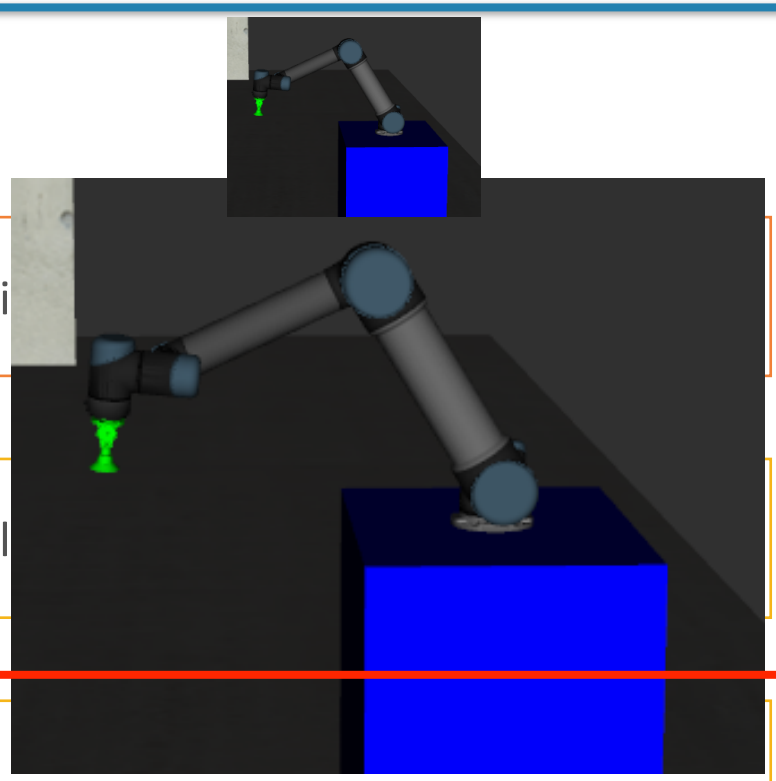
1.5 ROS Actions – “client-server” communication

Mukunda Bharatheesha

A case for non-blocking execution



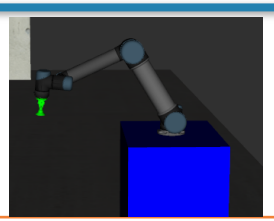
Pick Pi

Place Pl

Delive bile robot.

GoBack Get back for the next object.

A case for non-blocking execution



Pick	Pick an object.
Place	Place the object on the mobile robot.
Pick	Pick an object.

Deliver	Transport object to destination with mobile robot.
GoBack	Get back for the next object.

ROS Actions - introduction

- **ROS Actions** - No waiting until an execution is complete (non-blocking!)
  - waiting is an option if required.

- A generalized request-response system - the **client-server** infrastructure in ROS.

- Actions are defined by three message types: **goal** (request), **result** (response) and **feedback**.

ROS Actions – an example action definition

- Requirement: counter with a 1s delay between each count.
  - goal message – number to count up to (uint32)
  - result message – status message (string)
  - feedback message – number of counts completed (uint32)

hrwros_msgs/action/CounterWithDelay.action

```
uint32 num_counts
---
string result_message
---
uint32 counts_elapsed
```

goal field

result field

feedback field

ROS Actions – on the filesystem, utility cmds

- ROS action definitions reside in the ROS package with project specific message definitions
  - hrwros_msgs/**action** folder.

generate action messages manually

```
$ rosrun actionlib_msgs genaction.py <path_to_action_file>
```

show the contents of an action message

```
$ rosmsg show <stack_name>_msgs/<ActionMessage>
```

ROS Actions – processing a goal request

- **goalCallback function** processes a goal request
  - a goal can be pre-empted (and cancelled) **before** completion.

- **goal statuses**: ACTIVE, SUCCEEDED, ABORTED.

http://wiki.ros.org/actionlib/DetailedDescription

ROS Actions – code nomenclature

- A ROS Node: **action server**
  - “advertises” an action so that other nodes can request action goals to be processed.

- A ROS Node: **action client**
  - can “send” goal requests to the action server.