

RED HAT
SUMMIT

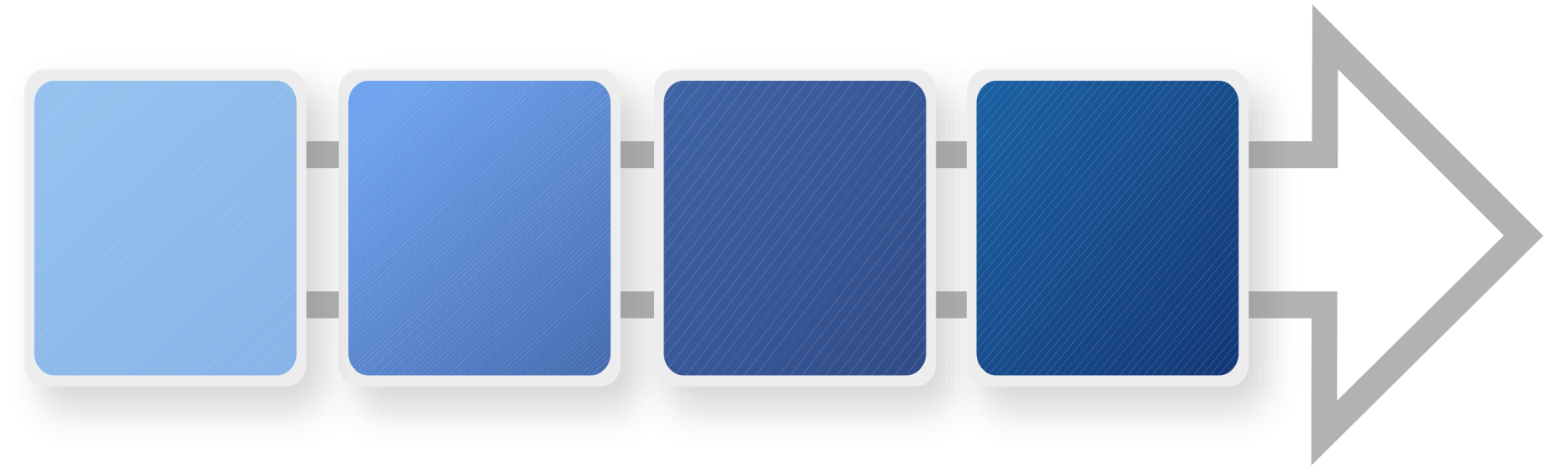
10 YEARS *and counting*
SAN FRANCISCO | APRIL 14-17, 2014

Software Collections:

Keeping Pace Without Sacrificing Platform Stability

Adam Miller
Senior Software Engineer, Red Hat Inc.

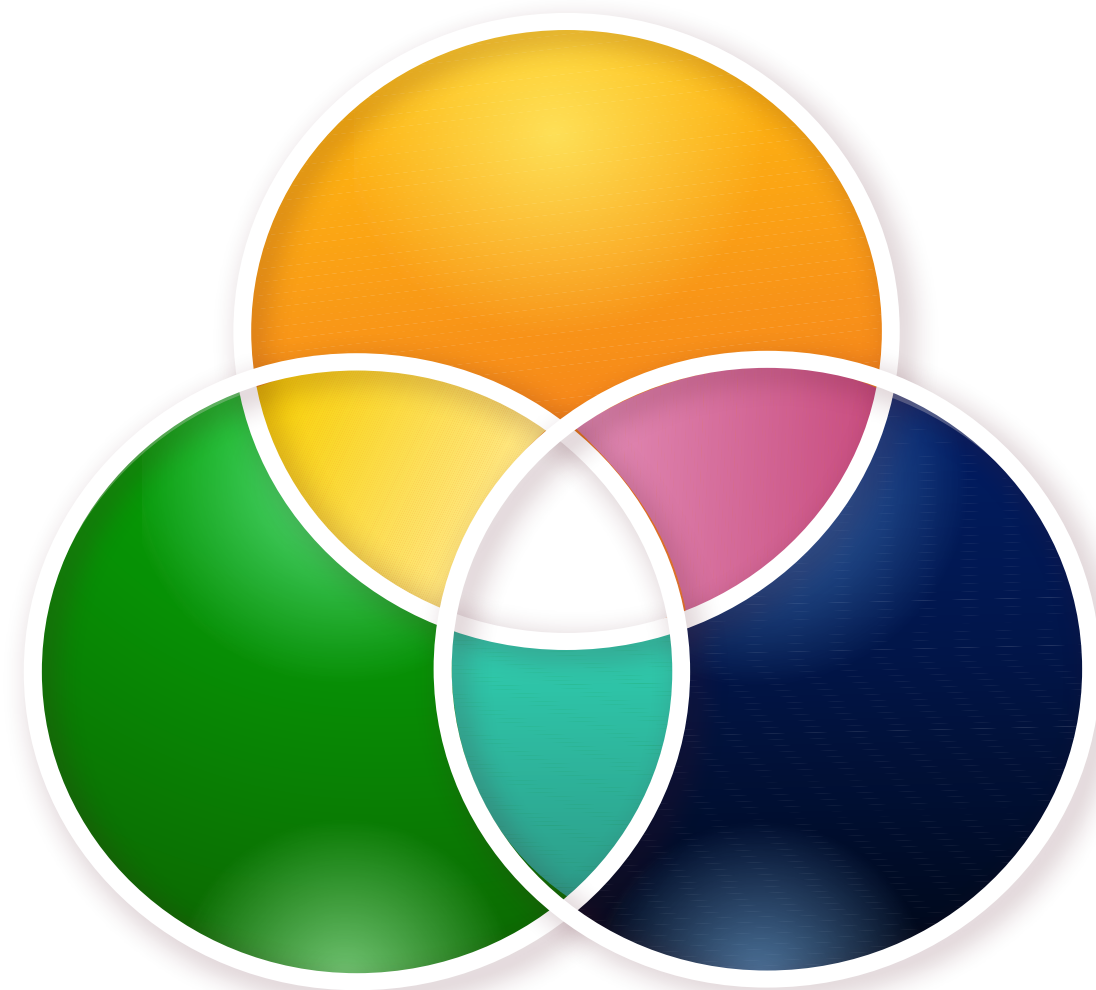
What we'll be talking about.



- What are Software Collections
- Reasons to (hopefully) care about Software Collections
- Problems Software Collections can solve for you and your team
- How Software Collections work
- Ways to utilize Software Collections
- Brief overview of building RPMs to extend Software Collections
- Real world examples of Software Collections in use today

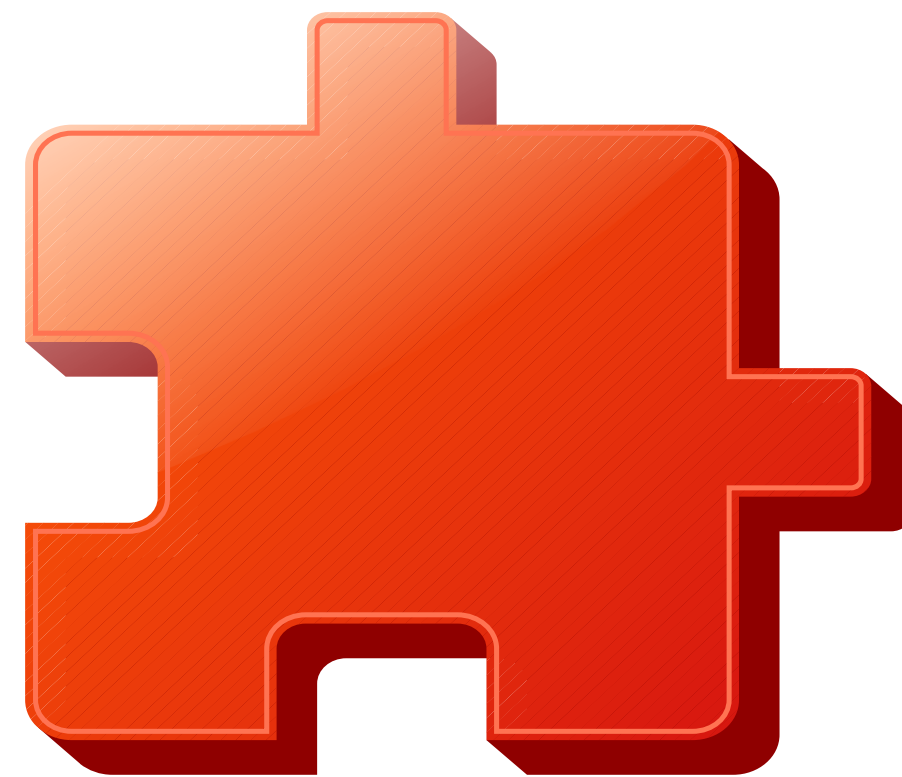
Problem(s)

- Users want “the new hotness” and want it at a rapid pace.
- Developers want to be able to work with the latest tools.
- Operations Teams require stability to maintain the business infrastructure.



Solution – Software Collections

- Delivering Software More Rapidly
 - Software Collections have a 3 year lifecycle per release
 - 18-month Cycle for Major Revisions
- Fully Supported product shipped with RHEL
 - All subscriptions higher tier than “Self-Support” receive access

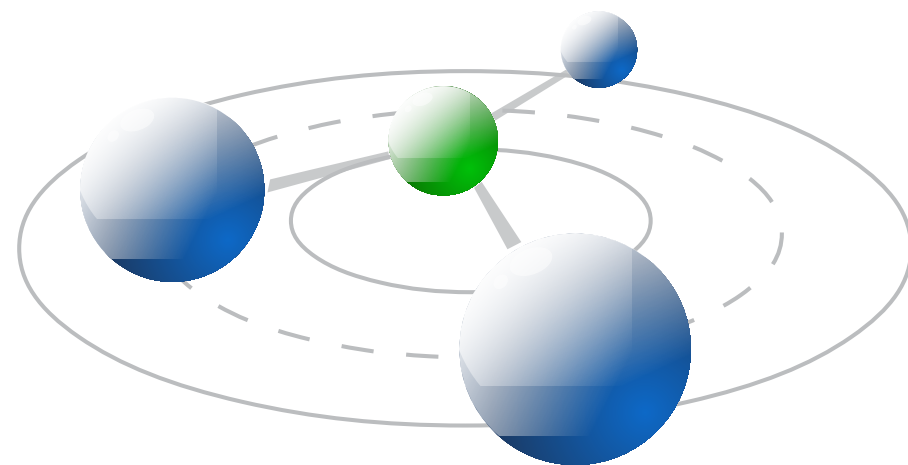


What are Software Collections?

- Vendored Runtime for Collections of Software in a “Namespace”
- Similar Tools specific to programming language:
 - VirtualEnv – Python
 - RVM – Ruby
 - NVM, nodeenv – Node.js
- Software Collections aren't limited to programming languages.
 - Service Daemons such as databases and web servers also possible.
 - Anything we can package in a RPM (via RPM Macros)

Reasons To (hopefully) Care

- Standard Ops/Admin toolset for deployment and management
 - Packaged as RPMs, Standard yum install and upgrades
 - Errata tooling and reporting is the same (`yum check-update --security`)
 - Allow scripts to run with unmodified shebang entries.
- Delivers newer technology more rapidly without sacrificing platform stability.
 - Aims to keep Ops, Dev, and Users happier. Working together.



How Do They Work?

- The `scl` command line utilities manage `$PATH`
 - Uses precedence in `$PATH` to allow for overriding system defaults
- Package RPMs with custom (“backwards” compatible) macros such as `%_scl_prefix /opt/provider`

Software Collections Filesystem Hierarchy

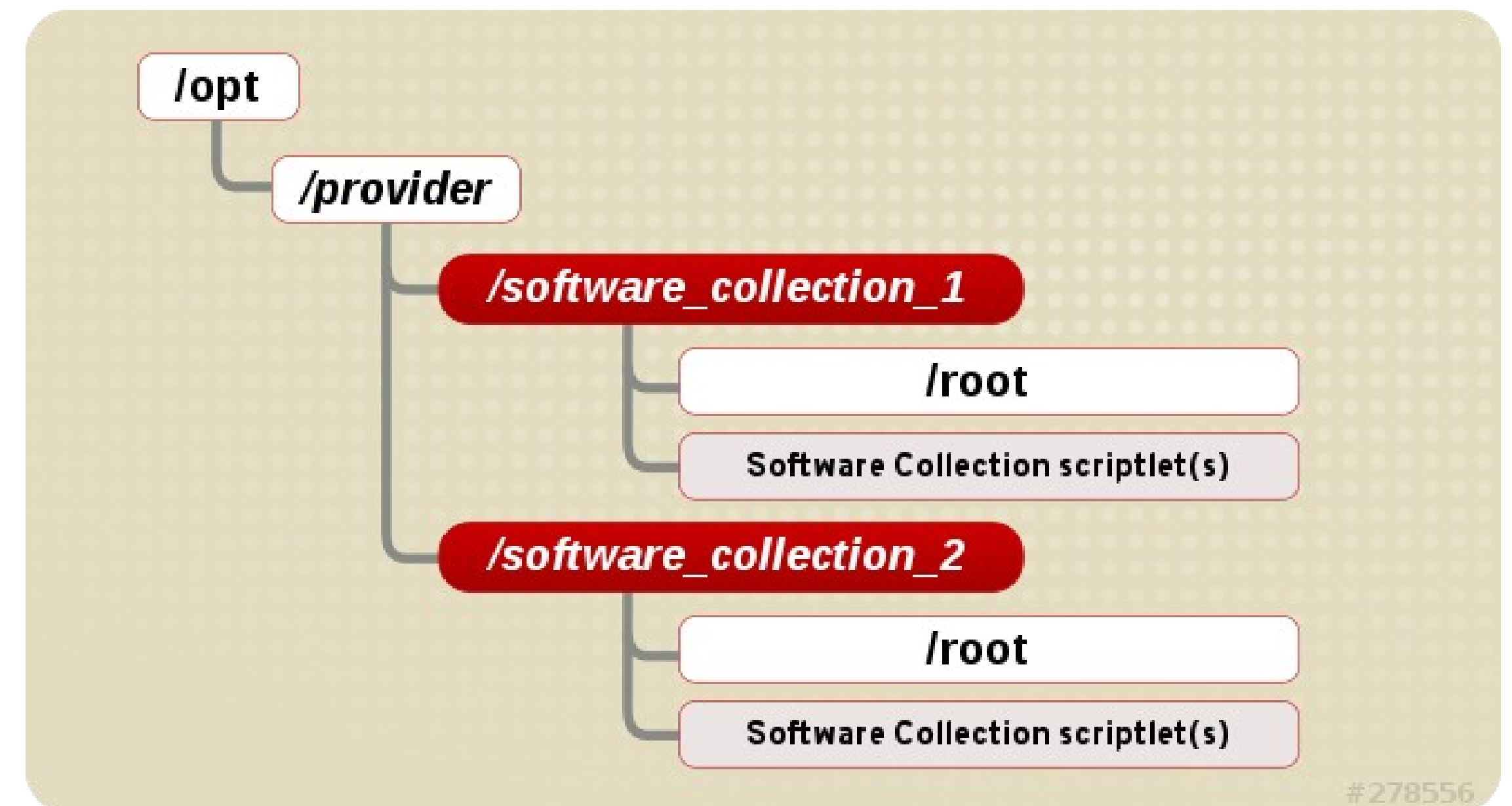


Image Credit: Fedora Documentation Team
http://docs.fedoraproject.org/en-US/Fedora_Contributor_Documentation/1/html/Software_Collections_Guide/sect-The_File_System_Hierarchy.html

RED HAT
SUMMIT

10 YEARS *and counting*

SAN FRANCISCO | APRIL 14-17, 2014


Using Software Collections

Installation and Methods of Utilizing

How To Install Software Collections

```
Terminal - root@rhel6:~
[root@rhel6 ~]# yum repolist all | grep scl
This system is receiving updates from Red Hat Subscription Management.
rhel-server-rhscl-6-beta-debug-rpms      Red Hat So disabled
rhel-server-rhscl-6-beta-rpms           Red Hat So disabled
rhel-server-rhscl-6-beta-source-rpms    Red Hat So disabled
rhel-server-rhscl-6-debug-rpms         Red Hat So disabled
rhel-server-rhscl-6-eus-debug-rpms     Red Hat So disabled
rhel-server-rhscl-6-eus-rpms           Red Hat So disabled
rhel-server-rhscl-6-eus-source-rpms    Red Hat So disabled
rhel-server-rhscl-6-rpms                Red Hat So enabled:      523
rhel-server-rhscl-6-source-rpms        Red Hat So disabled
[root@rhel6 ~]# yum-config-manager --enable rhel-server-rhscl-6-rpms &> /dev/nul
l
[root@rhel6 ~]# yum repolist enabled
Loaded plugins: product-id, security, subscription-manager
This system is receiving updates from Red Hat Subscription Management.
rhel-6-server-rpms                       | 3.7 kB    00:00
rhel-server-rhscl-6-rpms                  | 2.6 kB    00:00
repo id      repo name                                status
epel         Extra Packages for Enterprise Linux 6 - x86_64  10,654
rhel-6-server-rpms  Red Hat Enterprise Linux 6 Server (RPMs)      12,393
rhel-server-rhscl-6-rpms Red Hat Software Collections RPMs for Red Hat En  523
repolist: 23,570
[root@rhel6 ~]#
```

```
Terminal - root@rhel6:~
[root@rhel6 ~]# yum install postgresql92 mariadb55 ruby193 nodejs010 python33
```



NOTE: You can also specify specific packages such as `ruby193-ruby` to get the `ruby193` SCL version of `ruby`. (Format `$scl_prefix-$pkg_name`)

Software Collections Skeleton Root

- Each Software Collection has a skeleton root directory mapping to relative locations of files in /
 - Does not populate with unnecessary files, only lays down what is necessary but skeleton structure is needed for hierarchy.

```
Terminal - root@rhel6:~
[root@rhel6 ~]# tree /opt/rh/ruby193/ -L 2
/opt/rh/ruby193/
├── enable
└── root
    ├── bin
    ├── boot
    ├── dev
    ├── etc
    ├── home
    ├── lib
    ├── lib64
    ├── media
    ├── mnt
    ├── opt
    ├── proc
    ├── root
    ├── sbin
    ├── selinux
    ├── srv
    ├── sys
    ├── tmp
    ├── usr
    └── var

20 directories, 1 file
[root@rhel6 ~]#
```

Software Collections Scriptlets

- Software Collections have enable “scriptlets” that are used by the `scl` command line utilities
- Service Daemons also provide service-environment definitions for Software Collections to load

```
Terminal - root@rhel6:~
[root@rhel6 ~]# tree /opt/rh/postgresql92/ -L 2
/opt/rh/postgresql92/
├── enable
├── root
│   ├── bin
│   ├── boot
│   ├── dev
│   ├── etc
│   ├── home
│   ├── lib
│   ├── lib64
│   ├── media
│   ├── mnt
│   ├── opt
│   ├── proc
│   ├── root
│   ├── sbin
│   ├── selinux
│   ├── srv
│   ├── sys
│   ├── tmp
│   ├── usr
│   └── var
└── service-environment

20 directories, 2 files
[root@rhel6 ~]#
```


Running Single Commands with SCL

```
[root@rhel6 ~]# ruby --version  
ruby 1.8.7 (2011-06-30 patchlevel 352) [x86_64-linux]
```

```
[root@rhel6 ~]# scl enable ruby193 'ruby --version'  
ruby 1.9.3p448 (2013-06-27) [x86_64-linux]
```



Enable Multiple Software Collections

```
[root@rhel6 ~]# ruby --version; python --version  
ruby 1.8.7 (2011-06-30 patchlevel 352) [x86_64-linux]  
Python 2.6.6
```

```
[root@rhel6 ~]# scl enable ruby193 python33 bash  
[root@rhel6 ~]# ruby --version; python --version  
ruby 1.9.3p448 (2013-06-27) [x86_64-linux]  
Python 3.3.2
```



Enable Software Collections Without scl Utilities

```
[root@rhel6 ~]# ruby --version  
ruby 1.8.7 (2011-06-30 patchlevel 352) [x86_64-linux]
```

```
[root@rhel6 ~]# . /opt/rh/ruby193/enable  
[root@rhel6 ~]# ruby --version  
ruby 1.9.3p448 (2013-06-27) [x86_64-linux]
```

```
[root@rhel6 ~]# exec su - $USER  
[root@rhel6 ~]# ruby --version  
ruby 1.8.7 (2011-06-30 patchlevel 352) [x86_64-linux]
```



Setting Software Collections in ~/ .bashrc

- and other potentially silly nonsense.

```
[root@rhel6 ~]# tail -3 ~/.bashrc
if [ -f /opt/rh/ruby193/enable ]; then
. /opt/rh/ruby193/enable
fi

[root@rhel6 ~]# ruby --version
ruby 1.9.3p448 (2013-06-27) [x86_64-linux]
```

- **NOTE:** This is not a supported configuration and there are some risks, the `scl` utilities handle `$PATH` properly and set env variables such as `X_SCL`



Inspecting \$PATH

#NOTE: PATH broken up over multiple lines because LibreOffice
Impress doesn't line wrap gracefully.

```
[root@rhel6 ~]# scl enable ruby193 'echo $PATH'  
/opt/rh/ruby193/root/usr/bin:/usr/local/sbin:  
/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin:/root/bin
```

```
[root@rhel6 ~]# echo $PATH  
/usr/local/sbin:/usr/local/bin:/sbin:/bin:  
/usr/sbin:/usr/bin:/root/bin
```



Service Daemons and Software Collections

- Namespaced for consistency
 - Uses a similar method as shown for ~/ .bashrc

```
[root@rhel6 ~]# ls /etc/init.d/postgresql92-postgresql  
/etc/init.d/postgresql92-postgresql
```

```
[root@rhel6 ~]# service postgresql92-postgresql start  
Starting postgresql92-postgresql service:          [ OK ]
```



Software Collections Currently Available

- Red Hat Software Collections 1.0
 - Perl 5.16.3, PHP 5.4.14 , Python 2.7, Python 3.3, Ruby 1.9.3, MariaDB 5.5, MySQL 5.5, PostgreSQL 9.2, Node.js 0.10
- Red Hat Software Collections 1.1 (Beta at the time of this writing)
 - Includes all that RH SCL 1.0 included, plus bugfixes, security updates and the following additions or version revisions.
 - PHP 5.4.16, PHP 5.5.6, Ruby 2.0.0, Ruby On Rails 4.0.2, MongoDB 2.4.9, nginx 1.4.4, Apache httpd 2.4.6, Thermostat 1, V8 3.14
 - Adds a number of features and packages to the Software Collections shipped with 1.0

RED HAT
SUMMIT

10 YEARS *and counting*

SAN FRANCISCO | APRIL 14-17, 2014

Extend Software Collections

Use Software Collections with In-House or Unpackaged Code

Building RPMs



- Software Collections introduce a new set of RPM Macros
 - The Macros are backwards compatible, can produce both non-SCL and SCL builds
 - This is done by defines for the build environment
- Build with `rpmbuild`
 - `rpmbuild -ba -define "scl $name"`
- Mock toolchain works similarly
 - `mock -r $config -D "scl $name"`
 - <https://fedoraproject.org/wiki/Projects/Mock>

Extending Software Collections



- Packaging software that depends on a Collection is considered “extending”
 - It will exist inside the Namespace of the Software Collection
- Spec2SCL tool to assist with conversion of RPM spec files
 - Upstream - <https://bitbucket.org/bkabrda/spec2scl/>
 - `easy_install spec2scl`
 - From PyPi - <https://pypi.python.org/pypi/spec2scl>
 - `yum -y install spec2scl`
 - From EPEL - <https://fedoraproject.org/wiki/EPEL>

Example SCL-ized .spec File



```
%{?scl:%scl_package less}
%{!?scl:%global pkg_name %{name}}
Summary: A text file browser similar to more, but better
Name: %{?scl_prefix}less
Version: 458
Release: 7%{?dist}
License: GPLv3+
Group: Applications/Text
Source: http://www.greenwoodsoftware.com/less/%{pkg\_name}-%{version}.tar.gz
Source1: lesspipe.sh
Source2: less.sh
Source3: less.csh
```

Example SCL-ized .spec File – Continued



```
URL: http://www.greenwoodsoftware.com/less/
```

```
Requires: %{?scl_prefix}groff-base
```

```
BuildRequires: %{?scl_prefix}ncurses-devel
```

```
BuildRequires: %{?scl_prefix}autoconf %{?scl_prefix}automake %{?scl_prefix}libtool
```

```
%description
```

```
The less utility is a text file browser that resembles more #SNIP FOR BREVITY
```

```
%prep
```

```
%setup -n %{pkg_name}-%{version} -q
```

```
autoreconf
```

```
chmod -R a+w && chmod 644 *.c *.h LICENSE README
```

Example SCL-ized .spec File – Continued



```
%build
```

```
%{?scl:scl enable %scl - << \EOF}
```

```
%configure
```

```
make CC="gcc $RPM_OPT_FLAGS -D_GNU_SOURCE -D_LARGEFILE_SOURCE -D_LARGEFILE64_SOURCE  
-D_FILE_OFFSET_BITS=64" datadir=%{_docdir}
```

```
%{?scl:EOF}
```


Example SCL-ized .spec File - Continued



```
%install
%{?scl:scl enable %scl - << \EOF}
make DESTDIR=$RPM_BUILD_ROOT install
%{?scl:EOF}
mkdir -p $RPM_BUILD_ROOT%{_sysconfdir}/profile.d
install -p -c -m 755 %{SOURCE1} $RPM_BUILD_ROOT/%{_bindir}
install -p -c -m 644 %{SOURCE2} $RPM_BUILD_ROOT%{_sysconfdir}/profile.d
install -p -c -m 644 %{SOURCE3} $RPM_BUILD_ROOT%{_sysconfdir}/profile.d
ls -la $RPM_BUILD_ROOT%{_sysconfdir}/profile.d
```

Example SCL-ized .spec File – Continued



```
%files
```

```
%doc LICENSE
```

```
%{_sysconfdir}/profile.d/*
```

```
%{_bindir}/*
```

```
%{_mandir}/man1/*
```

RED HAT
SUMMIT

10 YEARS *and counting*

SAN FRANCISCO | APRIL 14-17, 2014

Real World Use Cases

Examples of Software Collections in Production

Red Hat OpenStack Platform

- Components within the Red Hat OpenStack platform rely upon Red Hat Software Collections are used to power automation and monitoring.
 - <https://www.redhat.com/openstack/>
 - <https://access.redhat.com/site/products/Cloud/OpenStack/>



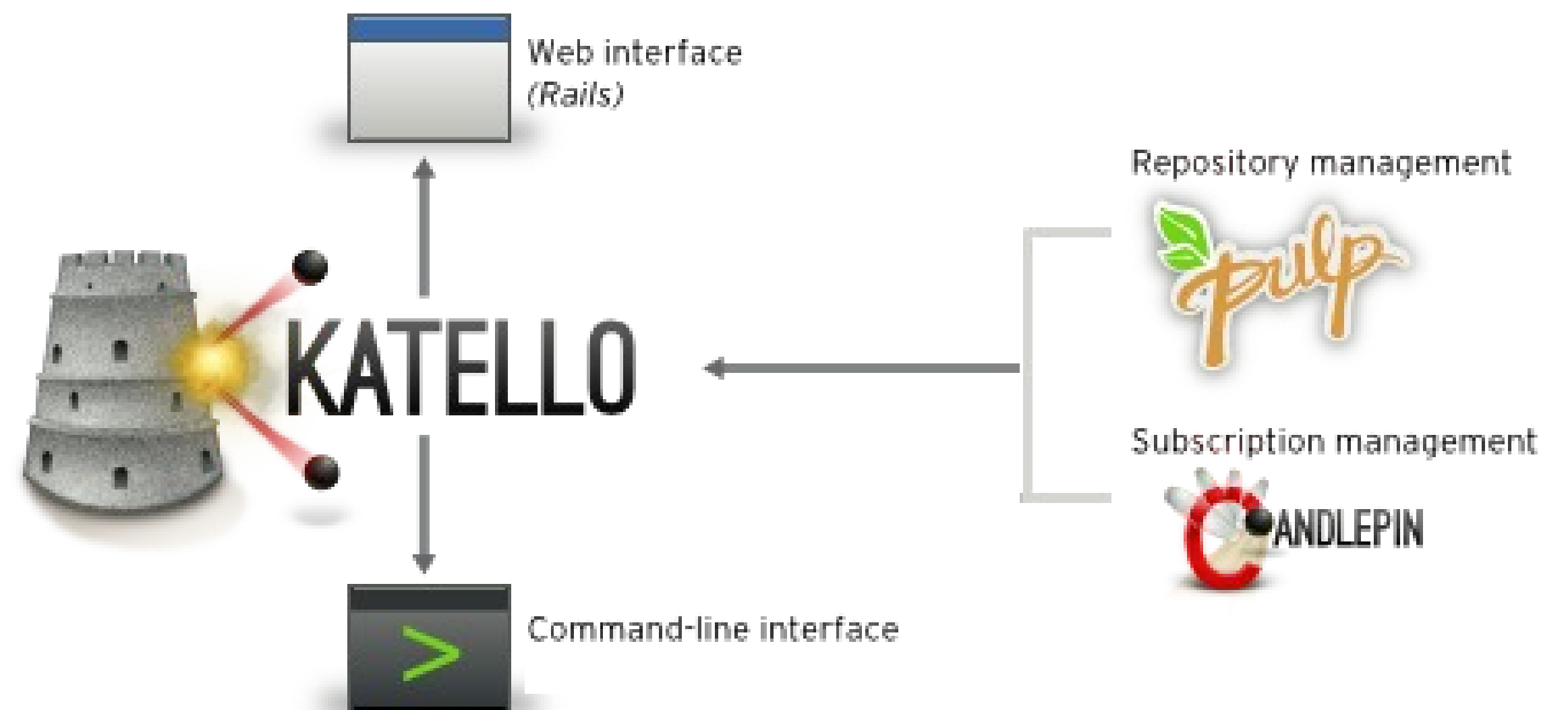
The Foreman

- Foreman is an open source project that helps system administrators manage servers throughout their lifecycle, from provisioning and configuration to orchestration and monitoring.
 - <http://theforeman.org>



Katello

- Katello is a software and systems management software suite integrating with Pulp, Candlepin, and The Foreman.
- <http://www.katello.org/>
 - <http://www.pulpproject.org/>
 - <http://candlepinproject.org/>
 - <http://theforeman.org/>



OpenShift (Origin, Online, Enterprise)

- OpenShift is the next generation Open Source Platform-as-a-Service developed by Red Hat. OpenShift handles infrastructure, middleware, and management so that you can focus on your app.
- OpenShift Online - (<https://www.openshift.com/>)
 - Over 1 Million Applications deployed by customers in Production
 - <http://www.eweek.com/developer/red-hat-delivers-openshift-online-paas-expands-ecosystem/>
- OpenShift Enterprise - (<https://www.openshift.com/products/enterprise>)
 - Large Deployments Private On-Premise PaaS
- OpenShift Origin - (<http://openshift.github.io/>)
 - Upstream Project



RED HAT
SUMMIT

10 YEARS *and counting*

SAN FRANCISCO | APRIL 14-17, 2014

Questions? Comments?

Snide Remarks? All Welcome!

References

- Official Red Hat Documentation – Software Collections
 - https://access.redhat.com/site/documentation/en-US/Red_Hat_Software_Collections/
- Official Red Hat Documentation – Developer Toolset
 - https://access.redhat.com/site/documentation/en-US/Red_Hat_Developer_Toolset/
- Software Collections Upstream
 - <https://softwarecollections.org/en/docs/>
- Fedora Community Documentation
 - http://docs.fedoraproject.org/en-US/Fedora_Contributor_Documentation/1/html/Software_Collections_Guide/index.html

RED HAT
SUMMIT

10 YEARS *and counting*

SAN FRANCISCO | APRIL 14-17, 2014

Thank You

Adam Miller
@TheMaxamillion
maxamillion on irc.freenode.net