Alternatives to systemd

This page describes the various init systems which are available as alternatives to systemd

Init is the first process started during system boot. It is a daemon process that continues running until the system is shut down. Init is the direct or indirect ancestor of all other processes, and automatically adopts all orphaned processes. It is started by the kernel using a hard-coded filename; if the kernel is unable to start it, panic will result. Init is typically assigned process identifier 1.

The init *scripts* (aka *rc* runtime configuration scripts) are launched by the init process to guarantee basic functionality on system start and shutdown. This includes (un)mounting of file systems and launching of daemons. A *service manager* takes this one step further by providing active control over launched processes, or process supervision. An example is to monitor for crashes and restart processes accordingly.

These components combine to the init *system*. Some init systems incorporate the service manager in the init process or have init scripts in close relation to them. Below, such init systems are referred to as *integrated*, although entries in different categories may explicitly depend on each other.

A nice (but still non-comprehensive) overview of init systems can be found in this blog entry, titled “A history of modern init systems (1992-2015)“

init systems

System	Description	Latest release
BusyBox init	quite small init without runlevels, solely signal driven IPC (no fifos, sockets, SysV IPC)	2019-10-25
dinit	dependency based C++ init with process supervision, roll-back, and socket activation	2020-01-01
finit	fast, event based, modular, extensible (C hooks/plugins) init with SysV runlevels, proper daemon supervision and logging, and optional builtin getty and inetd	2018-01-23
nosh	exec chaining, dependency based daemontools family C++ init and supervision suite with reliable logging, (console) virtual terminal management, systemd unit file compatibility; can also be used as service manager under a different init	2019-03-17
openrc-init	init system of OpenRC, shipped since version 0.25	2019-08-20
pies	dependency based GNU init daemon and super server with SysV runlevels, daemon supervision and logging, inetd functionality, socket activation, fine grained per service access control, understands SysV inittab, inetd.conf, and MeTA1 config files	2019-07-02
procd (2)	OpenWrt init and process/service management daemon with ubus integration	2018-07-30
runit-init	init from runit	2014-08-10
sinit	Simple init initially based on Rich Felker’s minimal init	2018-03-26
sninit	Small init implementation with SysV init like (sub)runlevels	2017-12-29
SysV init (2)	Traditional System V init. New release: v2.96 on July 07 2019	2019-09-11

System	Description	Latest release
ToyBox init (2)	similar to/(almost) compatible with BusyBox init (same config syntax/file)	2019-10-28
ToyBox oneit (2)	very simple init launcher (just (re)spawns a single child process and reacts to incoming signals)	2019-10-28
ueld	simple configuration, solely signal driven (like BSD and Busy/ToyBox init)	2018-07-09
uinit	Smallest init possible	2017-05-16

process supervisors (overview) and service managers

service manager: a **suite of programs** which start and stop services, both long-running daemons and one-time initialization scripts, in the proper order according to a dependency tree

Program	Description	Latest release
66	a collection of system tools built around s6 and s6-rc created to make the implementation and manipulation of service files easier	2019-12-16
Circus (docs)	runs, controls and monitors processes and sockets; written in Python; uses ZeroMQ	2019-12-27
daemontools encore	an enhanced version of daemontools (github repository)	2018-10-14
GNU Shepherd	service manager and init daemon written in Guile (formerly known as GNU dmd)	2019-05-11
MiniBase	small userspace utilities for Linux; includes a process supervisor that can be used as stage 2 of process #1 via exec chaining akin to s6-svscan	2019-11-14
OpenRC (2, 3)	dependency based service management suite; includes a process supervisor and a simple init alternative	2018-08-06
runit	process supervision suite in the line of daemontools; includes a simple init (2) replacement	2014-08-10
s6	small process supervision suite for UNIX ala daemontools, runit, and perp; can be used as stage 2 of process #1 via exec chaining	2019-10-21
s6-rc	service management suite for s6	2019-21-21
Supervisor	process control daemon written in Python	2018-02-15
ssm	simple service manager bash scripts for Linux	2018-03-06
supervise-daemon from OpenRC	superise-daemon introduced in OpenRC .0.21	2019-08-20
Supervision Framework	OpenRC-like service management suite for daemontools(-encore), runit, and s6	2018-09-01
systemctl-replacement	drop-in reading systemd service descriptions, with docker-oriented init function, written in Python	2019-10-24

udev alternatives for automatic Linux device file creation and management

udev (userspace /dev) is a device manager for the Linux kernel.

In April 2012, udev's source code was merged into the systemd source tree.[1]

Although udev can still be compiled for usage without systemd, Lennart Poettering said that they will not polish udevd outside of systemd[2], adding:

"Yes, udev on non-systemd systems is in our eyes a dead end, in case you haven't noticed it yet. I am looking forward to the day when we can drop that support entirely." So to be on the safe side, you probably want to use an alternative to udev:

DevMan	Description	Last release
eudev (2, 3)	Gentoo's maintained fork of udev	2017-11-22
mdev	the plug-and-play manager built into BusyBox	2018-07-31
mdevd (github)	nonforking Linux kernel uevent handler daemon; uses the mdev config file format	2018-08-14
nldev (2)	Suckless netlink frontend for mdev; replacements for udevd and udevadm (see also nlmon)	2018-08-03
smdev	mostly mdev-compatible Suckless program to manage device nodes	2015-04-12
vdev (github)	portable UNIX userspace device-file manager	2016-08-08

For more see this StackExchange question.

Related helper tool: udevil (2) (un)mounts removable devices without a password, shows device info, and monitors device changes. It can also mount ISO files, nfs://, smb://, ftp:, ssh:// and WebDAV URLs, and tmpfs/ramfs filesystems. Uses Glib and libudev (i. e. requires (e)udev) and can work completely without systemd, consolekit, policykit, dbus, udisks, gvfs & fuse (although it can coexist with any of these).

abandoned/defunct/discontinued/dormant/halted projects

Listed here

See also

- Gentoo wiki page: Comparison of init systems **:Talk** presents a more extensive / thorough "Detailed Feature Comparison Table"
- Gentoo wiki page: Comparison of init systems
- A history of modern init systems (1992-2015)
- Understanding (SysV) init
- init systems overview
- init system features and benefits
- Demystifying the init system (init written in Ruby)
- Process supervision: Problem apparently solved
- How process supervision and service management differ (according to Laurent Bercot, author of s6(-rc))

Init-specific articles:

- Upstart leaks memory (reported on 2018-03-16, still no reaction whatsoever)

- Hints on how to build, install, configure, and use ninit as your default init
- Hints on how to replace your current init with epoch and runit
- Gentoo wiki :: OpenRC
- guide: Using OpenRC on Arch Linux
- VoidLinux wiki :: Runit
- Obarun wiki :: s6-boot
- Replace systemd with busybox + minirc
- How to remove systemd from an Exherbo GNU/Linux installation
- An overview of daemontools family supervision suites
- Hints on how to perform socket activation with s6
- s6 init scheme (2): Exec chaining into different “init stages”
- runit init scheme (2, 3): Process #1 starts (and supervises) 3 subprocesses implementing different “init stages”
- Simpler, more reliable, supervision-friendly alternatives to syslogd and klogd: s6 + s6-ipcserver + ucspilogd + s6-log or runit + socklog

From:
https://without-systemd.frama.wiki/ - **Without Systemd**

Permanent link:
https://without-systemd.frama.wiki/alternatives_to_systemd?rev=1579205331

Last update: **2020/01/16 21:08**