dockeROS

Simply running ros nodes in docker containers on remote robots.

docker + ☛ROS = ❤️
Cloud Robotics

“Making use of centralized computational resources in robotics.”

- Unlimited computational power
- Distributed Systems
- Usage-based billing
- On-demand Services
- Edge Computing
- ROS can do a lot of this
- Especially with ROS2
- Open challenge: deployment
Example: Cloud Navigation

- Holistic Environment Model
- Global Planning
- External Sensors / Localization
- Cost Scaling Sensors / Computation

Source:
Learnings from Cloud Navigation

- Cloud deployment: done
- Integration in Robot: done
- Integration to Robot: done
  - vfk_msb_client
- Open point: Robot deployment
  - Many systems
  - Industrial Environments
Docker Intro

- **Image**
  - A binary contained with all its dependencies
  - A VM but **not**

- **Dockerfile**
  - Defines Image
  - Incremental

```
FROM ros:kinetic-ros-base
RUN apt-get update
RUN mkdir -p /ws/src/hello_world
COPY . /ws/src/hello_world
ENV ROS_PACKAGE_PATH=/ws/src/hello_world
RUN rosdep install -y -r --from-path /ws/src
RUN source /opt/ros/$ROS_DISTRO/setup.bash;
   cd /ws/src;
   catkin_init_workspace;
   cd /ws;
   catkin_make
RUN rm -rf /var/lib/apt/lists/*
CMD ["/ros_entrypoint.sh", 
    "rosrun", "hello_world", "talker"]
```
Docker Intro II

- **Registry**
  - A repository to store images
  - Public: hub.docker.com

- **Container**
  - A running image

- **Docker Host**
  - A place to run an image
  - Remotely accessible
dockeROS

Simply running ros nodes in docker containers on remote robots.

1. UX
2. “only” plumbing
dockeROS

- https://github.com/ct2034/dockeros
- License: BSD
- Reimplemented from zero
- <1000 lines of code
- Library
  - CLI
  - ...

  {build,run,stop,push} ...
Simply running ros nodes in docker containers on remote robots.
positional arguments:
  {build,run,stop,push}build: Creates an image that can run
roscommandrun: Runs an image with your_roscommand
(and builds it first)
  stop: Stops image that runs that command
  push: Push image to predefined registry
roscommand Everything after the subcommand will be
interpreted as the ros command to be run in your image
optional arguments:
  -h, --help show this help message and exit
  -e, --env use the existing docker environment (see
    https://dockr.ly/2zMPc17 for details)
  -i HOST:PORT, --ip HOST:PORT, --host HOST:PORT
    set the host (robot) to deploy image to
  -f DOCKERFILE, --dockerfile DOCKERFILE
    use a custom Dockerfile
  -n, --no-build dont (re-)build the image before running
dockerROS: Architecture

docker run rosrun amazing_pkg amazing_bin
dockeROS: CLI

dockeros run rosrun amazing_pkg amazing_bin

- Build
- Run
- Stop
- Push

- User defined packages
- System packages

- User defined Dockerfiles
- Default Dockerfiles
Edge Computing in Automation

- Web-based GUI
- Define SW running on edge devices
- dockeros is part of it
dockeROS: Bottom Line

- Simply running ros nodes in docker containers on remote robots.
- Future work:
  - Uses of library
  - roslaunch
  - ROS2
- **We are taking pull requests**
  - https://github.com/ct2034/dockeros
- Contact:
  - christian.henkel@ipa.fraunhofer.de