The hard life of non-admins OR System independent virtual environments: `Conda` and `BioConda`

K. Riege[1,2]

[1]RNA Bioinformatics and High Throughput Analysis, FSU Jena
[2]Interdisciplinary Center for Bioinformatics, University Leipzig

Machine Learning Selbsthilfegruppe

Motivation

Usecase: Install software on a nearly bare-metal machine
Problems:

- Missing compilers
- Missing script languages
- Missing libraries/modules
- Missing root access
- Missing w3 access
- Missing efforts for manual gefrickel

System C(++)

```
DIR="/my/ctools"
LIB="/my/clibs"
TMP="/my/tmp"

if [[ $download ]]; then
    wget -q "https://ftp.gnu.org/pub/gnu/ncurses/ncurses-6.0.tar.gz" -O "ncurses.tar.gz"
    tar -xzf ncurses.tar.gz -C $TMP
fi

cd $TMP/ncurses*
make clean
./configure --prefix=$LIB
make -j $THREADS
make install
```

System C(++) Method 1

```
if [[ $download ]]; then
  wget -q "http://www.bioinf.uni-leipzig.de/Software/segemehl/segemehl_0_2_0.tar.gz" -O "segemehl.tar.gz"
  tar -xzf segemehl.tar.gz -C $TMP
fi

# inlcudes/libaries for *FLAG options -isystem and e.g. -lncurses
export C_INCLUDE_PATH=$LIB/include
export CPLUS_INCLUDE_PATH=$LIB/include
export LD_LIBRARY_PATH=$LIB/lib
# inlcudes/libaries for *FLAG options -I -L
export CPATH=$LIB/include
export LIBRARY_PATH=$LIB/lib

cd $TMP/segemehl*
make clean
./configure --prefix=$DIR
make -j $THREADS
make install
```

System C(++) Method 2

```
if [[ $download ]]; then
  wget -q "http://www.bioinf.uni-leipzig.de/Software/segemehl/segemehl_0_2_0.tar.gz" -O "segemehl.tar.gz"
  tar -xzf segemehl.tar.gz -C $TMP
fi

cd $TMP/segemehl*
# use override directive of *FLAGS in Makefile
sed -ir 's/^\s*(C|LD|CPP|CXX)FLAGS\s*=\s*/override \1FLAGS += /' Makefile
make clean
./configure --prefix=$DIR
make -j $THREADS CFLAGS="-I$DIR/include" LDFLAGS="-L$DIR/lib"
make install

export PATH=$DIR/bin
```

System Perl - without dependencies

```
DIR="/my/perllibs"
TMP="/my/tmp"

if [[ $download ]]; then
  wget -q "https://github.com/bioperl/bioperl-live/archive/release-1-7-1.tar.gz" -O "bioperl.tar.gz"
  tar -xzf bioperl.tar.gz -C $TMP
fi

cd $TMP/bioperl*
# PREFIX defines installation DIR
perl Makefile.PL PREFIX=$DIR/perl5
make -j $THREADS
make install

# tell Perl where to find libaries
export PERL5LIB=$DIR/perl5/lib
export PATH=$DIR/perl5/bin
```

System Perl - cpanm way

```
DIR="/my/perllibs"

if [[ $download ]]; then
  wget -q "cpanmin.us" -O $DIR/bin/cpanm
  # -l defines installation $DIR
  $DIR/bin/cpanm -l "/dev/null" --scandeps --save-dists $DIR Bio::Perl
fi

# -l defines installation DIR
$DIR/bin/cpanm -l $DIR --mirror file://$DIR Bio::Perl
[[ $? -gt 0 ]] && return $?

# tell Perl where to find libaries
export PERL5LIB=$DIR/perl5/lib
export PATH=$DIR/perl5/bin
```

System R

```
REPO="/my/rrepo"

if [[ $download ]]; then
  cat << EOF > download.R
    source("https://bioconductor.org/biocLite.R")
    pkgs = c("DESeq2")
    avail = available.packages(contrib.url(BiocInstaller::biocinstallRepos()))
    deps = tools::package_dependencies(pkgs, avail, recursive=TRUE,)
    download.packages(unique(c(pkgs, unlist(unname(deps)))), "$REPO", repos=BiocInstaller::biocinstallRepos())
    # create local index/repository
    tools::write_PACKAGES("$REPO")
  EOF
  Rscript download.R
fi
```

System R

```
DIR="/my/rlibs"
# tell R where to find and install libaries
export R_LIBS=$DIR/lib

Rscript -e "install.packages(c(\"DESeq2\"), contriburl=\"file://$REPO\")"
```

System Python - without dependencies

```
DIR="/my/pythonlibs"
TMP="/my/tmp"

# define installation DIR for --user
export PYTHONUSERBASE=$DIR
if [[ $download ]]; then
  wget -q "https://github.com/marcelm/cutadapt/archive/v1.14.tar.gz" -O "cutadapt.tar.gz"
  tar -xzf cutadapt.tar.gz -C $TMP
fi

cd $TMP/cutadapt*
python setup.py install --forced --user

# tell Python where to find libaries
export PYTHONPATH=$DIR/lib/python2.7/site-packages
export PATH=$DIR/bin
```

System Python - pip way

```
DIR="/my/pythonlibs"

# define installation DIR for --user
export PYTHONUSERBASE=$DIR
if [[ $download ]]; then
  wget -q "https://bootstrap.pypa.io/get-pip.py" -O "get-pip.py"
  python get-pip.py --ignore-installed --user
  $DIR/bin/pip download -d $DIR cutadapt
fi

$DIR/bin/pip install --ignore-installed --user --use-wheel --no-index --find-links=$DIR cutadapt

# tell Python where to find libaries
export PYTHONPATH=$DIR/lib/python2.7/site-packages
export PATH=$DIR/bin
```

System Python - virtual environments

```
DIR="/my/pythonlibs"

# define installation DIR for --user
export PYTHONUSERBASE=$DIR
if [[ $download ]]; then
  wget -q "https://bootstrap.pypa.io/get-pip.py" -O "get-pip.py"
  python get-pip.py --ignore-installed --user
  $DIR/bin/pip download -d $DIR virtualenv cutadapt
fi

$DIR/bin/pip install --ignore-installed --user --use-wheel --no-index --find-links=$DIR virtualenv
```

System Python - virtual environments

```
# ! do never ever move or rename directories afterwards
VENV="/my/virtualenvs/project1"

# specify python version to use [2/3]
$DIR/bin/virtualenv -p python2 $VENV

# load proper shell variables
source $VENV/bin/activate
pip install --ignore-installed --user --use-wheel --no-index --find-links=$DIR cutadapt
source deactivate
```

Fully system independent all in one tool - (Bio-)Conda

```
CENV="/my/conda"
# use linux-64, linux-32, win-32, win-64 and/or osx-64
REPO="/my/condarepo/linux-64"

if [[ $download ]]; then
    wget -q "https://repo.continuum.io/miniconda/Miniconda2-latest-Linux-x86_64.sh" -O installMiniconda.sh
    bash installMiniconda.sh -b -f -p $DIR
    source $DIR/bin/activate
    # adapt ~/.condarc
    conda config --set channel_priority true
    conda config --add channels conda-forge
    conda config --add channels defaults
    conda config --add channels r
    conda config --add channels bioconda
```

Fully system independent all in one tool - (Bio-)Conda

```
    conda install -y conda-build
    conda clean -y --all

    conda remove -y -n project1 --all
    conda create -y -n project1
    source deactivate

    source $DIR/bin/activate project1
    conda install -y segemehl perl-bioperl bioconductor-deseq2 cutadapt

    # create local index/repository/cutom channel
    mv $DIR/pkgs/*.tar.bz2 $REPO
    conda index $REPO
    source deactivate
fi
```

Fully system independent all in one tool - (Bio-)Conda

```
bash installMiniconda.sh -b -f -p $DIR
source $DIR/bin/activate
# adapt ~/.condarc
conda config --set channel_priority false

conda remove -y -n project1 --all
conda create -y -n project1 --offline

source deactivate

source $DIR/bin/activate project1
conda install -y -c file://$REPO --override-channels --offline
    segemehl perl-bioperl bioconductor-deseq2 cutadapt
source deactivate
```

Limitations

Problem1: Conda generates url lookups from version numbers
correct url=openjdk_1.8.0_121
but version=8.17
thus lookup=openjdk_8.17
Solution: Manually install necessary depencies

```
conda install -y -c file://$REPO --override-channels --offline $REPO/openjdk-*.tar.bz2
```

Problem2: Conda environments are fully packed with symlinks which makes moving or copying the directory impossible
Solution: Follow links while compressing

```
tar -c --hard-dereference $CENV | pigz -p $THREADS > $CENV/conda.tar.gz
```

for java based tools and bioconductor packages install

```
conda install -y -c file://$REPO --override-channels --offline $
    REPO/python-dateutil-*.tar.bz2
conda install -y -c file://$REPO --override-channels --offline $
    REPO/bleach-*.tar.bz2
conda install -y -c file://$REPO --override-channels --offline $
    REPO/r-yaml-*.tar.bz2
conda install -y -c file://$REPO --override-channels --offline $
    REPO/r-psych-*.tar.bz2
conda install -y -c file://$REPO --override-channels --offline $
    REPO/r-htmlwidgets-*.tar.bz2
conda install -y -c file://$REPO --override-channels --offline $
    REPO/r-gplots-*.tar.bz2
conda install -y -c file://$REPO --override-channels --offline $
    REPO/r-viridis-*.tar.bz2
conda install -y -c file://$REPO --override-channels --offline $
    REPO/r-bh-*.tar.bz2
conda install -y -c file://$REPO --override-channels --offline $
    REPO/openjdk-*.tar.bz2
conda install -y -c file://$REPO --override-channels --offline $
    REPO/java-jdk-*.tar.bz2
```