

CDE: Run Any Linux Application On-Demand Without Installation



Philip Guo
pg@cs.stanford.edu
<http://www.pgbvine.net/cde.html>

Problem

It's hard to package up your software so that other people can reliably run it ...

Problem

It's hard to package up your software so that other people can reliably run it ...

... and it can be hard to install and run other people's software ...



The screenshot shows a web browser window with the URL mail.scipy.org/pipermail/numpy-discussion/2011-April/056018.html. The page title is "[Numpy-discussion] problem installing numpy on Fedora". The message is from Charles R Harris (charlesr.harris@gmail....) on Wednesday, April 13, 2011, at 09:17:15 CDT. The message content discusses a problem with installing numpy 1.6.0b2 on Fedora. The message has three links in the header: "Previous message: [Numpy-discussion] problem installing numpy on Fedora", "Next message: [Numpy-discussion] problem installing numpy on Fedora", and "Messages sorted by: [date] [thread] [subject] [author]". The message body shows the original text of the email, which includes a greeting, a description of the error, and a stack trace.

Charles R Harris charlesr.harris@gmail....
Wed Apr 13 09:17:15 CDT 2011

- Previous message: [\[Numpy-discussion\] problem installing numpy on Fedora](#)
- Next message: [\[Numpy-discussion\] problem installing numpy on Fedora](#)
- Messages sorted by: [\[date \]](#) [\[thread \]](#) [\[subject \]](#) [\[author \]](#)

On Wed, Apr 13, 2011 at 7:30 AM, c cook <csecook@gmail.com> wrote:

```
> Hello,
>
> I am trying to install numpy 1.6.0b2 version on Fedora and I am getting the
> following error:
>
> In file included from numpy/core/src/private/npy_config.h:4,
>                  from numpy/core/src/multiarray/common.c:8,
>                  from
> numpy/core/src/multiarray/multiarraymodule_onefile.c:8:
> build/src.linux-x86_64-2.4(numpy/core/include/numpy/config.h:34:1: warning:
> "HAVE_HYPOT" redefined
> In file included from /home/cse/Apps/Python-2.4/Include/Python.h:8,
>                  from numpy/core/src/multiarray/common.c:2,
```

Problem

It's hard to package up your software so that other people can reliably run it ...

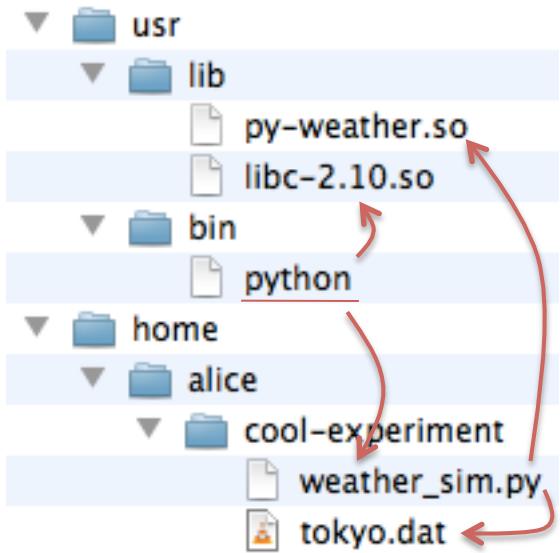
... and it can be hard to install and run other people's software ...

... especially on Linux

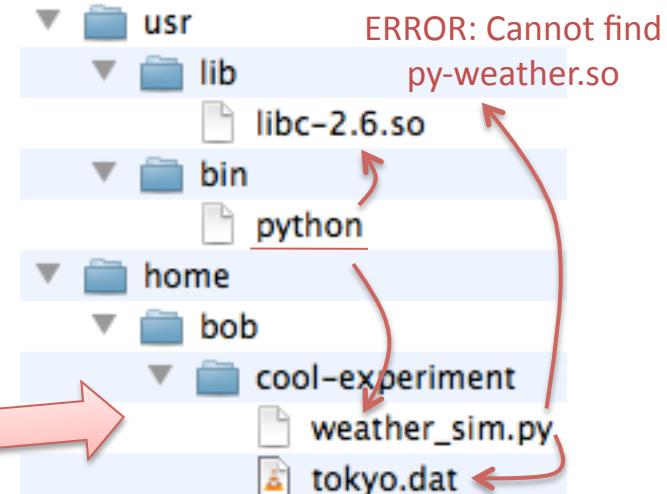


Example: Scientists and Sysadmins

Alice's computer



Bob's computer



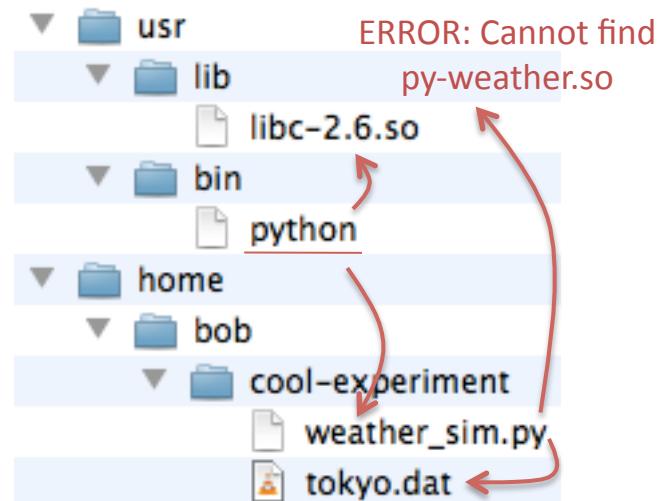
```
cd /home/alice/cool-experiment/  
python weather_sim.py tokyo.dat
```

```
cd /home/bob/cool-experiment/  
python weather_sim.py tokyo.dat
```

Sysadmin woes

Difficult to install
custom versions of
Python and 3rd-party
extension libs on
university machines

Bob's computer



```
cd /home/bob/cool-experiment/  
python weather_sim.py tokyo.dat
```

CDE: Automatic packaging of Code, Data, and Environment



1. Create package on your Linux computer

Prepend any set of commands with '`cde`', and CDE runs them and automatically packages up their dependencies

2. Transfer package

A package is simply a directory of files (~10MB – 500MB), so it can be transferred via USB stick, DVD, or over the Internet

3. Execute software from within package on any modern Linux computer

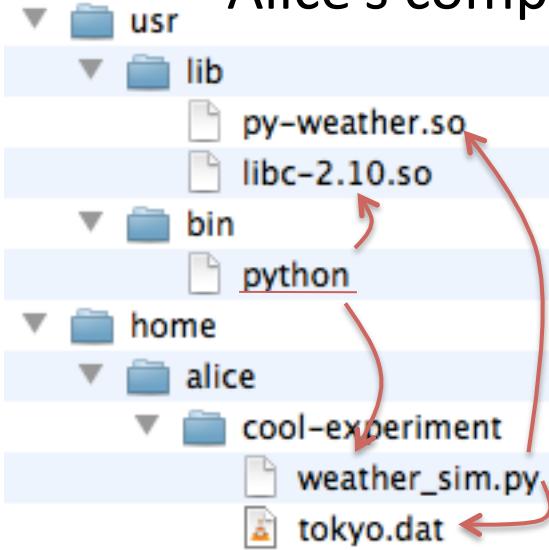
Prepend those same commands with '`cde-exec`', and CDE runs them natively without any installation

1. CDE overview

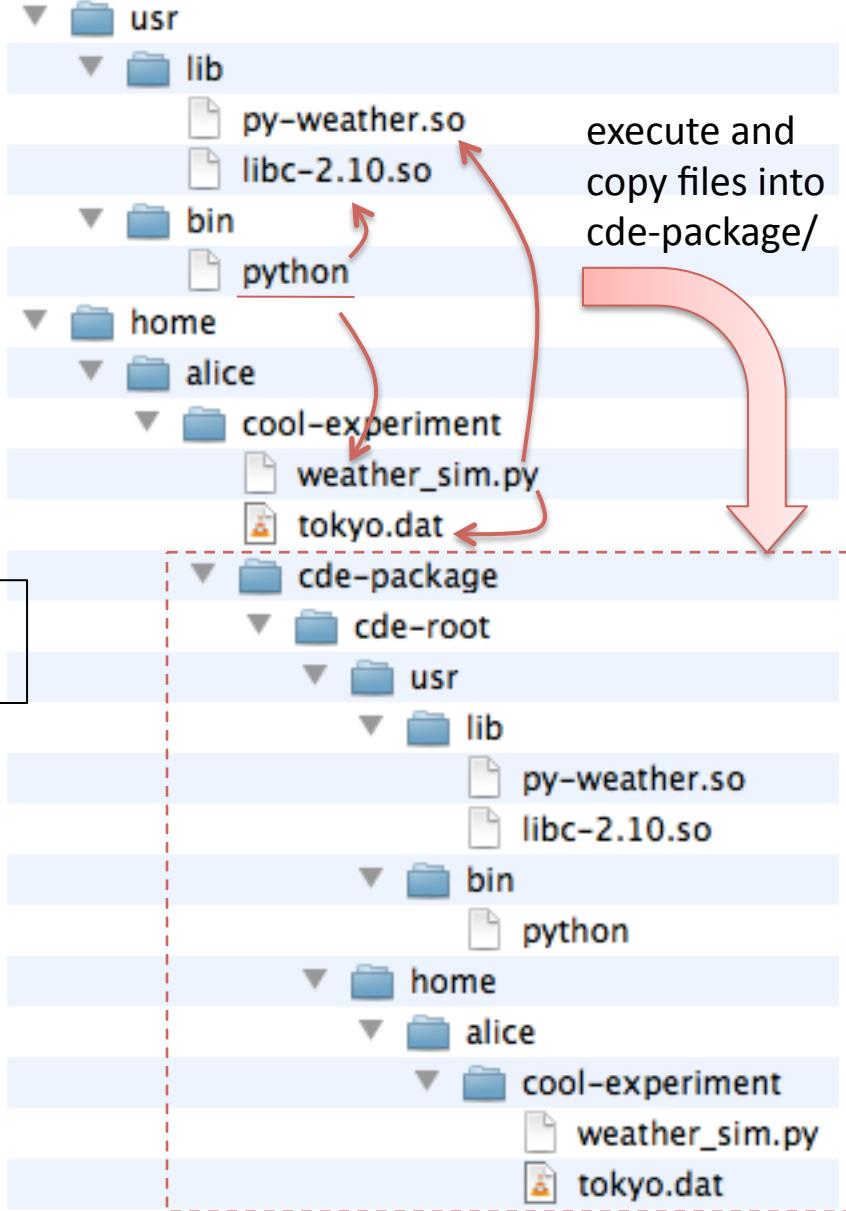
2. Seamless execution

3. Run any app on-demand

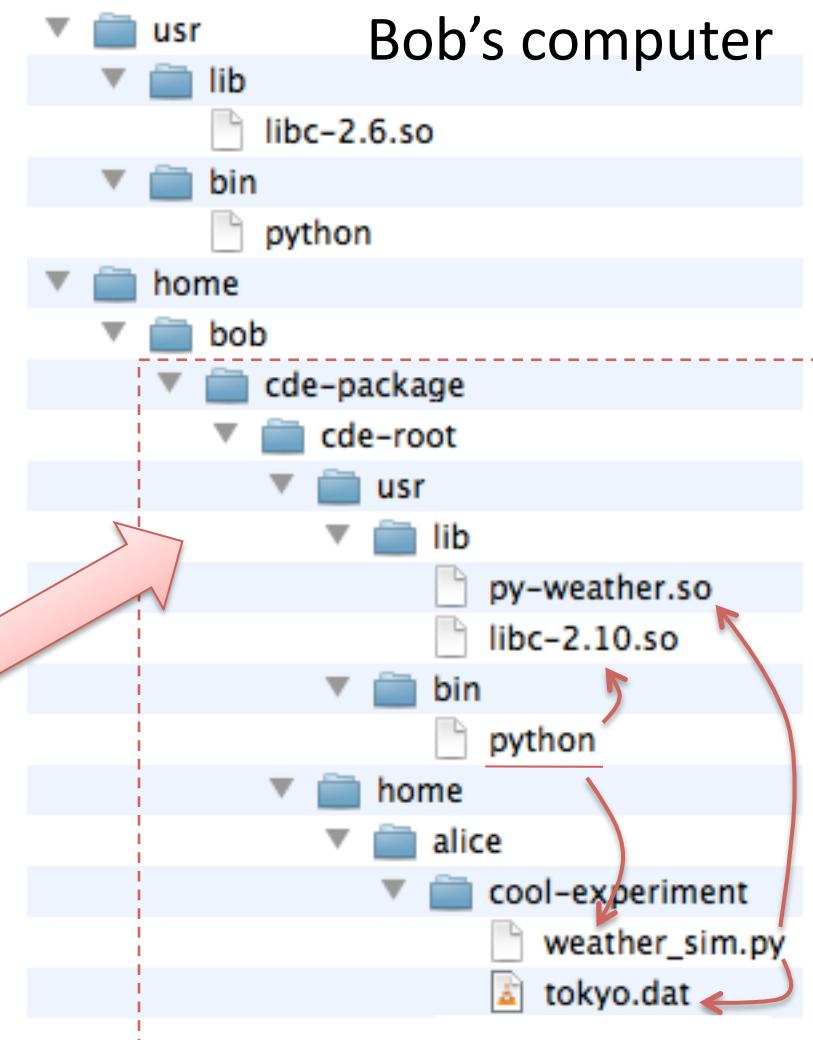
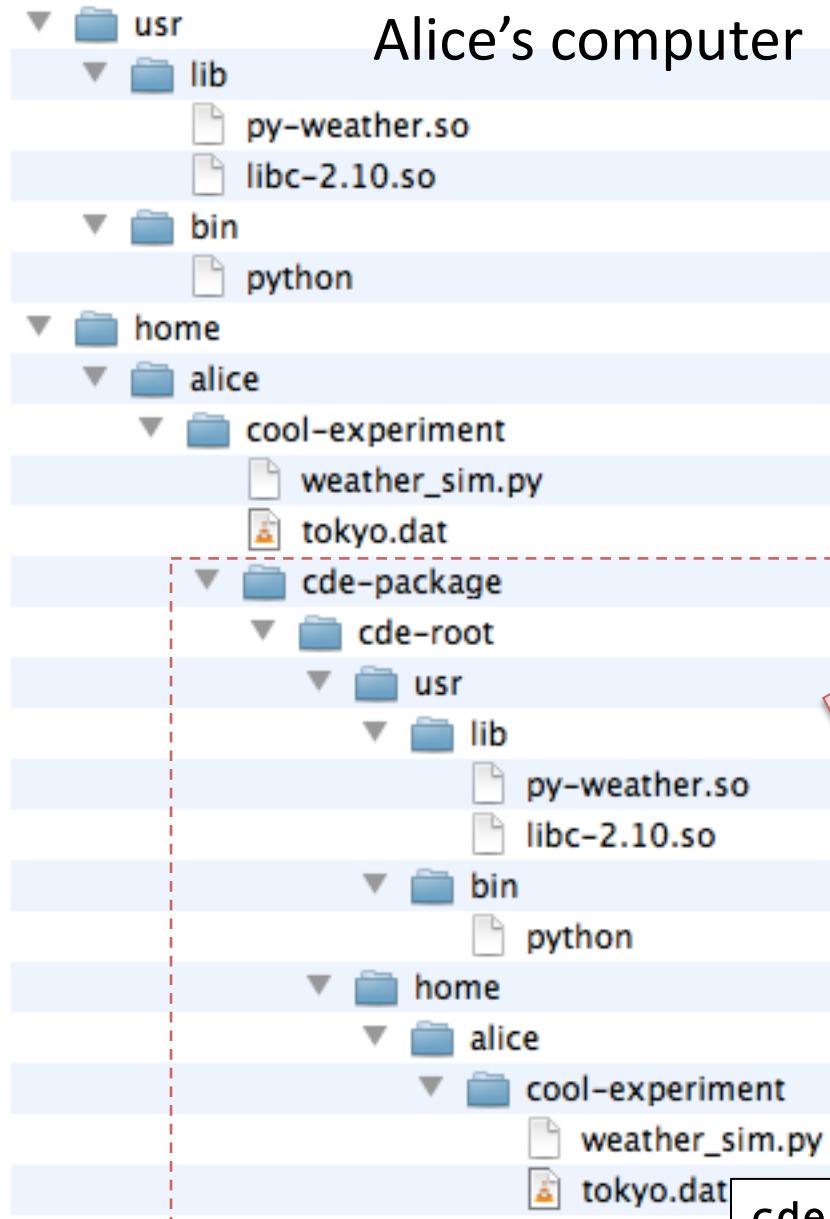
Alice's computer



```
cd /home/alice/cool-experiment/  
cde python weather_sim.py tokyo.dat
```

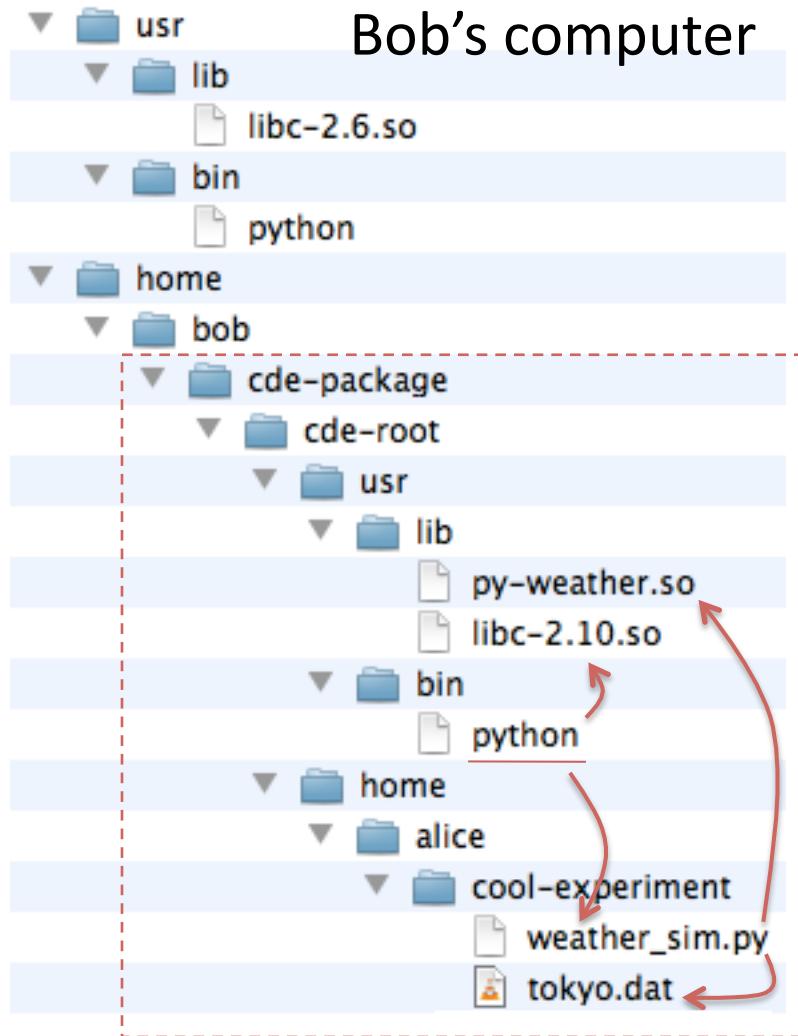


execute and
copy files into
cde-package/



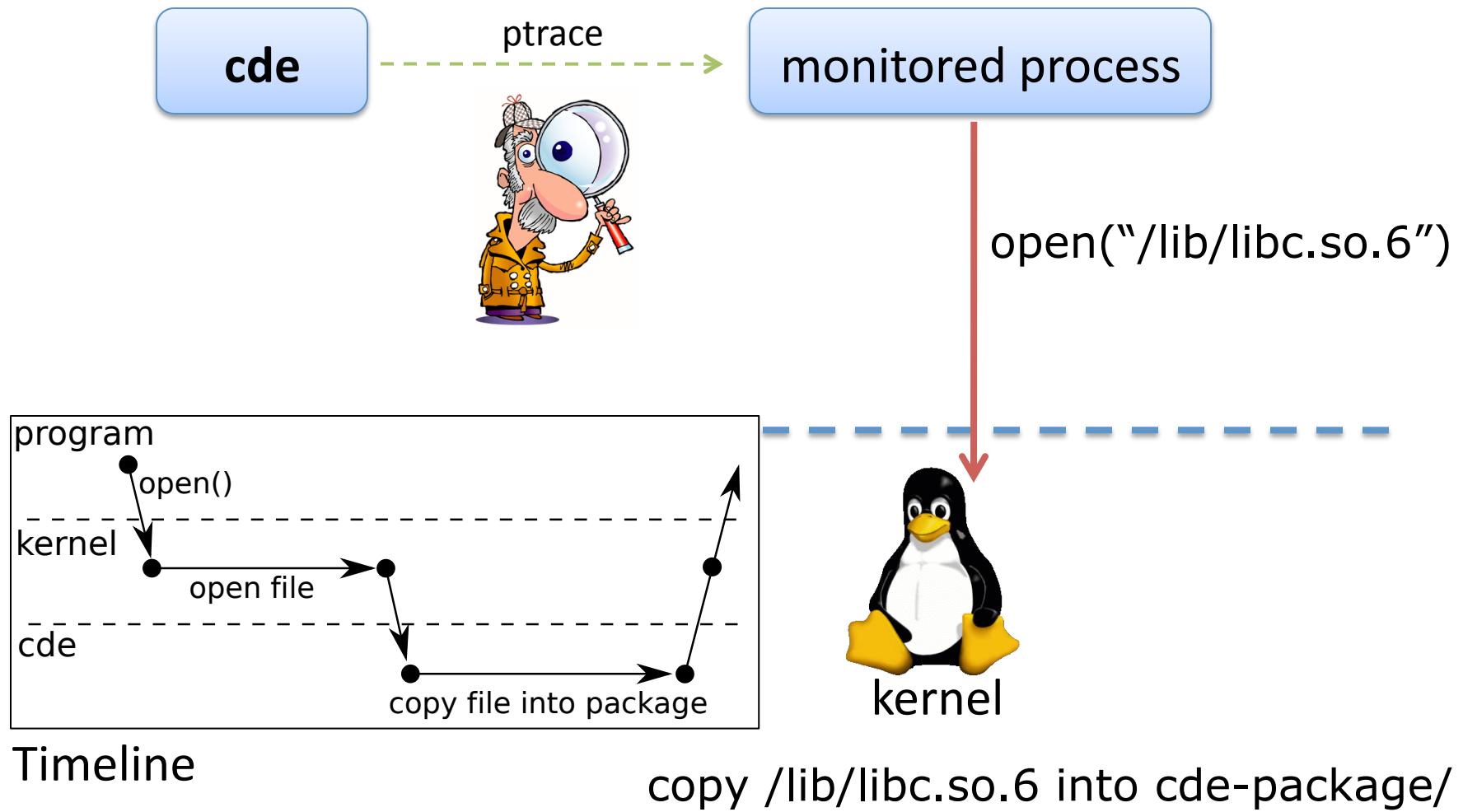
cde-exec python weather_sim.py tokyo.dat

Bob no longer
needs to bother
his university
sysadmins!

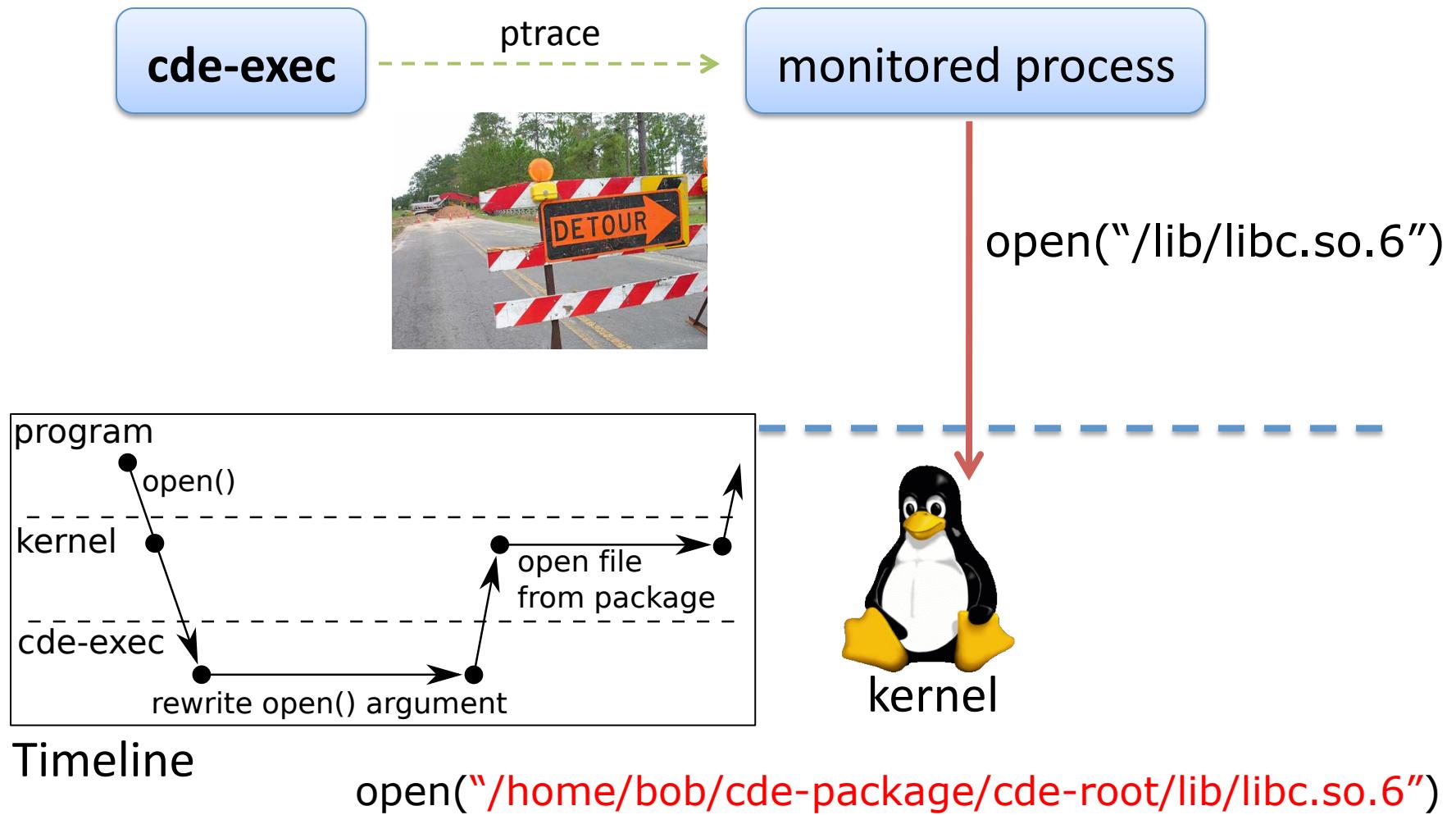


```
cde-exec python weather_sim.py tokyo.dat
```

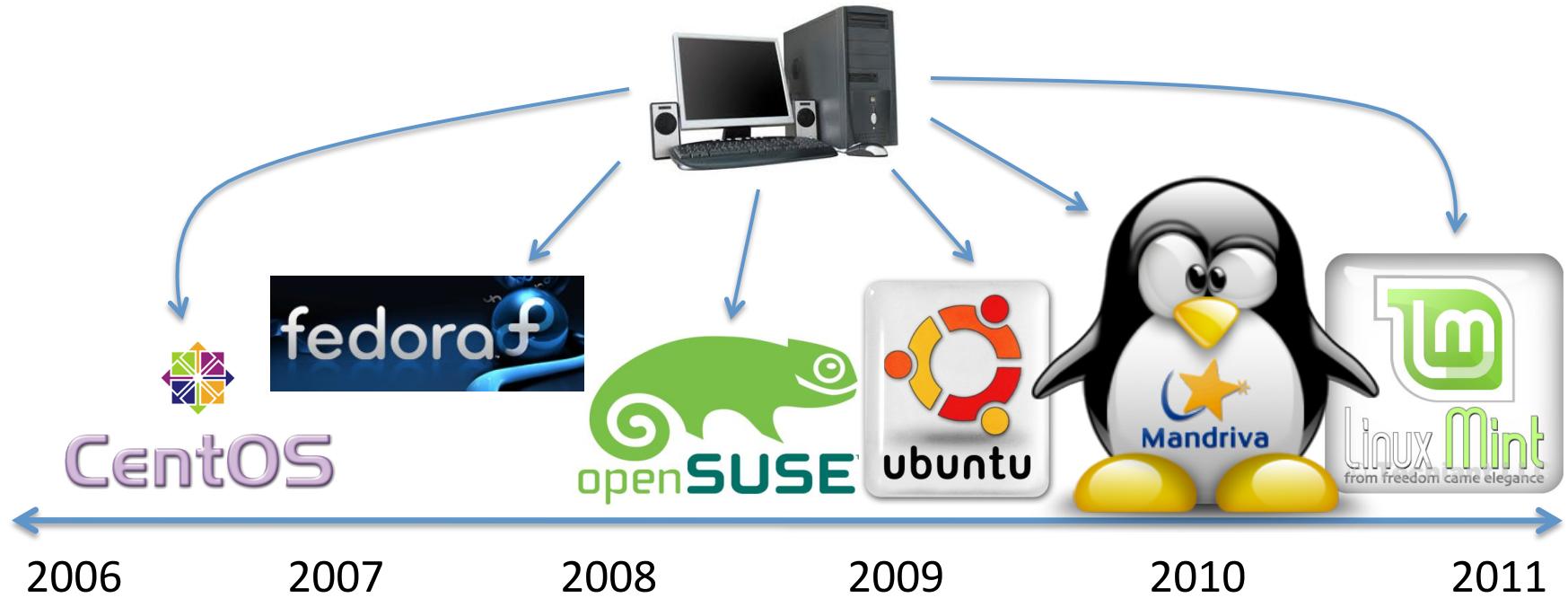
Creating a package with cde



Executing a package with cde-exec



Package portability



16 diverse CDE packages can execute on popular distros with **no installation or configuration**.

Requires user-kernel ABI to be compatible

Run-time slowdowns

Proportional to #syscalls / sec.

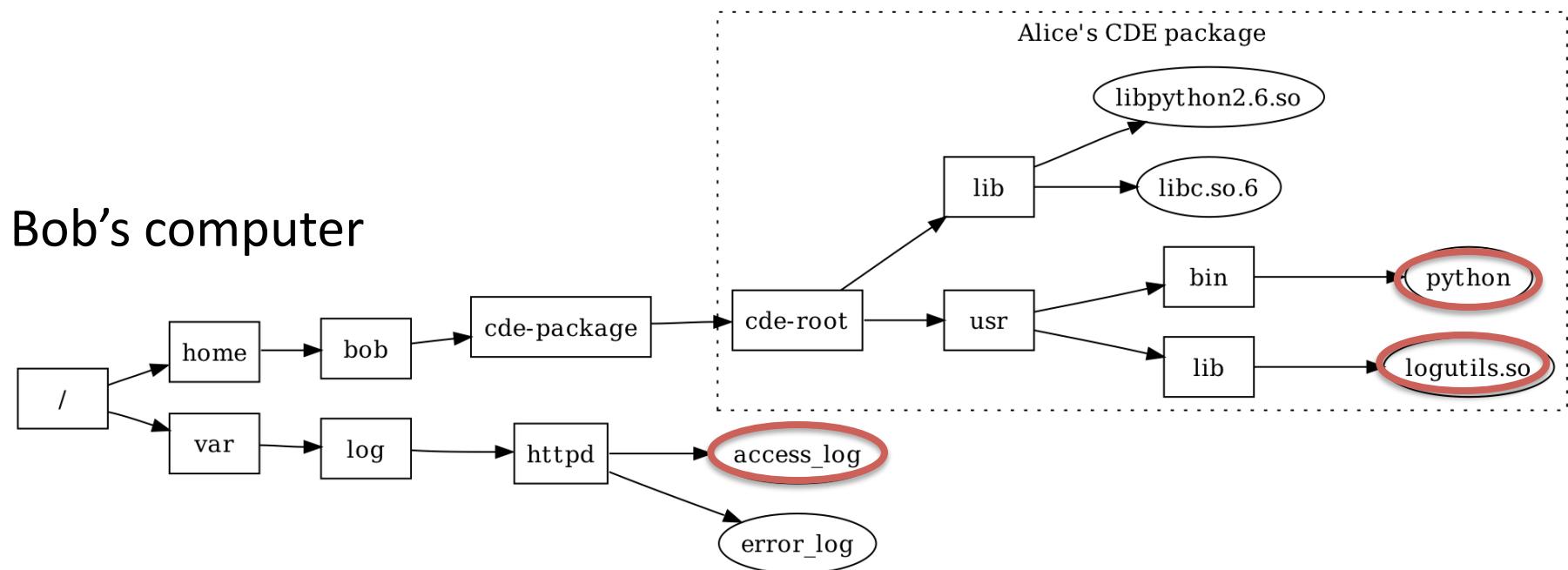
- SPEC CPU2006 benchmarks: 0% - 4%
- Batch file processing apps: 2% - 28%
- GUI apps: slower start-up

1. CDE overview

2. Seamless execution

3. Run any app on-demand

Seamless execution



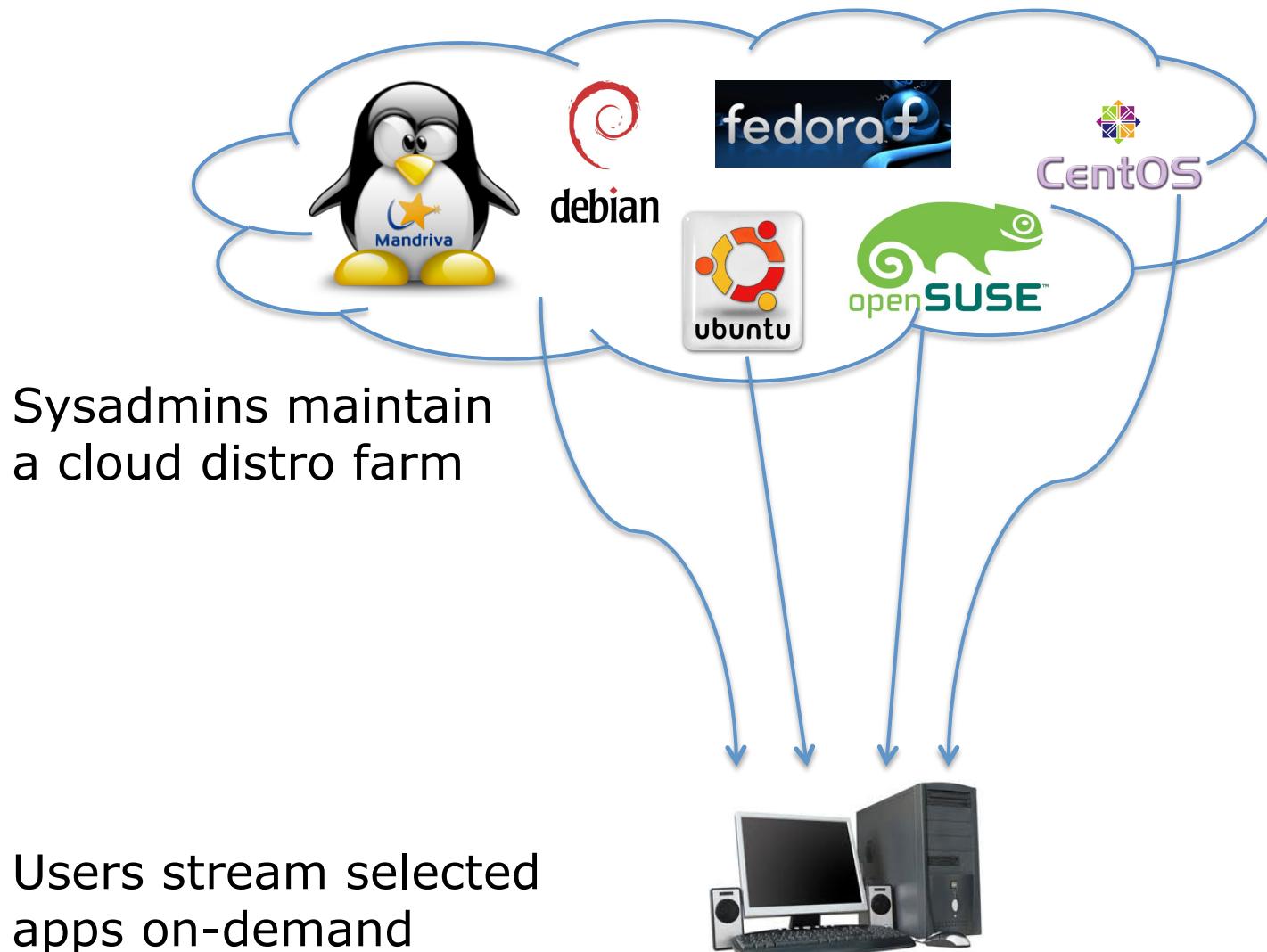
```
cde-exec python <Alice's scripts> /var/log/httpd/access_log
```

```
/home/bob/cde-package/cde-root/usr/bin/python
/home/bob/cde-package/cde-root/usr/lib/logutils.so
/home/bob/cde-package/cde-root/var/log/httpd/access_log
/var/log/httpd/access_log
```

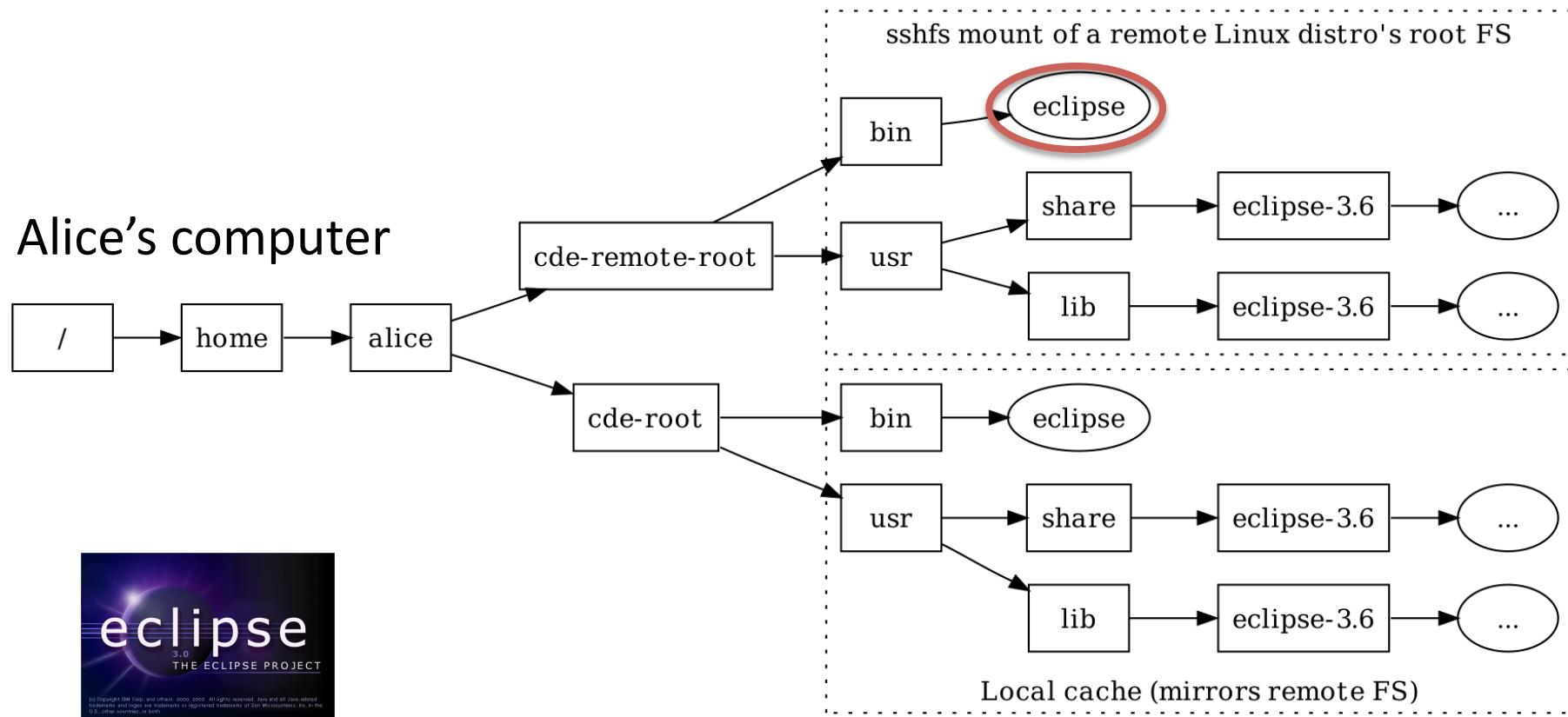


1. CDE overview
2. Seamless execution
3. Run any app on-demand

Run any app on-demand



CDE streaming mode

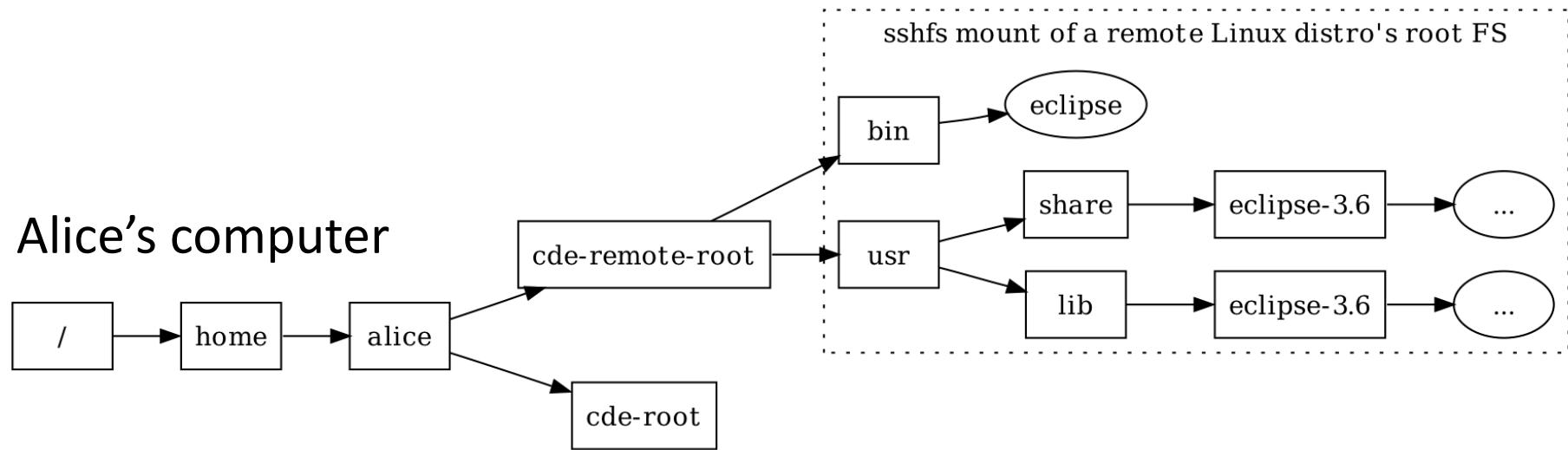


Alice wants to run Eclipse without installing it on her machine

`<mount cloud distro containing Eclipse>`

`cde-exec -s eclipse`

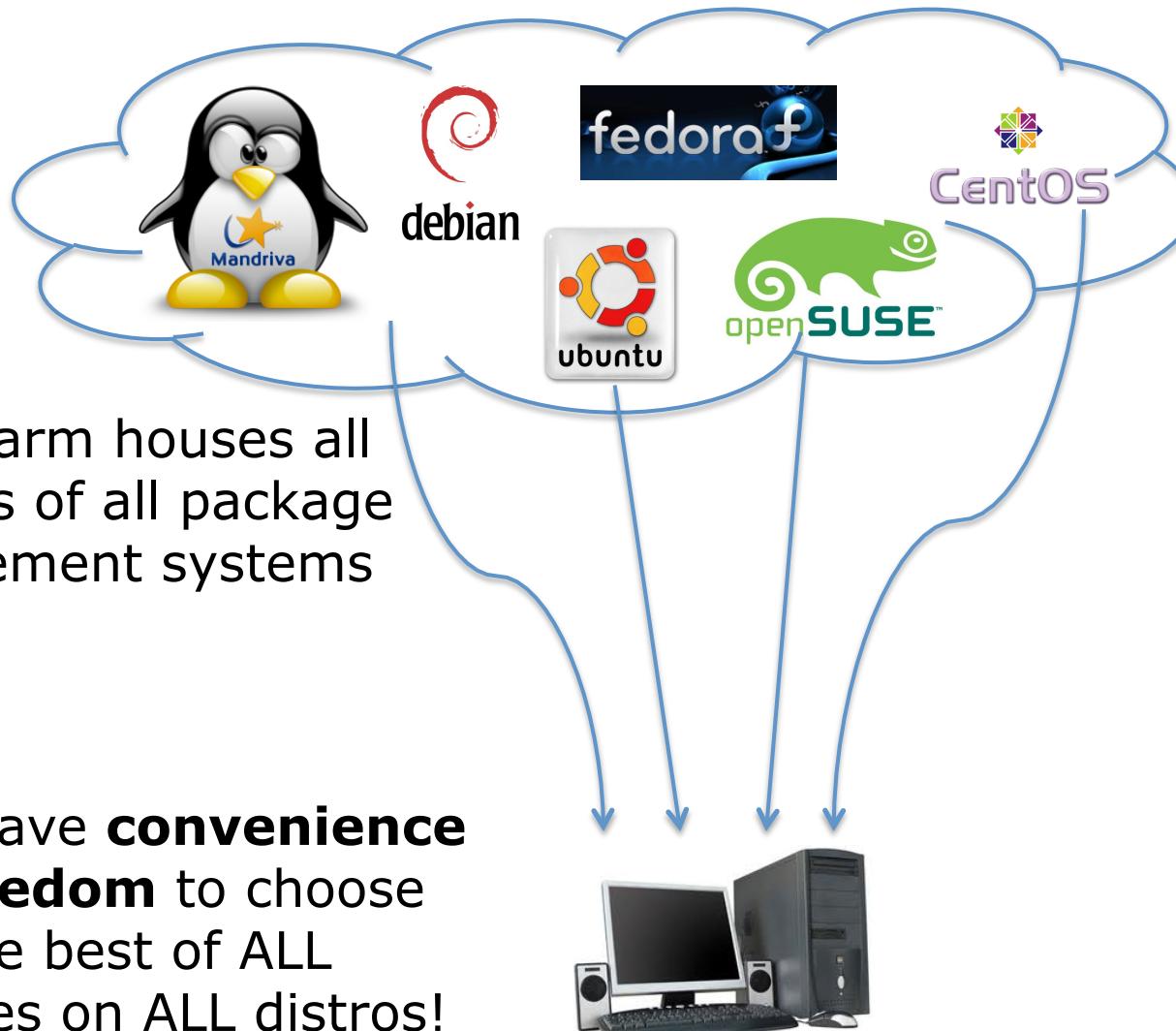
CDE streaming mode



Instead of delivering a package, have users connect to your server and stream your apps on-demand.

Solves the package incompleteness problem!

End Linux distro holy wars!



CDE: Automatic packaging of Code, Data, and Environment



~4000 downloads so far
(Google for “**cde linux**”)

Real-world use cases:

- Sharing prototype software
- Deploying custom web app stacks
- Reproducing research experiments
- Running new software on older distros
- Deploying computations to cluster/cloud