kind of strategy should the controller use to reach the desired situation as quickly as possible? Can the reaction be appropriate without additional domain information?

Considering machine safety the following issues apply to the different control schemes:

- Position control
    - What do you do if the user commands include "jitter"?
    - You could perform some kind of filtering, but this would influence performance.
    - You could also extra- or interpolate, but based on what rule?
- Speed control
    - Jitter is not an issue here, but how do you ensure safety during transitions?
    - But, how can the robot controller perform context checks?
- Torque control
    - The robot can move very fast, if inappropriate torque is commanded.
    - The user would have to set up their own torque controller (including damping etc.)

3.1.8. Question 8: How is your intended interface embedded within KRL (KUKA Robot Language)?

KUKA is assuming that the users will benefit from using the KUKA look & feel if they are planning to commercialise any of their results, especially if existing KUKA customers were concerned. The users were asked what alternatives they see to using KRL. What would they use to describe their tasks and what kind of tool chain would they use? The following tool chains were mentioned:

- 20Sim, 4C and dCSP/CTC++
- If our tools could generate KRL code then KRL would be fine with us.
- OroCos
- Please supply an API controllable using any standard programming language such as C/C++
- We would like to implement our own control algorithms

3.1.9. Question 9 and 10: What parts of the KUKA controller you'd like to reuse / replace w.r.t Question 1?

The respondents generally replied that they did not know enough about the controller to give adequate replies to this question. One important aspect was that they would like to have a small (build size) controller which could be used on mobile robots (→ power supply issues). The only functionality they would definitely want to use is the safety controller.

3.2. Developing the FRI based on the findings of the questionnaire

Based on the conclusions drawn above and the feedback from the questionnaire the FRI was designed. The objective was to build an interface that enables researchers to perform their research and release them from "reinventing the wheel" (i.e., developing controller features that already exist), e.g.:

- Build a completely new robot system
- Check sanity of the robot system (motors, gears, power supply)
- Fieldbus infrastructure (Digital/Analog IO)
- Move the robot to predefined positions

KUKA came up with a short-term realisation of a „Fast Research Interface" (FRI) based on the KUKA Robot Controller KRC2 lr

- Open to the research community
- Subject to radical change (no backward-compatibility)
- No „10 year" maintenance of the very same behaviour of the interface
- Interplay with KRC/KRL

The FRI has the following features:

- Cyclic timeframe: range 1 - 100 ms
- Several control modes of lightweight robot can be utilized
    - Joint specific position control
    - Joint specific impedance control
    - Cartesian impedance control
- Interplay to KRL Programs (scripting features)
    - Superposition of LIN/PTP/CIRC Motions
    - Standard robot programming and operation
    - Touchup - Teaching

- Selection of control mode
- “TrigByContact” for mating tasks
- “DesiredForce”
- “VirtualBox”
- Safety system and switches (Test T1/T2, Automatic)
- Fieldbus infrastructure

Once the interface had been finalised it was tried out by KUL and KUKA. A demonstrator was prepared for the LWR Info Day in September 2009 which took place in Augsburg and Wessling. Here two LWR were integrated with each other. One LWR served as the haptic input device for the other. The control of each robot was handled by the KUKA controllers and the overall system was managed by the KUL system which interfaced to the KUKA controllers via FRI.

3.2.1. The FRI interface

A brief overview of the interface is given here. Full technical details are available in the documentation supplied with the interface. The users are able to connect using the following means:

- Simple UDP socket communication, simple binary communication.
- C++ sample code (“SDK”) included to facilitate porting to several different operating systems.
- Operating systems that have already been used in early tests:
    - QNX
    - VxWorks
    - Windows
    - Linux

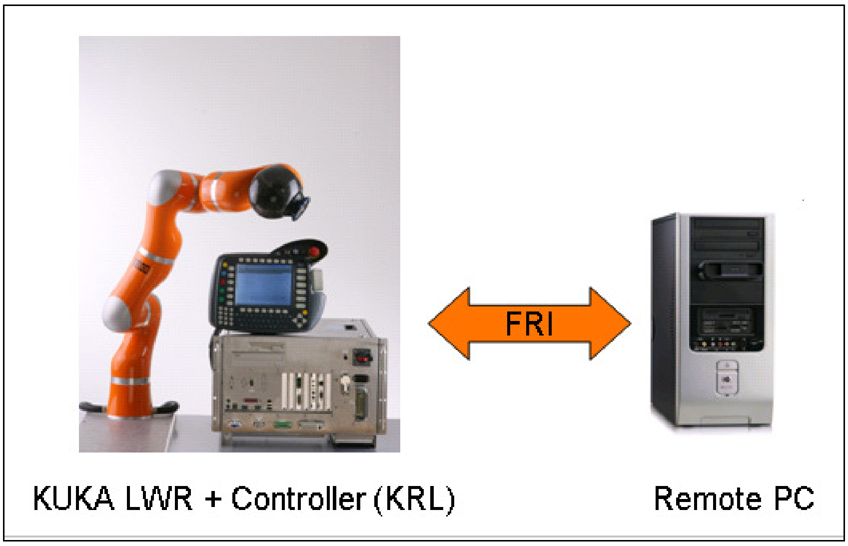

Figure 9: concept of FRI from a user perspective

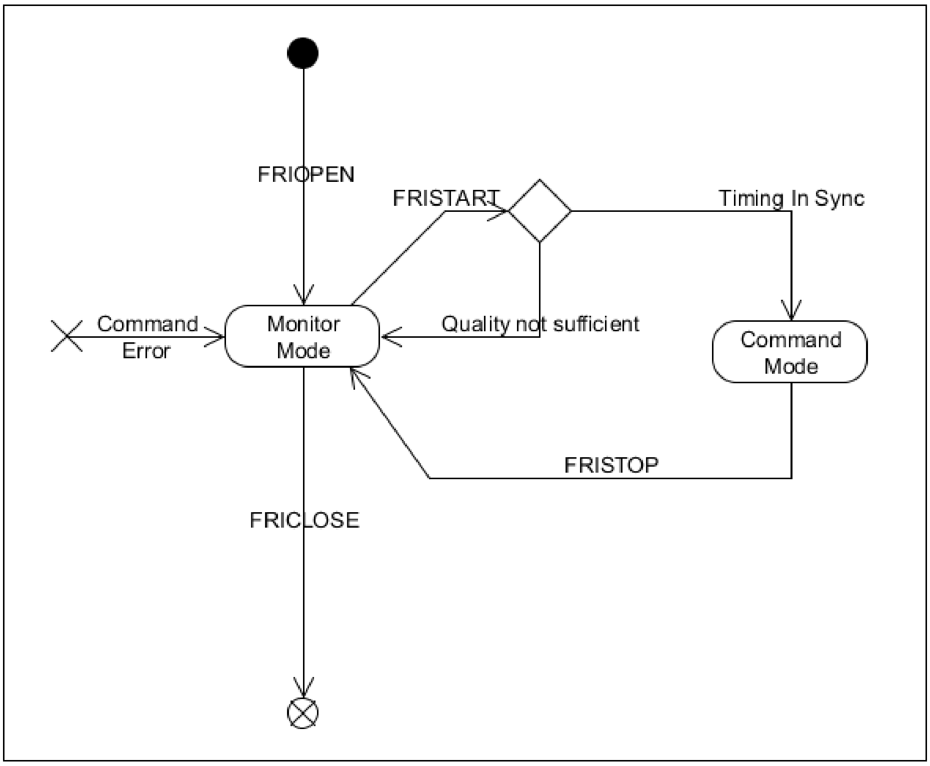

Figure 10: The FRI is based on a state machine allowing the user to connect to the interface and send commands. The connection can be terminated if its quality does not meet the required standards.

The user has to establish the connection using one of the means detailed above. He can then issue the following commands which will cause state changes as also illustrated in Figure 10:

- Command „friOpen“
  - FRI is opened
  - UDP socket is opened and communication to remote computer started
  - (Fixed) timing is specified by user; any timing between 1 and 100ms is allowed
  - Timing is validated by the controller (vital for further modes of operation)
  - Data exchange with KRL interpreter commences
  - Change of LWR-specific control modes possible
- Command „friStart“
  - Commands from remote computer are enabled
  - Several control modes are available
    - Position control
    - Cartesian stiffness control
    - Joint stiffness control
  - Levels of „autonomy“
    - Superposition of KRL based trajectory
    - Autonomous trajectory generation
  - Safety limits are specified
    - Safety check limits (speed, acceleration etc.) can be shrunk by user
    - Vital for developing software

- User definable for the interplay with KRL: 16 Real values, 16 Integer values, 16 Booleans
- The KRC controller communicates the following information to the remote computer
    - Administrative data
        - Control strategy
        - Timestamp
        - State of the drives (on/off)
        - User defined variables
        - „House Keeping Data“, like temperatures etc.
    - Robot data
        - Measured position
        - Commanded position (interpolated from controller)
        - Measured torque
        - Estimated external torque
        - Jacobian
        - Massmatrix
        - Gravity
- The remote computer is required to send the following information to the KRC. The exact nature of the variables and which ones are required depends on the chosen control mode (position, joint, or impedance).
    - Desired position
    - Desired velocity
    - Superposed torque
    - Stiffness spring constant
    - Stiffness damping factor
- The underlying control laws are given in the documentation, allowing the user to change the behaviour of the robot by changing parameters.
- Safety concerns were also considered during the design, but the user is responsible to ensure adequate separation between humans and machine to prevent injury.
    - All operating modes (Test – T1/T2) (Automatic) are supported
    - Strict observation of joint speeds
    - Strict observation of specified timing (sample rates)
    - losing too many packages will eventually stop the robot
    - not adhering to the specified timing will eventually stop the robot
    - User can shrink safety limits for the sake of software development

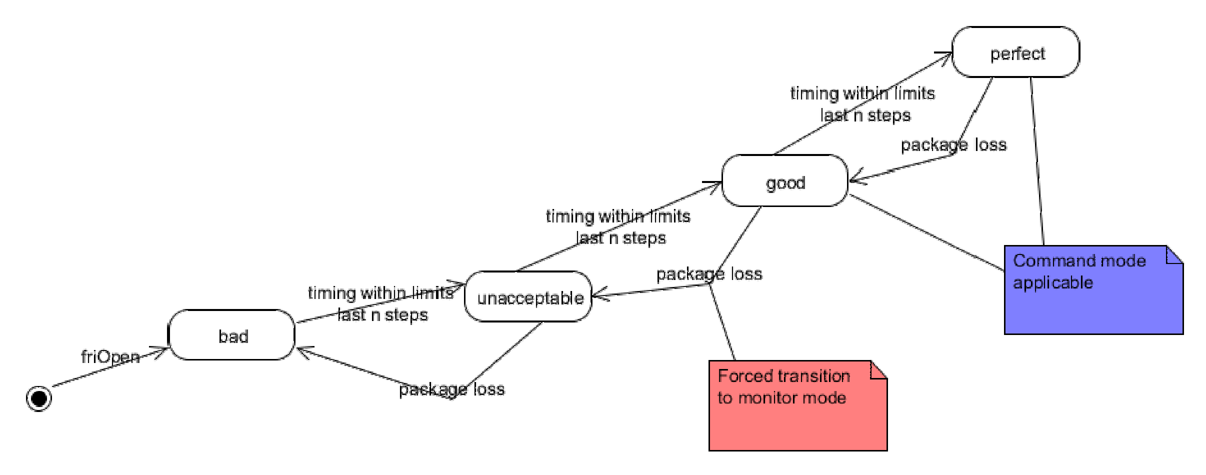

Figure 11: The remote computer has to stick to rules to keep the connection open.

Concluding it can be said, that the FRI was developed based on the requirements of the users which were assessed based on the questionnaires described in Sections 2 and 3. The FRI allows the users to start using the KUKA lightweight robot by following a few easy steps:

1. Port the KUKA sample application to the target operation system & compiler
2. Understand the mechanisms (by playing with the running sample application)
3. Integrate the FRI into the research software system
4. Done

4. Hardware requirements of the showcases

Another important source of information on the requirements regarding hardware were the showcases. They are described in detail in Deliverable D5.1. An overview is given in Section 2.2 and detailed descriptions for each showcase are given in Section 3 of D5.1. The following systems are currently in use within the consortium and will provide input here:

- BlueBotics Platform
- IPA Care-O-Bot 3
- IPA rob@work 2
- KUKA omniRob
- KUKA Lightweight Robot
- BRSU Johnny

The requirements of the community can be derived from questions 4 and 5 (Sections 2.1.4 and 2.1.5) of the first questionnaire. Further information will be collected using the third questionnaire. Based on the findings of this questionnaire and the detailed analysis of the requirements of the showcases, the hardware required will be determined. It is already clear that one important aspect to be considered is how to connect non-holonomic mobile platforms to holonomic arms and head devices and how to manage the control of such a system. Work on this topic is ongoing. All findings of the questionnaires and this investigation will be presented in an update of this deliverable.

5. Functional requirements of the community

As detailed in the executive summary, adequate results had been obtained by the previous two questionnaires. Additional information has the potential to greatly improve the quality of the work done in BRICS. Hence a third questionnaire was designed. However, mostly due to resource shortages during the financial crisis, the questionnaire has not been conducted yet. Therefore this section will present just the questionnaire but not the replies of participants or the analysis of these answers.

The only negative effect of this was that Task 1.2 has been delayed. Additional resources are already being committed to design the questionnaire and it was not possible or justifiable to carry out the full analysis twice. As preliminary results were available and sufficient for the other tasks to continue as planned, this did not lead to problems for the project. A revision of this deliverable will be made available once the questionnaire has been carried out and the results have been analysed.

5.1. Developing the third questionnaire

The questionnaire consists of several sections. In the first section, the general information is collected with a focus on the current situation of the researcher. The contact details of the respondent are requested. While this information is optional, it is hoped that the respondents will complete it to allow for clarifying questions at a later date should this be necessary. Information about currently available hardware and software as well as their current view of what they know of and find desirable.

Next, information about the use case is collected so that the replies take be evaluated in the right context. If the respondent is thinking of several use cases, he is asked to complete several copies of the questionnaire. This section is collecting more focused and detailed information than the first questionnaire. It uses categories developed and published in the EUROP Strategic Research Agenda.

The final section asks detailed questions about what kind of access models the respondent could imagine and the platform itself respectively. As such it is a contribution to Tasks 1.5 and 1.6. This part of the questionnaire was developed based on the experiences made with previous questionnaires and the access models developed within the consortium. It also asks questions about the procedures at the institutions and how to best organise benchmarks which may be one of the benefits of common research platforms.

5.2. The questionnaire

The prime objective of BRICS is to structure and formalise the robot development process itself and to provide tools, models, and functional libraries, which help accelerating this process significantly. With this questionnaire the BRICS consortium (www.best-of-robotics.org) is trying to find out how achieve this in a way to best serve you, the robotics researchers. We would like to know more about yourself, your work and the settings in which you try out your results in the real world. This way we hope to develop the above named tools, models and libraries for systems that will be useful to you. The more details you give below, the better we will be able to do this.

5.2.1. About you

5.2.1.1. Optional section

This first section is optional, but will allow us to contact you should we require further details. Please do not complete this section if it may influence your replies below.

- Name:
- Position / Role:
- Institution:
- Email:
- Phone:

5.2.1.2. Compulsory section

Please answer the following questions:

- What kinds of platforms are already available to you (including your own developments)? How much effort when into their development (approximate man months needed to set up the system with regards to selecting and implementing an architecture, integration of standard algorithms…)

- What kinds of components are already available to you (including your own developments)? How much effort when into their development (approximate man months)
- Which platforms / components are you already aware of which you would like to use? Please list the most attractive ones first (e.g. IPA Care-O-bot…)
- Which platforms / components are you already aware of which you are NOT so keen to use? Why are these not (so) attractive to you?
- Which operating system(s) do you use for your robots?
- Which programming and development environment do you use?
- What does a Model Driven Engineering Tool Chain need to be useful to you (e.g. simulation environment)?
- Do you use PlayerStage, Corba, MSRoboticsStudio, ROS or similar robotics related packages?

5.2.2. Use case

For two reasons we would like you to focus on a specific use case (e.g., "window cleaning robot") when completing this questionnaire. Firstly, this will help us to understand your answers better as the context is known. Secondly, this focus will most likely help you when answering the questions below. Please feel free to complete the questionnaire more than once to cover different use cases.

5.2.2.1. What is your professional position in the supply chain?

Please select one of the following:

- hardware manufacturer (Please state what kind of hardware you make: e.g. complete robotic system, a specific component (which one?), a sensor, …)
- system integrator (providing integrated systems, both HW & SW)
- research in robotics (including academic research, research institutes, industrial research…)
- education – please indicate whether you:
    - teach a subject related to robotics or
    - use robotics as a teaching aid
- other

5.2.2.2. What will your system be used for?

Please select a category below. To make it more clear what each of the categories describes, more detail is given ((a) – (x)). Please feel free to delete certain characteristics if they do not apply to your use case. Bold writing highlights where we have additional questions for you.

1) Standard industrial application
    a) pre-programmed task
    b) external interface (if any) only necessary for synchronisation
2) Advanced industrial application with non-continuous feedback control (i.e., only "look-then-move")
Please tell us: I am operating in a dynamic / static environment (delete as appropriate)
    a) pre-programmed task
    b) external sensors, but only discrete measurements
    c) the industrial controller still performs certain tasks **(delete / add as appropriate)**: path planning, interpolation, inverse kinematics, …
    d) simple interface sufficient (exchange of data without real-time requirements)
    e) real-time / non-real-time vs. (non-)continuous are two different issues

3) Advanced industrial application with continuous feedback control
Please tell us: I am operating in a dynamic / static environment (delete as appropriate)
   a) pre-programmed task
   b) external sensors used for continuous feedback control
   c) use of one or more sensor types **(delete / add as appropriate)**: cameras, force sensor, torque sensor, force torque sensor, laser scanners, ultrasound, radar, time of flight sensors, …
   d) major part of application is programmed on industrial controller
   e) sensor data processing is programmed on a separate system
   f) low cycle time and minimal dead time of feedback control is important for sensor-based control → real-time interface: exchange of data in fixed time intervals, e.g. interpolation cycle time)
4) Robot is used for research outside robotics field:
   a) robot is used for research outside the field of robotics, e.g., robot is used to automate measurements
   b) use cases 1, 2 or 3 above may be applicable **(please circle)**
5) Robotics research – system / application level:
   a) robot is used as part of a larger system to realise and evaluate new applications in the area of **(delete / add as appropriate)** artificial intelligence, cognitive systems, service robotics, …
   b) integration of robot controller in other systems should be easy
   c) functionality of robot controller should be controllable from outside
6) Robotics research – control level:
   a) robot is used to implement and evaluate new robotics algorithms in the area of control **(delete / add as appropriate)** inverse kinematics, dynamics, force control, visual servoing, …
   b) control of robot systems at low level (real-time constraints)
7) Robotics research – haptics:
   a) in this use case you can not afford delays greater than 10ms
   b) robot is used as haptic input device (e.g. for virtual reality) or slave for tele-presence systems; high sensitivity for force control (< 10 N) **(delete / add as appropriate)**
   c) control of robot systems at lowest level possible (real-time constr.)

5.2.2.3. Which product vision is most like your robot?

Please circle the one(s) most relevant or add your personal comment.

- Industrial robotics
    - Large Structure Manufacturing (incl. civil eng.)
    - Robot with Integrated Process Control
    - Rapidly Adaptable Manufacturing Cell
    - Coordinated Mobile Manipulator
    - Human-like Assembly Robot
    - Robot Automation for Small-Scale Manufacturing
    - Postproduction Automation (recycling, remanufacturing)
    - Micro-Manufacturing Robot
    - Robot Assistant in Industrial Environments
- Professional service robotics

  - Maintenance Robot
  - Forestry and Agriculture Robot
  - Mining Robot
  - Professional Cleaning Robot
  - Robot Assistant for Professionals
  - Surgical Robot
  - Rehabilitation Robot
  - Site Protection (professional sector)
  - Autonomous Transport of Goods
  - Autonomous Transport of People
  - Underwater Robot
  - Inspection in Environments Inaccessible to Humans
  - Motion Simulator
  - Robot Guide
  - Robot Teacher
  - Robot Trainer
- Domestic service robotics
  - Personal Robot
  - Robot Assistant for the Physically Challenged
  - Site Protection (domestic sector)
  - Robot Companion
  - Robot Toy
- Security robotics product visions
  - Robot Assistant in Security Contexts
  - Border Surveillance
  - Security checks of goods and people
  - Disaster Management
- Space robotics product visions
  - Orbital Robot Agent
  - Planetary Robot Agent
  - Orbital Robot Assistant
  - Planetary Robot Assistant
  - Orbital Robot Explorer
  - Planetary Robot Explorer

5.2.2.4. In which of the following IFR (International Federation of Robotics) classification do you see your use case?

Please select:

1. Industrial branches
   - Agriculture; hunting and forestry; fishing

- Mining and quarrying
- Manufacturing
- Electricity, gas, and water supply
- Construction
- Others…

2. Service robotics by categories and types of interaction
    - Servicing humans
    - Servicing equipment
    - Others… (e.g. surveillance)

5.2.2.5. What are the application requirements of your use case?

Please describe briefly the requirements of your robot in each of the following categories. In other words, what level of performance does your robot have to reach in each of them? Below we give one example per requirement to .

- Sustainability (e.g. avoid use of heavy metals in production):
- Configuration (e.g. system does not need to be configured):
- Adaption (e.g. manual hardware adaptation):
- Autonomy (e.g. autonomous mission execution in structured but changing environments):
- Positioning & mobility (e.g. differential drive robot navigating with 1cm accuracy using safety rated laser scanners):
- Manipulation & grasping (e.g. 1 arm, 2 finger gripper, flexible objects, repeatability 1cm, using 3D camera for visual servoing):
- Robot-robot interaction (e.g. centralised control):
- Human-robot interaction (e.g. verbal interaction with predefined commands):
- Process quality (e.g. planned time and execution time may not differ by more than 10%):
- Dependability which includes reliability, safety, security, availability, maintainability, robustness, integrity (e.g. robust against lighting changes):
- Physical properties (e.g. cost of production, minimum payload, modular exchange of sensors, batteries last for 4h):
- Standardisation (e.g. adheres to ISO 10218):

5.2.2.6. Further information about the platform and your research environment

Please answer the following questions:

- To which aspects of your robot do you need low-level access and why?

- For which tasks would you like to have a high-level API available and why?

5.2.3. Access to a platform

One concern for the BRICS consortium is whether common research platforms would benefit robotics research and if yes, how to best make these available. We may opt to make this happen in one way or another during the project runtime, but whether it will be possible to uphold such an effort after the project has ended also depends on the access models chosen. Please help us to determine the needs of the sector by answering the following questions

5.2.3.1. Do you think common robotic platforms will be beneficial to the field?

Please delete as appropriate: Yes / No.

5.2.3.2. Would you be interested in such a platform?

Please delete as appropriate: Yes / No.

Please state which of the following are important:

Important (in decreasing importance): _______

Not important: _____

a) I avoid from scratch developments
b) I will be able to do benchmarking
c) The robot I would have access to is likely to be of higher performance than the one I could build
d) The robot would have well documented interfaces and service/support from a manufacturer.
e) trying out research in a different domain (on a different type of platform)
f) other: ________

5.2.3.3. The actual and potential BRICS platforms

The actual and potential BRICS platforms are shown below:

Description from left to right:

1) KUKA omniRob: this platform consists of an omni-directional platform based on mecanum wheels and a KUKA lightweight robot.
2) Bluebotics mobile platform: this platform uses a differential drive.
3) Care-O-bot: an omni-directional platform with a Schunk arm.
   The Care-O-bot will also be equipped with a LWR. Would this be more attractive or less attractive? Why?
4) Bluebotics Gilberto: an omni-directional platform with human-robot interaction interfaces
5) KUKA youBot: a small, omni-directional platform with a 5 degree of freedom arm for educational or entertainment purposes.

Which of these platforms would be most useful for you and why?

5.2.3.4. What would your budget for such a platform be?

In the context of use case described, how much would be a reasonable price to pay? Please state the overall amount for the platform.

Note: Should you not be able to purchase such a platform, different access models could be developed (see next question). Even if you already know you would use such an option, please state the overall cost of the platform here. The cost of such access models can then be calculated by the consortium and the values are more comparable.

5.2.3.5. Access models

Besides buying a platform, several alternative access models could be imagined. This could be appropriate if the cost for the platform, which suits your needs best, exceeded your budget. Please consider the following scenarios:

a) Buying a new platform
b) Buying a used platform
c) <u>“CERN of robotics”</u>: the platforms are installed at one particular location. Researchers then visit for a certain period of time to implement his or her research on the platforms.
d) <u>“Rent-a-robot”</u>: renting or leasing a “standard” robot for a limited amount of time, which is configured for particular R&D purposes and set up at your location
e) “<u>Rent</u>-to-buy-a-robot”: similarly to the model above, here you rent a robot, but you have the option of buying it at the end of the lease for a price considering payments made already.
f) <u>“Local robotics clusters”</u>: setting up of a higher number of local robotics clusters as a mixture of both before-mentioned models. Several locally adjacent laboratories share the cost for buying an RRP. They also share the responsibility of maintaining it and finding a way of scheduling the access. Please state whether you would be willing to / interested in *starting a cluster*, *running an existing cluster*, or *joining an existing cluster* **(please delete as appropriate)**.
g) <u>“Tele-operation”</u> (tele-operated RRPs): Remote access to a physical platform as tele-operated robot, installation at one or several sites
h) <u>“Research Camps”:</u> attend a research camp where you have to work on a topic specified by the camp organiser but for the period of the camp gain access to this platform.

Please tell us which options are not possible for you and why (e.g.: We can not rent a robot as the procedures at our institution do not allow for this):

Please list those access models you would be interested in (in decreasing order of preference): Why? (e.g. we prefer to buy a robot as we can then do what we like and always have access.)

If you selected options c), f) or g), please state how long would you need exclusive access (e.g., 1 month every quarter):

5.2.3.6. How to organise benchmarks or competitions?

How would you organise benchmarks or competitions? Would you use these common platforms? Would you use the RoboCup approach? What would you change? What kind of challenges would be most interesting to you? Please elaborate:

6. Conclusions

Two questionnaires have been evaluated to date. The outcome of the first questionnaire can be summarised as follows: the community would see a real benefit in a common research platform as long as it is modular enough to allow individual research topics to be followed and the pricing is such that it would reduced costs overall. The consortium used these first results and their own experiences for initial decisions regarding, for example, the showcases. The results were described in more detail in Sections 2.2.

The second questionnaire focused on requirements for the interface of the KUKA LWR. The conclusions of each question are summarised in each of the subsections of Section 3.1. Based on its findings the interface was designed as described in Section 3.2. This interface has been used several times and to date the feedback was positive. One improvement has been suggested: if the interface returned a quality of service indicator, the controller may be able to adjust to the situation. KUKA has developed suggestions on how its controller could be expanded to fulfil various requirements stated which is currently being discussed within the consortium.

The hardware needs for the showcases has been analysed as described in Section 4. Especially questions 4 and 5 of the first questionnaire (Sections 2.1.4 and 2.1.5) gave feedback on the requirements of the community. If the “kit” included all components required for mobile manipulation, this would allow the majority of research interests to be followed. Such a system would consist of an ideally omni-directional mobile platform, one or two manipulators, a gripping device, and several different sensing components. The number of degrees of freedom required for the manipulator(s) depends on the application, but it can be said that generally 6 are a must and 7 are desirables, in particular if more than one manipulator is used. The nature of the gripping device again depends on the application but two or three finger grippers or dexterous hands were required most often. A wide variety of sensing devices were applicable ranging from vision, via laser scanners (mostly for navigation and safe mobility purposes) to time of flight sensors.

A third questionnaire has been designed to close some of the gaps left by the first two questionnaires. This questionnaire was presented and the reasoning behind its layout has been described. Once its findings are available an update of this deliverable will be made available.