Into the cloud - How we use to begin a Hands-On

First assignment:
First assignment:

- Install Linux
Into the cloud - How we use to begin a Hands-On

First assignment:

- Install Linux
- Set up SSH and GitHub
Into the cloud - How we use to begin a Hands-On

First assignment:

• Install Linux
• Set up SSH and GitHub
• Install Robot Operating System (ROS)
Into the cloud - How we use to begin a Hands-On

First assignment:
- Install Linux
- Set up SSH and GitHub
- Install Robot Operating System (ROS)
- Install this...
Into the cloud - How we use to begin a Hands-On

First assignment:

- Install Linux
- Set up SSH and GitHub
- Install Robot Operating System (ROS)
- Install this...
- Install that...
Into the cloud - Setup is frustrating

• Requires specific operating system
Into the cloud - Setup is frustrating

- Requires specific operating system
- Collides with existing software
Into the cloud - Setup is frustrating

- Requires specific operating system
- Collides with existing software
- Complex and fragile setup takes time
Into the cloud - Setup is frustrating

- Requires specific operating system
- Collides with existing software
- Complex and fragile setup takes time
- Documentation has low priority
Into the cloud - Virtualization

How can I make my platform easier accessible?
Welcome to the demo scenario of euROBIN. You will learn how to write a sequence of tasks to make a simulated robot

- open a door
- carry a package
- pick an item out of that package onto the table.

We will work with the TIAgO robot in the IAI Bremen apartment laboratory. Open a new tab in Jupyter Notebook, then open a Terminal. Execute the following launch file in that terminal to launch the environment.

```
roslaunch cram_projection_demos apartment_tiago.launch
```

Load the CRAM system for our robot demos. Hit CTRL-Enter in the code-blocks to execute them. Wait until the `[*]` symbol turns into a number. Hide the output by hitting the blue vertical bar to the left.

```
[: (asdf:load-system :cram-projection-demos)]
```

Initialize the simulation.

```
[: (roslib:utilities:startup-ros)]
```

The robot appears at 0.0 by default. Position it in the apartment.

```
[: (btr-utils:move-robot '(9.3 0.0 0.1))]
```

```
[: (urdf-proj:with-projected-robot
  (demos::eurobion-demo))]
```
Agenda

1. Abstract Machine

2. CRAM Plan Executive
   - Primitives
   - Parameters
   - Designators

3. Tutorials
Agenda

1. Abstract Machine

2. CRAM Plan Executive
   - Primitives
   - Parameters
   - Designators

3. Tutorials
Motivation

One plan to accomplish all variations of fetch and place:

- different \textit{objects, environments, robot platforms, applications}. 
Abstract Machines in Computer Science

Adapted from Pedro Domingos: “What’s Missing in AI: the Interface Layer”

<table>
<thead>
<tr>
<th>Field</th>
<th>Interface Layer</th>
<th>Below the Layer</th>
<th>Above the Layer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating Systems</td>
<td>virtual machines</td>
<td>hardware</td>
<td>software</td>
</tr>
<tr>
<td>Programming</td>
<td>high-level</td>
<td>compilers, optimizers, ...</td>
<td>programming</td>
</tr>
<tr>
<td>systems</td>
<td>languages</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Databases</td>
<td>relational model</td>
<td>query optimization, db design,</td>
<td>enterprise</td>
</tr>
<tr>
<td></td>
<td></td>
<td>transaction mgmt</td>
<td>applications</td>
</tr>
</tbody>
</table>
Abstract Machines in Computer Science

Adapted from Pedro Domingos: “What’s Missing in AI: the Interface Layer”

<table>
<thead>
<tr>
<th>Field</th>
<th>Interface Layer</th>
<th>Below the Layer</th>
<th>Above the Layer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating Systems</td>
<td>virtual machines</td>
<td>hardware</td>
<td>software</td>
</tr>
<tr>
<td>Programming systems</td>
<td>high-level</td>
<td>compilers,</td>
<td>programming</td>
</tr>
<tr>
<td></td>
<td>languages</td>
<td>optimizers, ...</td>
<td></td>
</tr>
<tr>
<td>Databases</td>
<td>relational model</td>
<td>query optimization, db design,</td>
<td>enterprise applications</td>
</tr>
<tr>
<td></td>
<td></td>
<td>transaction mgmt</td>
<td></td>
</tr>
</tbody>
</table>
| **Personal robotics**  | **CRAM**        | grounding in robot, AI tools,     | robot application pro-
|                        |                 | the nuts and bolts of intelligent| gramming              |
|                        |                 | robotics, ...                     |                       |

Raise the conceptual level at which service and personal robot applications are programmed!
Agenda

1 Abstract Machine

2 CRAM Plan Executive
   Primitives
   Parameters
   Designators

3 Tutorials
CRAM 2.0 system

COGITO
Metacognition
- Introspection
- Self-programming

Plan Executive
Generalized action plan
- Contextualize Action Designator
- Motion plan

RoboKudo
Perception Executive

Giskard
Action Executive

KnowRob 2.0
- Question answering
- Logic-based language
- Hybrid reasoning
- Ontology & axiomatizations
- Symbolic knowledge base
- Data structures
- NEEMs (episodic memory)

What, why, how?
Answer
Body motion query
Motion specifications

IROS 2023 Tutorial: Robot Programming with CRAM
Challenges Tackled by the Plan Executive

1. Define which actions to execute to achieve the goal.
2. Infer which parameters to use for each action.
3. Monitor task execution and react to failures.
### Primitives of Mobile Pick and Place for PR2-like Robots

<table>
<thead>
<tr>
<th>Primitive</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>moving-base</td>
<td>Move the base to the target pose.</td>
</tr>
<tr>
<td>moving-arm</td>
<td>Move the joints of the arm / arms to the target configuration in joint, cartesian or constraint space.</td>
</tr>
<tr>
<td>moving-finger</td>
<td>Move the joint of the hand / hands to the target joint position.</td>
</tr>
<tr>
<td>gripping</td>
<td>Close the hand / hands to grasp an object.</td>
</tr>
<tr>
<td>moving-torso</td>
<td>Move the torso joint to the target joint position.</td>
</tr>
<tr>
<td>moving-neck</td>
<td>Move the neck to the target configuration or to direct the camera gaze to a target pose.</td>
</tr>
<tr>
<td>detecting</td>
<td>Detect the described object in the environment and update the internal world state with the acquired information.</td>
</tr>
<tr>
<td>monitoring-joint-states</td>
<td>Monitor if the joint positions of robot body parts exceed the given threshold.</td>
</tr>
</tbody>
</table>
Parameters of Motion and Perception Primitives

<table>
<thead>
<tr>
<th>Primitive</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>moving-base</td>
<td>goal_pose, ..., speed, ...</td>
</tr>
<tr>
<td>moving-arm</td>
<td>goal_pose_for_hand, goal_positions, collisions, ...</td>
</tr>
<tr>
<td>moving-finger</td>
<td>goal_position</td>
</tr>
<tr>
<td>gripping</td>
<td>hand, grasping_force, object_properties, ...</td>
</tr>
<tr>
<td>moving-torso</td>
<td>goal_position, ...</td>
</tr>
<tr>
<td>moving-neck</td>
<td>goal_positions, goal_coordinate_to_look_at, ...</td>
</tr>
<tr>
<td>detecting</td>
<td>object_description, ...</td>
</tr>
<tr>
<td>monitoring-joints</td>
<td>joint_name, joint_value, monitoring_function, ...</td>
</tr>
</tbody>
</table>

Calculating parameter values that maximize success probability: heuristics, learning from experience, imitation learning, ask a human
Choice of Parameter Values is Crucial For Success

- Often very many possible values to choose from
  Example: from which side and with which hand to grasp?

- Effects can be:
  - immediate
  - short-term
  - long-term
(a location (on CounterTop))
Action Designators: Searching

Combining primitives into high-level actions
Agenda

1. Abstract Machine

2. CRAM Plan Executive
   - Primitives
   - Parameters
   - Designators

3. Tutorials
Robot Control with PyCRAM

http://cram-system.org/tutorials/
<table>
<thead>
<tr>
<th>Time</th>
<th>Session</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.30 - 8.45</td>
<td>Opening: Michael Beetz &amp; Jörn Syrbe</td>
</tr>
<tr>
<td>8.45 - 10.00</td>
<td>Introduction - Michael Beetz</td>
</tr>
<tr>
<td>10.00 - 11.00</td>
<td>COFFEE BREAK</td>
</tr>
<tr>
<td>11.00 - 12.30</td>
<td>Hands-on Robot Control in CRAM – Arthur Niedźwiecki</td>
</tr>
<tr>
<td>12.30 - 1.30</td>
<td>LUNCH</td>
</tr>
<tr>
<td>1.30 - 3.00</td>
<td>Hands-On Robotics Simulation in Multiverse – Giang Nguyen</td>
</tr>
<tr>
<td>3.00 - 4.00</td>
<td>COFFEE BREAK</td>
</tr>
<tr>
<td>4.00 - 5.30</td>
<td>Hands-On Knowledge openEASE – Sascha Jongebloed</td>
</tr>
<tr>
<td>5.30</td>
<td>End</td>
</tr>
</tbody>
</table>