A Perspective for systemd
What Has Been Achieved, and What Lies Ahead

GNOME.asia/FUDCON, Beijing, China

May 2014

What's systemd again?

What's systemd again?

A system and service manager

What's systemd again?

A system and service manager

A platform

What's systemd again?

A system and service manager

A platform

The glue between the applications and the kernel

Where are we now?

Where are we now?

systemd universally adopted: Fedora, RHEL 7, Mandriva, Suse, Debian, Ubuntu, . . .)

Where are we now?

systemd universally adopted: Fedora, RHEL 7, Mandriva, Suse, Debian, Ubuntu, . . .)

24 committers (Red Hat, Intel, Debian, Mandriva, Canonical, Google, Pantheon, . . .)

Where are we now?

systemd universally adopted: Fedora, RHEL 7, Mandriva, Suse, Debian, Ubuntu, . . .)

24 committers (Red Hat, Intel, Debian, Mandriva, Canonical, Google, Pantheon, . . .)

501 contributors in total

Where are we now?

systemd universally adopted: Fedora, RHEL 7, Mandriva, Suse, Debian, Ubuntu, . . .)

24 committers (Red Hat, Intel, Debian, Mandriva, Canonical, Google, Pantheon, . . .)

501 contributors in total

35-40 contributors per month

Where are we now?

systemd universally adopted: Fedora, RHEL 7, Mandriva, Suse, Debian, Ubuntu, . . .)

24 committers (Red Hat, Intel, Debian, Mandriva, Canonical, Google, Pantheon, . . .)

501 contributors in total

35-40 contributors per month

15K commits in total

Where are we now?

systemd universally adopted: Fedora, RHEL 7, Mandriva, Suse, Debian, Ubuntu, . . .)

24 committers (Red Hat, Intel, Debian, Mandriva, Canonical, Google, Pantheon, . . .)

501 contributors in total

35-40 contributors per month

15K commits in total

300 commits per month

Where are we now?

systemd universally adopted: Fedora, RHEL 7, Mandriva, Suse, Debian, Ubuntu, . . .)

24 committers (Red Hat, Intel, Debian, Mandriva, Canonical, Google, Pantheon, . . .)

501 contributors in total

35-40 contributors per month

15K commits in total

300 commits per month

830 subscribers to the mailing list

Where are we now?

systemd universally adopted: Fedora, RHEL 7, Mandriva, Suse, Debian, Ubuntu, . . .)

24 committers (Red Hat, Intel, Debian, Mandriva, Canonical, Google, Pantheon, . . .)

501 contributors in total

35-40 contributors per month

15K commits in total

300 commits per month

830 subscribers to the mailing list

= A healthy Open Source project

Our objectives

Our objectives

Turning Linux from a *bag of bits* into a competitive *General Purpose Operating System*.

Our objectives

Turning Linux from a *bag of bits* into a competitive *General Purpose Operating System*.

Building the Internet's Next Generation OS

Our objectives

Turning Linux from a *bag of bits* into a competitive *General Purpose Operating System*.

Building the Internet's Next Generation OS

Unifying pointless differences between distributions

Our objectives

Turning Linux from a *bag of bits* into a competitive *General Purpose Operating System*.

Building the Internet's Next Generation OS

Unifying pointless differences between distributions

Bringing innovation back to the core OS

Our objectives

Our objectives

Desktop,

Our objectives

Desktop, Server,

Our objectives

Desktop, Server, Container,

Our objectives

Desktop, Server, Container, Embedded,

Our objectives

Desktop, Server, Container, Embedded, Mobile,

Our objectives

Desktop, Server, Container, Embedded, Mobile, Cloud,

Our objectives

Desktop, Server, Container, Embedded, Mobile, Cloud, Cluster, ...

Our objectives

Desktop, Server, Container, Embedded, Mobile, Cloud, Cluster, ...

These areas are closer together than you might think

Our objectives

Desktop, Server, Container, Embedded, Mobile, Cloud, Cluster, ...

These areas are closer together than you might think

Reducing administrator complexity, reliability without supervision

Our objectives

Desktop, Server, Container, Embedded, Mobile, Cloud, Cluster, ...

These areas are closer together than you might think

Reducing administrator complexity, reliability without supervision

Everything introspectable

Our objectives

Desktop, Server, Container, Embedded, Mobile, Cloud, Cluster, ...

These areas are closer together than you might think

Reducing administrator complexity, reliability without supervision

Everything introspectable

Auto discovery, plug and play is key

Our objectives

Desktop, Server, Container, Embedded, Mobile, Cloud, Cluster, ...

These areas are closer together than you might think

Reducing administrator complexity, reliability without supervision

Everything introspectable

Auto discovery, plug and play is key

We fix things where they are broken, never tape over them

What is not our objective?

Never a product,

What is not our objective?

Never a product, but something that makes it easy to build products on

What is not our objective?

Never a product, but something that makes it easy to build products on

Never UI,

What is not our objective?

Never a product, but something that makes it easy to build products on

Never UI, but what you can build your UI on...

What is not our objective?

Never a product, but something that makes it easy to build products on

Never UI, but what you can build your UI on...

Never finished, never complete,

What is not our objective?

Never a product, but something that makes it easy to build products on

Never UI, but what you can build your UI on...

Never finished, never complete, but tracking progress of technology

What is not our objective?

Never a product, but something that makes it easy to build products on

Never UI, but what you can build your UI on...

Never finished, never complete, but tracking progress of technology

Never specific,

What is not our objective?

Never a product, but something that makes it easy to build products on

Never UI, but what you can build your UI on...

Never finished, never complete, but tracking progress of technology

Never specific, always generic

What is not our objective?

Never a product, but something that makes it easy to build products on

Never UI, but what you can build your UI on...

Never finished, never complete, but tracking progress of technology

Never specific, always generic

Never the cathedral,

What is not our objective?

Never a product, but something that makes it easy to build products on

Never UI, but what you can build your UI on...

Never finished, never complete, but tracking progress of technology

Never specific, always generic

Never the cathedral, just the building blocks to build it

We care for privacy

Encryption,

We care for privacy

Encryption, anonymity,

We care for privacy

Encryption, anonymity, security,

We care for privacy

Encryption, anonymity, security, verifiability,

We care for privacy

Encryption, anonymity, security, verifiability, trustability

What we already cover:

init system,

What we already cover:

init system, journal logging,

What we already cover:

init system, journal logging, login management,

What we already cover:

init system, journal logging, login management, device management,

What we already cover:

init system, journal logging, login management, device management, temporary and volatile file management,

What we already cover:

init system, journal logging, login management, device management, temporary and volatile file management, binary format registration,

What we already cover:

init system, journal logging, login management, device management, temporary and volatile file management, binary format registration, backlight save/restore,

What we already cover:

init system, journal logging, login management, device management, temporary and volatile file management, binary format registration, backlight save/restore, rfkill save/restore,

What we already cover:

init system, journal logging, login management, device management, temporary and volatile file management, binary format registration, backlight save/restore, rfkill save/restore, bootchart,

What we already cover:

init system, journal logging, login management, device management, temporary and volatile file management, binary format registration, backlight save/restore, rfkill save/restore, bootchart, readahead,

What we already cover:

init system, journal logging, login management, device management, temporary and volatile file management, binary format registration, backlight save/restore, rfkill save/restore, bootchart, readahead, encrypted storage setup,

What we already cover:

init system, journal logging, login management, device management, temporary and volatile file management, binary format registration, backlight save/restore, rfkill save/restore, bootchart, readahead, encrypted storage setup, EFI/GPT partition discovery,

What we already cover:

init system, journal logging, login management, device management, temporary and volatile file management, binary format registration, backlight save/restore, rfkill save/restore, bootchart, readahead, encrypted storage setup, EFI/GPT partition discovery, virtual machine/container registration,

What we already cover:

init system, journal logging, login management, device management, temporary and volatile file management, binary format registration, backlight save/restore, rfkill save/restore, bootchart, readahead, encrypted storage setup, EFI/GPT partition discovery, virtual machine/container registration, minimal container management,

What we already cover:

init system, journal logging, login management, device management, temporary and volatile file management, binary format registration, backlight save/restore, rfkill save/restore, bootchart, readahead, encrypted storage setup, EFI/GPT partition discovery, virtual machine/container registration, minimal container management, hostname management,

What we already cover:

init system, journal logging, login management, device management, temporary and volatile file management, binary format registration, backlight save/restore, rfkill save/restore, bootchart, readahead, encrypted storage setup, EFI/GPT partition discovery, virtual machine/container registration, minimal container management, hostname management, locale management,

What we already cover:

init system, journal logging, login management, device management, temporary and volatile file management, binary format registration, backlight save/restore, rfkill save/restore, bootchart, readahead, encrypted storage setup, EFI/GPT partition discovery, virtual machine/container registration, minimal container management, hostname management, locale management, time management,

What we already cover:

init system, journal logging, login management, device management, temporary and volatile file management, binary format registration, backlight save/restore, rfkill save/restore, bootchart, readahead, encrypted storage setup, EFI/GPT partition discovery, virtual machine/container registration, minimal container management, hostname management, locale management, time management, random seed management,

What we already cover:

init system, journal logging, login management, device management, temporary and volatile file management, binary format registration, backlight save/restore, rfkill save/restore, bootchart, readahead, encrypted storage setup, EFI/GPT partition discovery, virtual machine/container registration, minimal container management, hostname management, locale management, time management, random seed management, sysctl variable management,

What we already cover:

init system, journal logging, login management, device management, temporary and volatile file management, binary format registration, backlight save/restore, rfkill save/restore, bootchart, readahead, encrypted storage setup, EFI/GPT partition discovery, virtual machine/container registration, minimal container management, hostname management, locale management, time management, random seed management, sysctl variable management, console managment, ...

What we are working on:

network management

What we are working on:

network management

systemd-networkd

What we are working on:

Local DNS cache, mDNS responder, LLMNR responder, DNSSEC verification

What we are working on:

IPC in the kernel

What we are working on:

IPC in the kernel

kdbus, sd-bus

What we are working on:

Time synchronisation with NTP

What we are working on:

Time synchronisation with NTP

systemd-timesyncd

What we are working on:

More integration with containers

What we are working on:

More integration with containers

systemctl -M, journalctl -M, loginctl -M, systemd-run -M, . . .

systemctl -r, . . .

What we are working on:

Sandboxing of Services

Sandboxing of Apps

What we are working on:

OS Image format

Container image format

App image format

What we are working on:

OS Image format

Container image format

App image format

= GPT with auto-discovery

What we are working on:

Stateless systems, instantiatable systems, factory reset

What we are working on:

Stateless systems, instantiatable systems, factory reset

/usr is the OS

/etc is (optional) configuration

/var is (optional) state

What we are working on:

Atomic node initialisation and updates

What we are working on:

Integration with the cloud

Service management across nodes

What we are working on:

Verifiable OS images

All the way to the firmware

Boot Loading

That's all, folks!