
Fedora 22

Release Notes

Release Notes for Fedora 22



Edited by The Fedora Docs Team

Copyright © 2015 Fedora Project Contributors.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at <http://creativecommons.org/licenses/by-sa/3.0/>. The original authors of this document, and Red Hat, designate the Fedora Project as the "Attribution Party" for purposes of CC-BY-SA. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, MetaMatrix, Fedora, the Infinity Logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

For guidelines on the permitted uses of the Fedora trademarks, refer to https://fedoraproject.org/wiki/Legal:Trademark_guidelines.

Linux® is the registered trademark of Linus Torvalds in the United States and other countries.

Java® is a registered trademark of Oracle and/or its affiliates.

XFS® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

All other trademarks are the property of their respective owners.

Abstract

This document provides the release notes for Fedora 22. It describes major changes offered as compared to Fedora 21. For a detailed listing of all changes, refer to the Fedora Technical Notes.

1. Welcome to Fedora 22	2
-------------------------------	---

1.1. Welcome to Fedora	2
1.2. Overview	3
1.3. Hardware Overview	3
1.4. Feedback	5
2. Fedora Products	5
2.1. Fedora Cloud	6
2.2. Fedora Server	8
2.3. Fedora Workstation	9
2.4. Fedora Spins	10
3. Changes in Fedora for System Administrators	11
3.1. Kernel	11
3.2. Installation	11
3.3. File Systems	13
3.4. Virtualization	14
3.5. Web Servers	14
3.6. Server Configuration Tools	15
3.7. Big Data	17
4. Changes in Fedora for Desktop Users	18
4.1. Desktop	18
4.2. Networking	20
4.3. Internationalization	22
5. Changes in Fedora for Developers	24
5.1. Development	24
5.2. Development Tools	26
5.3. GCC Tools	27
5.4. Haskell	27
5.5. Web Development	28
A. Contributors	28
A.1. Writers	28
B. Revision History	29
Index	29

1. Welcome to Fedora 22

1.1. Welcome to Fedora

You can help the Fedora Project community continue to improve Fedora if you file bug reports and enhancement requests. Refer to [Bugs And Feature Requests](http://fedoraproject.org/wiki/BugsAndFeatureRequests)¹, on the Fedora wiki, for more information about bug and feature reporting. Thank you for your participation.

To find out more general information about Fedora, refer to the following pages, on the Fedora wiki (<http://fedoraproject.org/wiki>):

- [Fedora Overview](http://fedoraproject.org/wiki/Overview)²
- [Fedora FAQ](http://fedoraproject.org/wiki/FAQ)³

¹ <http://fedoraproject.org/wiki/BugsAndFeatureRequests>

² <http://fedoraproject.org/wiki/Overview>

³ <http://fedoraproject.org/wiki/FAQ>

- [Help and Discussions](#)⁴
- [Participate in the Fedora Project](#)⁵

1.1.1. Need Help?

There are a number of places you can get assistance should you run into problems.

If you run into a problem and would like some assistance, go to <http://ask.fedoraproject.org>. Many answers are already there, but if you don't find yours, you can simply post a new question. This has the advantage that anyone else with the same problem can find the answer, too.

You may also find assistance on the **#fedora** channel on the IRC net **irc.freenode.net**. Keep in mind that the channel is populated by volunteers wanting to help, but folks knowledgeable about a specific topic might not always be available.

1.2. Overview

As always, Fedora continues to develop ([Red Hat contributions](#)⁶) and integrate the latest free and open source software. The following sections provide a brief overview of major changes from the last release of Fedora.

For more details about the features that are included in Fedora 22 refer to the individual wiki pages that detail feature goals and progress: <https://fedoraproject.org/wiki/Releases/22/ChangeSet>

1.3. Hardware Overview

Fedora 22 provides software to suit a wide variety of applications. The storage, memory and processing requirements vary depending on usage. For example, a high traffic database server requires much more memory and storage than a business desktop, which in turn has higher requirements than a single-purpose virtual machine.

1.3.1. Minimum System Configuration

The figures below are a recommended minimum for the default installation. Your requirements may differ, and most applications will benefit from more than the minimum resources.

1GHz or faster processor
1GB System Memory
10GB unallocated drive space

⁴ <http://fedoraproject.org/wiki/Communicate>

⁵ <http://fedoraproject.org/wiki/Join>

⁶ http://www.fedoraproject.org/wiki/Red_Hat_contributions

Low memory installations

Fedora 22 can be installed and used on systems with limited resources for some applications. Text, vnc, or kickstart installations are advised over graphical installation for systems with very low memory. Larger package sets require more memory during installation, so users with less than 768MB of system memory may have better results performing a minimal install and adding to it afterward.

1.3.2. Display resolution

Graphical Installation requires 800x600 resolution or higher

Graphical installation of Fedora 22 requires a minimum screen resolution of 800x600. Owners of devices with lower resolution, such as some netbooks, should use text or VNC installation.

Once installed, Fedora will support these lower resolution devices. The minimum resolution requirement applies only to graphical installation.

1.3.3. Graphics Hardware

1.3.3.1. Minimum Hardware for Accelerated Desktops

Fedora 22 supports most display adapters. Modern, feature-rich desktop environments like GNOME3 and KDE Plasma Workspaces use video devices to provide 3D-accelerated desktops. Older graphics hardware may *not support* acceleration:

Intel prior to GMA9xx

NVIDIA prior to NV30 (GeForce FX5xxx series)

Radeon prior to R300 (Radeon 9500)

1.3.3.2. CPU Accelerated Graphics

Systems with older or no graphics acceleration devices can have accelerated desktop environments using LLVMpipe technology, which uses the CPU to render graphics. LLVMpipe requires a processor with SSE2 extensions. The extensions supported by your processor are listed in the `flags` section of `/proc/cpuinfo`

1.3.3.3. Choosing a Desktop Environment for your hardware

Fedora 22 Workstation's default desktop environment, GNOME3, functions best with hardware acceleration. Alternative desktops, such as those featured in [Section 2.4, "Fedora Spins"](#), are recommended for users with older graphics hardware or those seeing insufficient performance with LLVMpipe.

Desktop environments can also be added to an existing installation and selected at login. To list the available desktops, use the `dnf grouplist` command:

```
dnf group list -v hidden | grep desktop
```

Install the desired group:

```
dnf group install "KDE Plasma Workspaces"
```

Or, use the short group name to install:

```
dnf install @mate-desktop-environment
```

1.4. Feedback

Thank you for taking the time to provide your comments, suggestions, and bug reports to the Fedora community; this helps improve the state of Fedora, Linux, and free software worldwide.

To provide feedback on Fedora software or other system elements, please refer to [Bugs And Feature Requests](#)⁷. A list of commonly reported bugs and known issues for this release is available from [Common F22 bugs](#)⁸, on the wiki.

To provide feedback on the Release Notes, use <http://bugzilla.redhat.com/>⁹.

2. Fedora Products

For more than ten years, the Fedora Project has provided a distribution at the leading edge of the open source ecosystem. Fedora's releases have offered the latest technologies, integrating new and exciting upstream developments into a vast and diverse set of packages.

Users have built powerful desktops, reliable servers, and more recently, versatile cloud instances from the high quality packages in the Fedora repository. Fedora's strong commitment to upstream integrity gives developers a place to showcase their work, and benefit from Fedora's active testing and development volunteers.

In those ten years, much has changed. The **Fedora .next** initiative represents a considered strategy for maintaining the quality of the distribution and Fedora's position in introducing new technologies, while also providing a more consistent target for developers.

The result of this initiative is three distinct Fedora flagship flavors. Fedora Cloud for scalable infrastructure, Fedora Server for organizational infrastructure, and Fedora Workstation for the developer and desktop user.

Fedora Spins, such as live media featuring alternative desktop environments, will continue to be produced.

Fedora also offers images and support for an ever-increasing number of ARM devices, both large and small.

⁷ <http://fedoraproject.org/wiki/BugsAndFeatureRequests>

⁸ http://fedoraproject.org/wiki/Common_F21_bugs

⁹ [https://bugzilla.redhat.com/enter_bug.cgi?product=Fedora Documentation&component=release-notes](https://bugzilla.redhat.com/enter_bug.cgi?product=Fedora+Documentation&component=release-notes)

fedora-release Packages

The named Fedora flavors are identified in part by a set of packages with configuration details and dependencies specific to that flavor. For example, the presence of *fedora-release-server* identifies a system as Fedora Server, and ensures that core features like **Cockpit** are present.

Update or installation operations may involve the *fedora-release* packages, sometimes inadvertently due to the way dependencies are resolved. Environment groups are most susceptible to related conflicts, such as installing extra desktops on Fedora Workstation.

If you encounter package conflicts, add `--exclude fedora-release*` to your yum or dnf command. The [F21 Common Bugs](#)¹⁰ page describes this in detail.

The netinstall and DVD ISOs

The Fedora netinstall ISO, an installation image that allows you to select and download packages at install time, is now available as the Fedora Server netinstall.

The DVD installation image is not produced for Fedora 21.

2.1. Fedora Cloud

Fedora Cloud is the base building block of the Fedora flavors, ready to be deployed on the fly to meet your changing needs. These small images are backed by the vast set of applications and utilities available in the Fedora package repositories.

Cloud images are available in several varieties:

2.1.1. Cloud Base

Just the basics, this image is ready to be customized for your needs.

2.1.2. Atomic

Cutting edge [Project Atomic](#)¹¹ tools make this image the ideal host for containers. Instead of upgrades to individual packages, Atomic upgrades using **rpm-ostree** technology allow administrators to upgrade and roll back the entire core system as a single operation.

By combining Docker for versatile container deployments with SELinux to secure them, Fedora Atomic is an ideal solution for secure, modular deployments.

¹⁰ [https://fedoraproject.org/wiki/](https://fedoraproject.org/wiki/Common_F21_bugs#Installation_of_environment_groups.27_fails_due_to_conflicts_between_fedora-release_packages)

[Common_F21_bugs#Installation_of_environment_groups.27_fails_due_to_conflicts_between_fedora-release_packages](https://fedoraproject.org/wiki/Common_F21_bugs#Installation_of_environment_groups.27_fails_due_to_conflicts_between_fedora-release_packages)

¹¹ <http://www.projectatomic.io/>

2.1.3. Smaller Cloud images

Fedora maintainers have worked to reduce dependencies and streamline packaging to produce effective Cloud images that are 10% smaller than the previous release. Base qcow2 images are under 200MB, so your Fedora Cloud deployments will use minimal storage resources.

The reduced size is made possible by these notable changes:

- Fedora packages traditionally include upstream-provided documentation for user or administrator reference, as well as license text. Starting with Fedora 21, Fedora package maintainers have begun using a special location for license files, allowing Cloud images to be built with the license while avoiding the extra space consumed by unneeded documentation.

Packages which are included in the official Fedora Docker image, and more, now use a special location for license files, `/usr/share/licenses/`.

- Kernel packages are smaller due to improved packaging, as described in [kernel-packaging](#)

2.1.4. Converting Cloud instances to Fedora Server

Fedora Cloud excels for meeting organizational needs at scale with largely undifferentiated compute resources, but in some cases, an administrator might want to [adopt their cattle](#)¹² by converting a cloud instance to an individually managed Fedora Server installation.

To convert Fedora Cloud to Fedora Server, use the script provided in the `cloudtoserver` package.

2.1.5. Atomic Upgrade and Rollback via RPM-OSTree

Fedora 22 includes **RPM-OSTree**, a mechanism used on Fedora Atomic installations to perform atomic upgrades and rollbacks for the entire system (kernel as well as userspace).

Instead of performing a package-by-package installation and upgrade on each client machine, the tooling supports composing "sets of packages on a server side. Clients can then perform atomic upgrades as a tree.

On systems using **RPM-OSTree**, standard package managers (**DNF** and **Yum**) do not work normally; they can only be used in read-only mode.



Important

The version of **RPM-OSTree** available in the default updates repository and on installation media will not work due to a known issue. To use this tool, enable the `updates-testing` **DNF** repository and update the `rpm-ostree` package.

For additional information about this tool and Fedora Atomic, see the [Fedora Project Wiki](#)¹³.

¹² <http://www.slideshare.net/randybias/architectures-for-open-and-scalable-clouds/20>

¹³ <https://fedoraproject.org/wiki/Changes/RpmOstree>

2.1.6. Tunir

Tunir is a very simple CI (Continuous Integration) system written keeping Fedora Cloud images at mind. At the same time it is generic enough to be used by anyone to configure and run tests in their local system. The goal is to have a system which is simple to setup, and easy to maintain.



Important

This is a Self Contained Change.

This tool right now can create virtual machines based on cloud images (without needing an actual cloud), or can run the tests in a bare metal box, or it can even create jobs inside Docker containers.

Example:

```
sudo tunir --job dockerjob --stateless
```

The above command will run a stateless job named "dockerjob", it will not save the result into any database as it is a stateless run.

For additional information about this tool, see the [Tunir Documentation](#)¹⁴.

2.2. Fedora Server

2.2.1. Fedora Server Roles

A Featured Server role is an installable component of Fedora Server that provides a well-integrated service on top of the Fedora Server platform. These prepared roles simplify deployment and management of a service compared to setting up an upstream server from scratch; their use is recommended but optional; existing users of upstream servers based on Fedora RPMs will not be impeded.

For kickstart installations, you can use the *Fedora Server* environment group to deploy Server.

2.2.1.1. Framework for Server Role Deployment

A new D-Bus service is available, exposing available server roles, and making it possible to deploy, configure and manage them. Appropriate functionality will also be exposed as a command-line utility.

<https://fedoraproject.org/wiki/Changes/FrameworkForServerRoleDeployment>

2.2.1.2. Domain Controller Server Role

Fedora Server can deploy a domain controller powered by FreeIPA. The role greatly simplifies configuration of a primary domain controller.

When combined with SSSD, complex tasks such as single-sign-on and authenticated access to network resources is easily accomplished.

¹⁴ <http://tunir.rtfid.org/>

2.2.2. Database Server Role

Rapidly deploy instances of the powerful **postgresql** database server using the new Database Server Role for rolekit.

2.2.3. Cockpit Management Console

The **Cockpit Management Console** (the *cockpit* package) is available by default in Fedora Server. This tool provides a powerful, easy to use, web-based graphical interface for managing multiple Linux servers. Features include:

- systemd service management
- Journal log viewer
- Storage configuration including LVM
- **Docker** container management
- Basic network configuration
- Adding and removing local users

Any user known to the server can log in to the **Cockpit** console by opening `http://server-ip-address:9090`.

2.2.4. XFS as a Default File System

The XFS file system is now used by default when installing Fedora Server. See [Section 3.3.1, “XFS as a Default File System for Fedora Server”](#) for details.

2.3. Fedora Workstation

The Fedora Workstation product provides an easy to use, powerful environment for developers to both work and play. Desktop users can enjoy the familiar GNOME Desktop Environment, with support for devices and applications used every day. Developers will appreciate how Workstation is configured for their needs, and provides useful tools like **DevAssistant**.

For kickstart installations, you can use the *Fedora Workstation* environment group to deploy Workstation.

2.3.1. GNOME powered

Fedora Workstation developers have provided many enhancements to streamline and improve the GNOME Desktop Environment. Read more about changes to GNOME in Fedora 22 in [Section 4.1.2, “GNOME”](#)

2.3.2. Captive Portal Detection

Fedora Workstation, by default, enables a captive portal detection that requests known content from a trusted Fedora server. If the request is redirected, a window automatically appears for you to interact with the portal's login webpage.

To disable this feature, remove `/etc/NetworkManager/conf.d/20-connectivity-fedora.conf`

2.3.3. Developer oriented firewall

Developers often run test servers that run on high numbered ports, and interconnectivity with many modern consumer devices also requires these ports. The firewall in Fedora Workstation, `firewalld`, is configured to allow these things.

Ports numbered under 1024, with the exceptions of `sshd` and clients for **samba** and **DHCPv6**, are blocked to prevent access to system services. Ports above 1024, used for user-initiated applications, are open by default.

Refer to <https://fedoraproject.org/wiki/FirewallD> for details on customizing the firewall configuration, or install the `firewall-config` package for a graphical tool.

2.3.4. Nautilus Improvements

The Nautilus code base was cleaned up by porting it from the deprecated `GtkAction` API to `GAction`. As part of this, the view, gear and app menus are updated to match the current designs. In addition, the problematic floating statusbar and keyboard shortcut for deleting things are solved.

2.4. Fedora Spins

The Fedora Project also produces a number of **Spins** that provide alternative desktop environments, or package sets crafted for specific use cases. **Spins** are live images that you can use to preview a variety of software that Fedora has to offer.

Spins available for download from <https://spins.fedoraproject.org> have included:

- *KDE*

A complete, modern desktop built using the KDE Plasma Desktop.

- *Security*

Security analysis tools.

- *Electronic-Lab*

Fedora's high-end hardware design and simulation platform.

- *Scientific-KDE*

Open Source Scientific Computing.

- *Design-suite*

Open Creativity.

- *Games*

A perfect show-case of the best games available in Fedora.

- *Robotics*

Dive into Robotics.

- *Jam-KDE*

Unleash your inner musician.

3. Changes in Fedora for System Administrators

3.1. Kernel

Fedora 22 features the 4.0.0 kernel.

3.1.1. Modular Kernel Packaging

The *kernel* package is now a meta package that pulls in *kernel-core* and *kernel-modules*. The *kernel-core* package is smaller than a full package and is well-suited for virtualized environments. By optionally uninstalling *kernel-modules*, cloud image size can be reduced.

The *kernel-modules* package should be included when Fedora is installed on real hardware.

Fedora's initramfs is configured to include only drivers required for your system, allowing you to boot Fedora faster. A single, fully featured initramfs is provided under a **Rescue** boot entry in the GRUB menu to allow use after hardware changes. To recreate initramfs after hardware or driver changes, use the rescue boot option and run the command **dracut --regenerate-all**.



Initramfs Changes

Please note, that a new initramfs is only automatically generated by the *kernel-core* package but not the *kernel-modules* package. If you only installed *kernel-core* at first and install *kernel-modules* at a later point in time, you need to create a new initramfs manually using dracut, if any of the newly installed modules has become critical for your system's boot up.

The **dracut** utility is used to create the initramfs on Fedora. To regenerate an initramfs for all installed kernels, use the following command:

```
# dracut --regenerate-all
```

3.2. Installation

3.2.1. General Anaconda Changes

- Development of the **Anaconda** installer and related components such as *pykickstart*, *pyparted* and *initial-setup* has been moved from Fedorahostedg to Github (<https://github.com/rhinstaller>).
- Full documentation of Kickstart commands and options is now in the rhinstaller/pykickstart Github repository as well: <https://github.com/rhinstaller/pykickstart/blob/master/docs/kickstart-docs.rst>. The version on the Fedora Wiki has been removed. Keeping the documentation in the Git repository will allow the development team to maintain multiple, more accurate versions of documentation matching with various releases of pykickstart.
- The localization effort for **Anaconda** has migrated from Transifex to [Zanata](https://fedora.zanata.org/project/view/anaconda)¹⁵.

¹⁵ <https://fedora.zanata.org/project/view/anaconda>

- The new **DNF** package manager is now used to install packages. You can use the `inst.nodnf` option to revert back to **Yum** if needed. See [Section 3.6.1, “Yum replaced by DNF”](#) for more information about **DNF**.

3.2.2. Changes in Anaconda's Graphical Interface

- The advanced storage section of the **Manual Partitioning** screen now allows adding zFCP storage devices. The screen also now has a **Refresh** button, allowing you to refresh the list of network (iSCSI, FCoE, etc.) storage devices without having to leave the screen.
- The graphical interface now has animated transitions when moving back and forth between screens. This improvement aims to improve user experience by emphasizing the relationship between the main menu (**Installation Summary**) and other screens.
- **Anaconda** is now maximized, instead of full-screen, when running on top of a desktop (e.g. when installing from a Live DVD).
- When changing the settings on an existing connection in the **Network & Hostname** screen, you no longer have to turn the connection on and off for the changes to take effect.

3.2.3. Changes in Anaconda Boot Options

- The `inst.dnf` boot option, which was added in Fedora 21, has been replaced by the `inst.nodnf` option, which behaves in an opposite way. Use `inst.nodnf` to force the installer to use the older **Yum** package manager to install packages instead of **DNF**, which is now default.
- A new option, `inst.kdump_addon=`, has been added. Use `inst.kdump_addon=on` to enable the **Kdump** configuration add-on in the graphical and text user interface as well as in Kickstart. The **Kdump** configuration screen is disabled by default.

3.2.4. Changes in Kickstart Syntax

- The `--nobase` option for the `%packages` section has been removed.
- New command: `sshkey`. Use this command to install a SSH key to the `authorized_keys` file for a specified user using the following syntax:

```
sshkey --username=user "ssh_key"
```

Replace `user` with the user name, and `ssh_key` with the SSH key. The key must be enclosed in quotes because it may contain spaces. Also note that the user must either be `root`, or it must exist (must be created by the `user` command in the Kickstart file, or by a package specified in the `%packages` section).

- New section: `%anaconda`. This section can now be used in a Kickstart file to control the behavior of the installer, but not the installed system. Currently, the only command supported in this section is `pwpolicy`, described below. This section must end with an `%end` statement.
- New command: `pwpolicy`. This command sets password requirements such as minimum length for a named password entry.

```
pwpolicy name [--minlen=LENGTH] [--minquality=QUALITY] [--strict|notstrict] [--emptyok|notempty] [--changesok|nochanges]
```

Replace *name* with one of the following: **root**, **user**, or **luks**, to set a policy for the root password, user passwords, or LUKS (disk encryption) password.

Available options are:

- **--minlen=** - The minimum allowed password length. This parameter will be passed to the `libpwquality` library. The default minimum length is **8**.
- **--minquality=** - Minimum allowed quality of the password, as calculated by `libpwquality`. If the **--strict** option is used, passwords with lower quality will not be allowed. If **--notstrict** is used, using a password of lower than specified quality will display a warning require the user to click **Done** twice in the graphical user interface to confirm. The default quality value is **50**.
- **--strict** - Passwords with lower quality than specified in **--minquality** will be rejected completely. This is the default.
- **--notstrict** - Passwords with lower quality than specified in **--minquality** will be accepted, but **Anaconda** will display a warning and require the user to click **Done** twice before accepting the password.
- **--emptyok** - Allow empty passwords. This is the default.
- **--notempty** - Do not allow empty passwords.
- **--changesok** - Allow changing a password pre-configured in the Kickstart file to be changed interactively in the graphical user interface.
- **--nochanges** - Passwords set in the Kickstart file can not be changed in the GUI. This is the default.

The defaults are set in the `/usr/share/anaconda/interactive-defaults.ks` file provided by **Anaconda** on installation media. To override the default Kickstart file (and therefore change the installer's password policy), a `product.img` file with a separate `%anaconda` section must be created and passed to **Anaconda**.

3.3. File Systems

3.3.1. XFS as a Default File System for Fedora Server

The Fedora Server variant of Fedora 22 now uses the XFS file system by default. Other variants (Workstation, Cloud) continue to use `ext4` as a default; this can be changed during the installation.

XFS is a highly scalable, high-performance file system that supports file systems up to 16 exabytes (approximately 16 million terabytes), files up to 8 exabytes (approximately 8 million terabytes), and directory structures containing tens of millions of entries. XFS also supports metadata journaling, which facilitates quicker crash recovery. The maximum supported size of a single XFS file system is 500 TB (the limit for `ext4` is 50 TB).



Important

The size of an XFS file system can not be reduced after it is created - it can only be made bigger, not smaller. Use ext4 if you require the ability to shrink the file system at any point after the installation.

3.4. Virtualization

3.4.1. AArch64 QEMU/KVM VM Installation with libvirt and virt-manager Support

You may now use `libvirt` and `virt-manager` to install a virtual machine on the AArch64 (64-bit ARM) architecture with the KVM hypervisor. For specific instructions, see:

- [Architectures/AArch64/Install with QEMU](#)¹⁶
- [QA:Testcase Virt AArch64 on x86](#)¹⁷

3.4.2. UEFI VMs Installation with libvirt and virt-manager Support

UEFI installation options are now automatically available if UEFI/OVMF binaries are installed.

Instructions for installing virtual machines with UEFI are available at:

- [Using UEFI with QEMU](#)¹⁸
- [QA:Testcase Virtualization UEFI](#)¹⁹

3.5. Web Servers

3.5.1. Ipsilon

The **ipsilon** identity provider is now included in the Fedora 22 updates repository, allowing this application to be installed using the **DNF** package manager.

ipsilon is a server and a toolkit to configure Apache-based Service Providers. The server is a pluggable `mod_wsgi` application which provides federated single sign-on to web application. User authentication is always performed against a separate Identity Management system, such as an IPA server, and communication with applications is performed using a federation protocol such as SAML or OpenID.

See the [project page on Fedorahosted](#)²⁰ for more information.

¹⁶ https://fedoraproject.org/wiki/Architectures/AArch64/Install_with_QEMU

¹⁷ https://fedoraproject.org/wiki/QA:Testcase_Virt_AAarch64_on_x86

¹⁸ https://fedoraproject.org/wiki/Using_UEFI_with_QEMU

¹⁹ https://fedoraproject.org/wiki/QA:Testcase_Virtualization_UEFI

²⁰ <https://fedorahosted.org/ipsilon/>

3.6. Server Configuration Tools

3.6.1. Yum replaced by DNF

The *yum* package manager has been replaced in Fedora 22 by its successor, *dnf*. The yum fork has been available in Fedora for testing since Fedora 18, and is now the default command line package manager.

Most *dnf* commands use directives that are familiar to *yum* users, and it uses the same RPM package repositories. Behind the scenes, *dnf* uses an improved dependency solver, *hawkey*, along with *librepo* for repository operations and *libcomps* for package groups.

The `/usr/bin/yum` command will redirect to `/usr/bin/dnf` and print a warning about the redirection. The legacy *yum* package manager can be manually installed; the legacy command line utility has been renamed to **yum-deprecated**.

Read more about using *dnf*! Consult the upstream documentation at <http://dnf.readthedocs.org/en/latest/index.html>. Extra plugins are documented at <https://rpm-software-management.github.io/dnf/plugins-extras/index.htm>

The behavior of *dnf* differs from *yum* in some areas:

3.6.1.1. Updates that don't work are skipped

If a portion of a transaction is not viable, *dnf* will automatically exclude it and transparently continue with the portions that will work. For example, if a package has unmet dependencies during a **dnf update** action, that package will not be updated, but others will. This is similar to *yum*'s **--skip-broken** directive, but evaluates the impact of the problem against the entire transaction. Because this is the default behavior, there is no **--skip-broken** switch for *dnf*.

To reveal details about a problematic package direction, you can use the **--best** option. **dnf update --best** will force *dnf* to resolve the transaction using the latest versions of involved packages, and report any problems instead of skipping them. This is equivalent to *yum*'s behavior without **--skip-broken**.

3.6.1.2. Repos that don't work are skipped

If a configured and enabled repository does not respond, *dnf* will skip it and continue the transaction with the available repos. This differs from *yum*, which would immediately stop if a repository was not available.

3.6.1.3. Update and Upgrade are the same

The commands **dnf update** and **dnf upgrade** are equivalent. This differs from *yum*, where **yum upgrade** would have the same effect as **yum update --obsoletes**, and take obsolete packages into account.

3.6.1.4. Dependencies are not upgraded on package installation

When installing a new package, previously installed dependencies will not be upgraded. *Yum* offered an option for this behavior, **upgrade_requirements_on_install**. To upgrade with *dnf*, use **dnf update**.

If *dnf* reports that dependencies on installed packages are unmet while installing a new package, update the dependent packages before trying again.

3.6.1.5. Clean on remove

When removing a package, dnf will automatically remove any dependent packages that were not explicitly installed by the user. If a package was independently installed, it won't be uninstalled this way. Only packages installed as dependencies are removed.

This behavior is configured by the `clean_requirements_on_remove` option in `/etc/dnf/dnf.conf`

3.6.1.6. Repo cache refresh schedule

By default, dnf will check for updates in configured repositories hourly, starting ten minutes after the system boots. The action is controlled by a systemd timer unit, `/usr/lib/systemd/system/dnf-makecache.timer`.

To adjust this, copy the timer file to `/etc/systemd/system/dnf-makecache.timer` and edit it.

Alternatively, setting the `metadata_timer_sync` in `/etc/dnf/dnf.conf` to a number of seconds configures the minimum number of seconds between makecache operations. If the timer has not expired, `dnf makecache` will exit immediately.

dnf will also honor the `metadata_expire` option set in individual repo configs, and refresh repo metadata at runtime if it is too old. This option is described in `man yum.conf`.

3.6.1.7. Repository Actions

The `repository-packages` directive can be used to search for or get info about packages in a specific repository, list installed packages from that repository, and more. This simplifies operations that would have required use of `--excluderepo` and `--includerepo` options with yum, and is especially useful for managing similar packages from different repositories.

3.6.1.8. Listing dependencies

To find out what package supplies a particular provide, use the `dnf provides foo` command. This replaces `yum resolvedep foo`.

To list the dependencies of a package, use `dnf repoquery --requires foo`. This replaces `yum deplist foo`.

3.6.1.9. dnf will remove kernels

`kernel` packages are not protected by dnf. Unlike with yum, you can remove all kernel packages, including the running package, if you direct it to. Be cautious with removing kernels, and specify the full version and release when removing them for best results.

3.6.1.10. Replacing packages

When a system requires the capabilities of a package you want to replace, use the `--allowerasing` option. For example, `dnf --allowerasing mariadb` will allow you to replace mysql with mariadb, without disrupting packages that require capabilities provided by both packages. This replaces `yum shell` and `yum swap` functionality.

3.6.1.11. DNF Langpacks Plug-in

DNF supports installing language packs using the `dnf-langpacks` plug-in, which is expected to work identically to the older `yum-langpacks` plug-in. See [Section 4.3.3, "DNF Langpacks Plug-in"](#) for details.

3.6.1.12. Support for disabled repositories

The **Software** tool and **PackageKit** now support searching for packages in disabled repositories.

If a user searches for a package using one of these applications and the package is found in a repository which includes the line **enabled_metadata=1** in its definition, a dialog window will be displayed informing the user that the package has been found, but an additional repository must be enabled before it can be installed.

The same message can also inform the user about the reason why the repository is disabled by default.

This change allows Fedora remixes to ship pre-configured but disabled repositories for any reason - for example, if said repositories contain non-free software. Fedora itself does not have any such repositories pre-configured; therefore this feature will not be visible on a Fedora 22 installation unless you specifically configure one or more repositories with the **enabled_metadata=1** statement.

3.6.2. Preupgrade Assistant

Fedora 22 introduces the Preupgrade Assistant, a diagnostics utility which assesses the system for possible in-place upgrade limitations and provides a report with the analysis results. It is based on a module system, with each module performing a separate test, checking for package removals, incompatible obsoletes, changes in libraries, names changes, or deficiencies in the compatibilities of some configuration files. The Preupgrade Assistant does not modify your system except for storing log files.

Data gathered by the Preupgrade Assistant can be used for migrating the system using a Kickstart file. It also provides post-upgrade scripts to finish more complex problems after an in-place upgrade. The `preupgrade-assistant-contents` package is part of the `preupgrade-assistant` package and it delivers the set of scripts and plug-ins that are used to assess the system. Every module runs its own test and display an exit code that represents the result of that test (for example **PASS**, **FAIL**, **NEEDS_ACTION**, etc.). Contents can be done by users on the base of the Packaging Guidelines here: <http://fedoraproject.org/wiki/Packaging:PreupgradeAssistant>. Package owners are responsible for adding a module if it is suitable, for example changes in the MariaDB database between system versions.

To install the Preupgrade Assistant with all available contents, use the **dnf install preupgrade-assistant - *** command. You can find information on how to run the Preupgrade Assistant here: <https://github.com/phracek/preupgrade-assistant/blob/master/README>.

3.7. Big Data

3.7.1. Elasticsearch

The **Elasticsearch** indexing server has been integrated into the updates repository in Fedora 22. You can now install this application using **DNF** instead of relying on the stand-alone upstream installer.

Elasticsearch is a distributed, scalable, highly available search and analysis tool built on top of Apache Lucene, available under the Apache 2 license.

For information about **Elasticsearch**, see the [official project website](#)²¹.

²¹ <https://www.elastic.co/products/elasticsearch>

4. Changes in Fedora for Desktop Users

4.1. Desktop

4.1.1. GDM on Wayland

The **GNOME Display Manager** (GDM) in Fedora 22 will default to the **Wayland** display server instead of Xorg. While the default GNOME session still uses X, this change brings the move to Wayland one step closer.

Wayland is a compositing display server, using your computer's video hardware for rendering. On systems where Wayland will not run, GDM should transparently fall back to using the X backend.

If you need to disable Wayland for GDM, edit `/etc/gdm/custom.conf` to reflect the following:

```
[daemon]
WaylandEnable=false
```

4.1.2. GNOME

GNOME has been updated to the latest upstream release 3.16. This release comes with a number of feature improvements, some of which are listed below.

- The GNOME 3 notification system has been completely redesigned for 3.16. The message tray, used in previous versions of GNOME, was replaced with a new message list that the user can access from the GNOME Shell top bar.

GNOME Shell's calendar drop down has also been substantially changed to improve the overall user experience.

- The list and grid views in the **Files** application have been improved so that they are easier to use.
- The Activities overview, login screen and other parts of the GNOME Desktop come with an updated look.
- The 3.16 release introduces a new style of scrollbars that are only shown when the pointer is moved. They change size when the user clicks on them.
- The 3.16 release includes new games, **gnome-2048** and **gnome-taquin**.
- Support for codec, font, and MIME handler installation has been moved from **gnome-packagekit** to **gnome-software**. A new UI has been added to support these features.

For more information about the release, go read the GNOME 3.16 Release Notes at <https://help.gnome.org/misc/release-notes/3.16/>.

4.1.3. KDE

The Plasma desktop in Fedora 22 is upgraded to version 5, which is based on Qt 5 and KDE Frameworks 5. **Plasma 5** has a new theme called Breeze, which offers cleaner visuals and better readability, improves certain workflows and provides a more consistent and polished interface overall. Users familiar with KDE Workspace 4 should have no problem adapting to the new environment. The environment has been migrated to a fully hardware-accelerated graphics stack based on OpenGL (ES).

4.1.4. MATE

4.1.5. XFCE 4.12

Fedora 22 comes with an updated, improved version of the popular **Xfce** desktop environment. This new release introduces a number of bug fixes and enhancements to the desktop environment and its default applications.

Notable changes include:

- The window switcher (**Alt+Tab**) now supports themes and live previews and has several different modes.
- A Hidpi theme has been added for displays with high DPI.
- The Panel is now extensible via Gtk3 plug-ins, and supports intelligent hiding.
- The desktop has a new wallpaper settings dialog, per-workspace wallpaper support, and better multi-monitor handling.
- Support has been added for the new `libinput` input library.
- Many improvements were made to the **Thunar** file manager, including tab support, bug fixes and performance enhancements.
- The **Mousepad** text editor has been rewritten for better speed and simplicity.
- The **Parole** media player has been ported to Gtk3 and has a new interface.

See the [Xfce 4.12 Tour](#)²² for a visual tour of new features, and the [Xfce 4.12 Changelog](#)²³ for a full list of changes.

4.1.6. LXQt 0.9.0

LXQt is the Qt port and the upcoming version of **LXDE**, the Lightweight Desktop Environment. It is the product of the merge between the LXDE-Qt and the Razor-qt projects. Version 0.9.0 of this desktop environment is available in Fedora 22.

Like **LXDE**, this environment focuses on speed and low resource (CPU and RAM) consumption, making it especially suitable for systems with older or otherwise constrained hardware.

To install **LXQt**, use the `dnf groupinstall lxqt` command, and then select this environment the next time you log in. For more information about the project, see the [LXQt official website](#)²⁴. Also see the [LXQt 0.9 release announcement](#)²⁵ for detailed information about the 0.9.0 release.

²² <http://xfce.org/about/tour>

²³ <http://xfce.org/download/changelogs/4.12>

²⁴ <http://lxqt.org/>

²⁵ <http://sourceforge.net/p/lxde/mailman/message/33373317/>



Note

If you are interested in **LXQt**, consider joining the Fedora Project's [LXQt Special Interest Group](#)²⁶.

4.1.7. Qtile

Fedora 22 includes **Qtile** - a lightweight, extensible, tiling window manager written in Python. The available version is 0.9.1.

To install **Qtile**, use the `dnf install qtile` command.

See the [project website](#)²⁷ for introduction, general information, documentation and screenshots, and the [release notes](#)²⁸ for information about recent changes.

4.1.8. Libinput used for input devices

Input devices in supporting environments, notably GNOME with Fedora Workstation and KDE, will use a new driver, *libinput*. The new driver replaces a variety of drivers, such as *synaptics*, enabling more consistent behavior across a variety of devices.

libinput improves support for multi-touch devices and software emulated buttons. The driver is implemented directly in wayland sessions, and in X sessions through the *xorg-x11-drv-libinput* wrapper.

Input devices will be configurable through GNOME Settings, KDE System Settings, *xfce-settings*, or *xinput*. Some niche features are not available via *libinput*, but the previous behavior can be restored by removing the *xorg-x11-drv-libinput* package, and ensuring the appropriate x11 driver packages, probably *x11-drv-synaptics* or *xorg-x11-drv-evdev*, are installed.

Note that *xorg-x11-drv-libinput* is only installed by default on new Fedora 22 installations, if you are upgrading and you want to use the new features provided by *libinput*, you can install the package manually:

```
# dnf install xorg-x11-drv-libinput
```

To learn about the features and behavior of *libinput*, refer to `man libinput` or <http://wayland.freedesktop.org/libinput/doc/latest/pages.html>

4.2. Networking

4.2.1. BIND

BIND has been updated to the latest major release 9.10. This release comes with new features, some of which are listed below.

²⁶ https://fedoraproject.org/wiki/LXQt_SIG

²⁷ <http://www.qtile.org>

²⁸ <https://github.com/qtile/qtile/blob/master/CHANGELOG>

- New zone file format, "map", stores zone data in a format that can be mapped directly into memory, allowing significantly faster zone loading.
- New tool "delv" (domain entity lookup and validation) with dig-like semantics for looking up DNS data and performing internal DNSSEC validation has been added.
- New "prefetch" option improving the recursive resolver performance has been added.
- Improved EDNS processing allowing better resolver performance.
- Substantial improvements have been made in response-policy zone (RPZ) performance.
- ACLs can now be specified based on geographic location using the MaxMind GeoIP databases.
- The statistics channel can now provide data in JSON format as well as XML.
- The new "in-view" zone option allows zone data to be shared between views, so that multiple views can serve the same zones authoritatively without storing multiple copies in memory.
- Native PKCS#11 API has been added.
- New tool "named-rrchecker" can be used to check the syntax of individual resource records.
- New tool "dnssec-importkey" allows "offline" DNSSEC keys.
- Network interfaces are re-scanned automatically whenever they change.
- New "max-zone-ttl" option enforces maximum TTLs for zones.
- Multiple DLZ databases can now be configured, and are searched in order to find one that can answer an incoming query.
- "named-checkzone" and "named-compilezone" can now read journal files.

This release also includes feature changes, some of which are listed below.

- The version 3 XML schema for the statistics channel, including new statistics and a flattened XML tree for faster parsing, is no longer optional.
- The option "named" now listens on IPv6 as well as IPv4 interfaces by default.
- The internal and export versions of the BIND libraries have been unified so that external library clients can use the same libraries as BIND itself.
- The default setting for the -U option has been adjusted to improve performance.
- Adaptive mutex locks are now used on systems which support them.
- The option "rndc flushtree" now flushes matching records from the address database and bad cache as well as the DNS cache.
- The isc_bitstring API is no longer used and has been removed from the libisc library.
- The timestamps included in RRSIG records can now be read as integers indicating the number of seconds since the UNIX epoch, in addition to being read as formatted dates in YYYYMMDDHHMMSS format.

For more information about the release, go read the BIND 9.10 Release Notes at <http://ftp.isc.org/isc/bind9/9.10.0-P2/RELEASE-NOTES-BIND-9.10.0-P2.txt>.

²⁹ <https://fedora.zanata.org>

³⁰ https://fedoraproject.org/wiki/L10N/Translate_on_Zanata

4.3. Internationalization

Note

The localization effort for Fedora packages has moved from Transifex into [Zanata](#)²⁹. If you are interested in helping localize Fedora into your language, follow the instructions in the [Fedora Localization Guide](#)³⁰.

4.3.1. Changes in ibus

ibus is an input method switching framework. It is used to switch between input languages while logged into a desktop environment. Notable changes for ibus in Fedora 22 include:

- A user's ibus input method engines will automatically load when logging into GNOME, allowing them to swith engines without waiting for them to load.
- Input method engines are hidden on GNOME's lock screen.
- KDE has a special panel icon for ibus. Behind the scenes, this is determined by the `XDG_CURRENT_DESKTOP` environment variable, and will use `GtkStatusIcon` for supporting desktops and the KDE icon for KDE. If you log in with methods such as **startkde** instead of with a login manager, you may have to set this variable manually.

4.3.2. Unicode 7.0 Support in Glibc

An update to `Glibc` locale data (character map, character width, and `LC_CTYPE` information) in Fedora 22 enables support for Unicode 7.0. Previous Fedora releases supported Unicode 5.1. This change adds almost 8000 new characters, and also corrects Unicode data for some existing characters per the latest Unicode standard.

See the [backward compatibility report](#)³¹ for information about removed, changed and added characters.

4.3.3. DNF Langpacks Plug-in

The **DNF** package manager, which replaces **Yum** in Fedora 22, now supports langpack installation through the `dnf-langpacks` plug-in. The functionality is the same as the `yum-langpacks` plug-in for **Yum**.

³¹ <https://raw.githubusercontent.com/pravins/glibc-i18n/master/report-utf8>

 **Note**

Currently, the plug-in can not install langpacks automatically due to a missing **DNF** feature. See BZ#[1114422](#)³² for details. Manual installation is possible using the **dnf langinstall *language*** command, and a list of available languages can be obtained using **dnf langavailable**.

To display all available commands for *dnf-langpacks*, use the **dnf --help** command. All listed commands beginning with **lang*** are provided by this plug-in.

4.3.4. New Default Console Font

The default console font has been changed to `eur1atgr` in Fedora 22. The new font has the same typeface as the previously used `latarcyrheb-sun16` font, but supports a broader range of characters from the Latin and Greek alphabets as well as some commonly used symbols. Users should therefore notice less replacement characters displayed on the console when texts using non-ASCII characters are being displayed.

The `eur1atgr` font does not support Arabic, Cyrillic and Hebrew characters; messages written in these alphabets will continue to use the `latarcyrheb-sun16` font.

Full documentation detailing supported code pages and characters is available in the font's [README file](#)³³.

4.3.5. Minglish Input Method for the Marathi Language

A new input method, "Minglish", is available for the Marathi language in Fedora 22.

Minglish allows you to write Marathi text phonetically using an English keyboard layout, based on the English letter pronunciation. For example, the word `####` is formed by typing **anish**, whereas normally you would have to type **FniS** using the traditional phonetic input method, or **anisha** using ITRANS.

To enable the new input method on your system, install the *m17n-db* package using **DNF**, and then select Minglish in your desktop environment or IBus input preferences.

4.3.6. Lohit2 Odia Font Update

Fedora 22 brings an update to the Lohit Odia font, which is the default font for the Odia language in Fedora.

This update aims at cleaning up Odia type tables and make them effective and efficient by following all the standards around font technology. It makes this font follow the latest open type specification and incorporates changes made in language guidelines in recent years.

Users should not notice any significant changes apart from some "rare" words now being displayed correctly.

³² https://bugzilla.redhat.com/show_bug.cgi?id=1114422

³³ <http://git.altlinux.org/people/legion/packages/kbd.git?p=kbd.git;a=blob;f=data/consolefonts/README.eurlatgr>

See <http://pravin-s.blogspot.in/2013/08/project-creating-standard-and-reusable.html> for additional information about changes made in the new version. The fonts are provided by the *google-noto-sans-oriya-fonts* and *google-noto-sans-oriya-ui-fonts* packages.

5. Changes in Fedora for Developers

5.1. Development

5.1.1. Perl 5.2.0

Perl has been updated to version 5.20 in Fedora 22. This new release brings several new features, changes to existing behavior, and removes a few features; notable changes are described below.

New features:

- Subroutine **prototype** attribute allows to declare subroutine prototype in postfix manner.
- The new **%hash{ . . . }** and **%array{ . . . }** syntax returns a list of key/value and index/value pairs, respectively.
- Added support for Unicode 6.3 tables.
- More internal counters support 64-bit values (arrays with more than 2^{32} elements, regular expression matching strings with more than 2^{32} characters).
- Copy-on-write mechanism is used when assigning scalars.
- Speed improvements (lazy evaluation of logical conjunction and disjunction in void context, constant hash look-ups, return at the end of a subroutine).
- New modules have been added into the core (**experimental**, **IO::Socket::IP**).

Changes in existing behavior:

- **use locale** in an UTF-8 locale environment enables Unicode character type processing.
- Perl initializes locale according to all locale environment variables (such as **LANG**).
- The **-F** Perl option implies the **-a** option, and the **-a** option implies the **-n** option.
- Character after a **\c** sequence in a double-quoted string or regular expression must be a printable character and may *not* be a **{**.
- In **use locale** scope, regular expression matching which depends on locale properties (like character case) taints values.
- The regular expression pattern element **\p{All}** previously matched only Unicode code points. Now it matches all code points, making it equivalent to **qr/. /s**. Therefore, **\p{All}** is no longer synonymous with **\p{Any}**, which continues to match only Unicode code points.
- The output format for **Data::Dumper** has changed.
- Locale decimal point in Perl code is subject to locale only in scope of **use locale**.

Removed features:

- `do subroutine(LIST)` is no longer supported and results in a syntax error.
- The `PerlIO_vsprintf()` and `PerlIO_sprintf()` functions have been removed.

For more information, see the [Perl 5.20 perldelta](#)³⁴, [Perl 5.20.1 perldelta](#)³⁵, and [Perl 5.20.2 perldelta](#)³⁶.

5.1.2. Unicode 7.0 Support in Glibc

Glibc now supports Unicode specification 7.0. See [Section 4.3.2, “Unicode 7.0 Support in Glibc”](#) for information.

5.1.3. Ruby 2.2

Fedora 22 provides Ruby 2.2, the latest stable version of Ruby. This update brings many updates necessary for the increasingly diverse and expanding demands of this language such as incremental and symbol garbage collection and many others.

This update also brings a soname bump. All Ruby packages which use binary extensions should be rebuilt. Ruby 2.2 preserves backward compatibility with Ruby 2.1 (introduced in Fedora 21); therefore no changes to source code are necessary.

You can find a full list of changes in Ruby 2.2 in the [changelog](#)³⁷. Full documentation is available at the [Ruby Documentation](#)³⁸ website.

5.1.4. python-dateutil 2.4

The `python-dateutil` module, which provides extensions to the standard `datetime` module available in Python, has been updated to version 2.4 in Fedora 22.

Previous releases of Fedora provided version 1.5 of this module. Upstream releases immediately following this version only supported Python 3, which meant the Fedora package could not be updated without breaking compatibility. The current version will work with both Python 2 and 3.

Separate subpackages are provided for Python 2 and 3: The `python-dateutil` package should be used for Python 2 applications, and the `python3-dateutil` package provides support for Python 3.

See the [upstream python-dateutil release notes](#)³⁹ for detailed information about this release. Also see the [README file](#)⁴⁰ for general module information and usage examples.

5.1.5. wxPython 3

Fedora 22 includes version 3 of wxPython - Python bindings for the wxWidgets GUI library.

This toolkit allows you to create programs with a robust, highly functional graphical user interface simply and easily. It is implemented as a Python extension module (native code) which wraps the popular wxWidgets cross-platform GUI library, which is written in C++.

³⁴ <http://search.cpan.org/dist/perl-5.20.0/pod/perldelta.pod>

³⁵ <http://search.cpan.org/dist/perl-5.20.1/pod/perldelta.pod>

³⁶ <http://search.cpan.org/dist/perl-5.20.2/pod/perldelta.pod>

³⁷ https://github.com/ruby/ruby/blob/v2_2_0/NEWS

³⁸ <http://ruby-doc.org/>

³⁹ <https://github.com/dateutil/dateutil/blob/master/NEWS>

⁴⁰ <https://github.com/dateutil/dateutil/blob/master/README.rst>

For more information about changes in this version, see the [Recent Changes in wxPython](#)⁴¹ page. Also see Debian Project's [wxpy30-update script README](#)⁴² for information about migrating to version 3.0.

5.2. Development Tools

5.2.1. Gradle 2.2

Fedora 22 brings **Gradle 2**, the popular build automation tool written in Java. **Gradle** can automate building, testing, publishing, deployment etc. of software packages or other types of projects, such as automatically generated documentation or websites.

This change also implements integration with software used for Java packaging in Fedora (*XMvn* and *Javapackages*), which makes it possible to use standard Fedora packaging techniques to build RPM packages with **Gradle** with all features such as automatic artifact installation or auto-requires/provides.

For more information about the project, see the current version documentation at the [official Gradle website](#)⁴³.

5.2.2. Git 2.4.1

Fedora 22 has includes **Git 2.4.1**. Git is a very popular version control and collaboration tool. The previous version of Git was 2.1.0 in Fedora 21.

New Features:

git push --atomic

New to Git in 2.4 and for Fedora is the *atomic* option. The *--atomic* option does an all - or - nothing push update. This can be useful for automated tools. An example would be adding *--atomic* to the end of a simple git push: **git push origin --atomic** For more information on how to do an atomic push please visit <https://github.com/blog/1994-git-2-4-atomic-pushes-push-to-deploy-and-more>.

Push to Deploy

Introduced in Git 2.3.0 and new to Fedora is the ability to push to deploy. This feature will allow someone to push a branch to a git server even if the branch is checked out on that server.

Inverted Grep

Inverted grep, or **--inverted grep** will let you search for items that do not match the pattern you have entered. Inverted grep will give the ability to do "everything else but" searching. For example, if you wanted to search the git log for everything the user grundblom did that was not grammer edits You could use the command: **git log --all --author=grundblom --invert-grep --grep=grammer**

Other enhancements and updates have been made in the following areas:

- Ports

⁴¹ <http://wxpython.org/recentchanges.php>

⁴² <http://anonscm.debian.org/cgi/collab-maint/wx-migration-tools.git/tree/README>

⁴³ <http://gradle.org/docs>

- UI, Workflows, and Features
- Performance
- Internal Implementation
- Development Support

For more information about **Git 2.4.1**, please visit the release notes at: <https://raw.githubusercontent.com/git/git/master/Documentation/RelNotes/2.4.1.txt>.

5.3. GCC Tools

5.3.1. GCC 5

The GNU Compiler Collection (**GCC**) has been upgraded to version 5, and Fedora packages with dependencies on specific versions of **GCC** have been rebuilt with this new version.

This update provides a large number of bug fixes and enhancements. The notable newly supported features and changes are:

- Improved C++11 support and added C++14 support.
- The default mode for C is now `-std=gnu11` instead of `-std=gnu89`.
- OpenMP 4.0 offloading support.
- OpenACC 2.0 support.
- Improved vectorization support.

See the [GCC 5 Release Notes](#)⁴⁴ for a full list of changes. For help with porting packages previously built with an earlier version of **GCC**, see the [Porting to GCC5](#)⁴⁵ document.

5.3.2. Boost 1.57

The set of widely used C++ libraries, **Boost**, has been updated to version 1.57 in Fedora 22. This update brings a number of bug fixes, performance improvements and enhancements.

Boost 1.57 brings a soname bump. Fedora packages have been rebuilt using the updated libraries where necessary. Any other client packages must be rebuilt as well; this update does not preserve full ABI compatibility with previous releases.

See the [Boost 1.57 Release Notes](#)⁴⁶ for detailed information about changes in the new version.

5.4. Haskell

5.4.1. GHC 7.8

The **GHC** Haskell compiler has been updated to version 7.8 in Fedora 22.

⁴⁴ <https://gcc.gnu.org/gcc-5/changes.html>

⁴⁵ https://gcc.gnu.org/gcc-5/porting_to.html

⁴⁶ http://www.boost.org/users/history/version_1_57_0.html

Fedora packages have been updated and rebuilt with the new **GHC** version. User packages will require a rebuild as well, and may need code changes before they build with the new version.

For full release notes, describing all bug fixes and enhancements, see the [GHC System User's Guide](#)⁴⁷. This document contains information about **GHC 7.8.1** as well as several subsequent bug fix updates, which are also included in Fedora 22.

5.5. Web Development

5.5.1. Django 1.8

The latest release of the versatile, extensible web framework **Django** is available for Fedora 22. The Python-based framework's 1.8 release has been designated as the Django project's second 'long term support release, and will receive security updates for at least three years from release date.

Fedora packages that depend on Django have been updated for the latest release. Those working with Django can review the project's thorough documentation to learn about new, deprecated, and removed features.

- New features: <https://docs.djangoproject.com/en/dev/releases/1.8/#what-s-new-in-django-1-8>
- Backwards incompatible changes: <https://docs.djangoproject.com/en/dev/releases/1.8/#backwards-incompatible-changes-in-1-8>
- Deprecated features: <https://docs.djangoproject.com/en/dev/releases/1.8/#features-deprecated-in-1-8>
- Removed Features: <https://docs.djangoproject.com/en/dev/releases/1.8/#features-removed-in-1-8>

5.5.2. Ruby on Rails 4.2

Fedora 22 includes **Ruby on Rails** 4.2, the latest version of the well-known web application framework written in Ruby. Highlights in this release include Active Job, asynchronous mails, Adequate Record, Web Console and foreign key support. The upstream release notes are at http://guides.rubyonrails.org/4_2_release_notes.html

Information for developers upgrading existing applications to Ruby on Rails 4.2 can be found in the Upgrading Ruby on Rails guide at http://guides.rubyonrails.org/upgrading_ruby_on_rails.html

A. Contributors

A large number of people contribute to Fedora each release. Among these are a number of writers and translators who have prepared these release notes. The following pages list those contributors.

A.1. Writers

Fedora Documentation Project

- The Fedora Docs team

⁴⁷ https://downloads.haskell.org/~ghc/7.8.4/docs/html/users_guide/index.html

B. Revision History

Revision **Thursday, May 28 2015**
22-03

Corrections to the GNOME & Git beats

Fedora Docs
[docs AT lists.fedoraproject.org](https://docs.fedoraproject.org)

Revision **Monday, May 25 2015**
22-02

Fedora 22 Final release

Fedora Docs
[docs AT lists.fedoraproject.org](https://docs.fedoraproject.org)

Revision **Tuesday, April 21 2015**
22-01

Fedora 22 Beta release

Fedora Docs
[docs AT lists.fedoraproject.org](https://docs.fedoraproject.org)

Revision **Wednesday, April 8 2015**
22-00

Cleared content for pre-release preparation

Fedora Docs
[docs AT lists.fedoraproject.org](https://docs.fedoraproject.org)

Index

B

Bug Reporting, 2

F

FAQ

 Fedora, 2

 Fedora.next, 5

H

Help, 2

K

kernel, 11

L

Linux kernel, 11

R

Reporting

 Bug and Feature Request, 2

W

Wiki

 Fedora, 2

