

# Cognitive Interaction Toolkit - Feature #413

## CITK rapman integration

2015-08-07 20:29 - Florian Lier

<b>Status:</b>	Closed	<b>Start date:</b>	2015-08-07
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Jon Weisz	<b>% Done:</b>	0%
<b>Category:</b>	docker	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>		<b>Spent time:</b>	0.00 hour
<b>Description</b>			
<p>This issue is mainly targeted at the integration of rapman (<a href="https://github.com/Robobench/robobench-subuser-registry">https://github.com/Robobench/robobench-subuser-registry</a>) into the CITK. The aim of this task is to create docker images with X11/GPU support based on the CITK distribution tool chain.</p> <p>Preliminary roadmap:</p> <ol style="list-style-type: none"><li>1) Let's start with the <a href="https://opensource.cit-ec.de/projects/citk/repository/revisions/master/entry/distributions/icub-iros2015.distribution">https://opensource.cit-ec.de/projects/citk/repository/revisions/master/entry/distributions/icub-iros2015.distribution</a></li><li>2) Target system will be Ubuntu 14.04 (x64).</li><li>3) Install "base" packages for the CITK tool chain as explained in the tutorials section (<a href="https://toolkit.cit-ec.uni-bielefeld.de/bootstrapping-local-jenkins-distribution-deployment">https://toolkit.cit-ec.uni-bielefeld.de/bootstrapping-local-jenkins-distribution-deployment</a>).</li><li>4) Download the jenkins.tar.gz, extract, create_user, fire up Jenkins</li><li>5) Clone the distribution repository</li><li>6) Select icub-iros2015.distribution and parse the distribution specific packages from distribution file (later this list will be aggregated from the included *.project files)</li><li>7) Execute job generator as explained on the replication page (<a href="https://toolkit.cit-ec.uni-bielefeld.de/node/1399">https://toolkit.cit-ec.uni-bielefeld.de/node/1399</a>)</li><li>8) Implement a mechanism in order to detect successful build. This can be done using the Jenkins REST API, here is a start: <a href="https://opensource.cit-ec.de/projects/citk/repository/docker/revisions/master/entry/testJob.sh">https://opensource.cit-ec.de/projects/citk/repository/docker/revisions/master/entry/testJob.sh</a></li><li>9) Execute the experiment already included in the distribution as explained here: <a href="https://toolkit.cit-ec.uni-bielefeld.de/experiments/icub-ball-tracking-iros2015">https://toolkit.cit-ec.uni-bielefeld.de/experiments/icub-ball-tracking-iros2015</a> (Execution part)</li><li>10) In order to achieve 9) X11/GPU acceleration is required</li><li>11) Setup a github docker repository for the CITK</li><li>12) Upload icub-docker image</li><li>13) Write a cmd line tool in order to automate this process</li></ol>			

### History

#### #1 - 2015-09-01 15:53 - Florian Lier

Yay! Finally the first update on this issue. I managed to install the citman (also had to fix the caching mechanism, please see latest commit, revert if I made things worse). I was also able to install a demo. I finally got stuck when running the demo:

```
user@user-Latitude-E4300:~/docker-cli/citk-cli$ ./citman run-demo --demo=fsmt-exp-run-icub-nightly-balltracking-sim
Traceback (most recent call last):
  File "./citman", line 211, in <module>
    c = CitManager(sys.argv[1:])
  File "./citman", line 71, in __init__
    self.__parse(inputs)
  File "./citman", line 77, in __parse
    self.subcommands[which](result_dict)
  File "./citman", line 22, in __call__
    self.__method(**kwargs)
  File "./citman", line 206, in run_demo
    return self.dm.run_demo(distro, project=demo, use_system_dist=True)
  File "/home/user/docker-cli/citk-cli/distribution_image_manager.py", line 467, in run_demo
    cmd = self.get_run_command(project)
  File "/home/user/docker-cli/citk-cli/distribution_image_manager.py", line 521, in get_run_command
    cmd = self.projects.projects[project_name]['variables']['shell.command']
KeyError: 'fsmt-exp-run-icub-nightly-balltracking-sim'
```

Here are a few thoughts for the next hangout:

- 1) Make a proper setup.py for the citman

- 2) Discuss Docker non-root usage
- 3) How to integrate the citman into CITK, i.e., deployment via recipes

## #2 - 2015-09-05 22:00 - Florian Lier

Okay this time the installation went smoothly. Also the distro and demo listing is nice.

However, this happened:

```
~/docker/citk-cli$ ./citman install-demo --demo=fsmt-exp-icub-nightly
Attempting Download
{u'errorDetail': {u'message': u'Error: image citk/icub-iros2015-trus:build-finalized not found'}, u'error': u'Error: image
citk/icub-iros2015-trus:build-finalized not found'}
Warning: Distro icub-iros2015 could not be downloaded.
```

## #3 - 2015-09-05 23:35 - Florian Lier

Okay, the status update above is to be considered useless, since I configured my FW using 'paranoia mode'. After allowing a few ports binding to localhost, e.g, docker **sigh**. I managed to achieve this:

```
~/docker/citk-cli$ sudo ./citman install-demo --demo fsmt-exp-morse-12-findhuman
Attempting Download
{u'errorDetail': {u'message': u'Error: image citk/morse-1.2-stable-trus:build-finalized not found'}, u'error': u'Error: image
citk/morse-1.2-stable-trus:build-finalized not found'}
Warning: Distro morse-1.2-STABLE could not be downloaded.
Success: Distro morse-1.2-STABLE build and demo ready.
```

However, running the demo resulted in this

```
~/docker/citk-cli$ ./citman run-demo --demo fsmt-exp-morse-12-findhuman
> /home/miri/docker/citk-cli/distribution_image_manager.py(574)get_run_command()
-> logging.error('Failed to find command for project name %s'%(project_name))
```

## #4 - 2015-09-06 00:16 - Florian Lier

Last update for today ... I just noticed that the machine has a nvidia card but uses the nouveau driver ...

## #5 - 2015-09-24 22:53 - Florian Lier

- Status changed from In Progress to Closed