



# 红帽 Linux 系统企业版 7 桌面迁移与管理指南

---

GNOME 3 桌面的迁移规划和管理之指南。

Petr Kovář                      Jana Švárová

翻译、校对：柳叶 - Ye (Echo)

Liu

翻译、校对：吴敏峰 - Minfeng (Mindy)

Wu

校对、编辑：任浩 - Hao (Naomi)

Ren

校对、责任编辑：鄭中 - Chester

Cheng



## GNOME 3 桌面的迁移规划和管理之指南。

Petr Kovář  
Red Hat 工程部出版中心  
pkovar@redhat.com

Jana Švárová  
Red Hat 工程部出版中心  
jsvarova@redhat.com

翻译、校对：柳叶 – Ye (Echo) Liu  
澳大利亚昆士兰大学 笔译暨口译研究所  
liu.ye.ch@foxmail.com

翻译、校对：吴敏峰 – Minfeng (Mindy) Wu  
澳大利亚昆士兰大学 笔译暨口译研究所  
ylwt\_5@126.com

校对、编辑：任浩 – Hao (Naomi) Ren  
澳大利亚昆士兰大学 笔译暨口译研究所  
renhao0823@gmail.com

校对、责任编辑：鄭中 – Chester Cheng  
红帽全球服务部 & 澳大利亚昆士兰大学笔译暨口译研究所  
ccheng@redhat.com, uqcchun1@uq.edu.au

## 法律通告

Copyright © 2014 Red Hat, Inc.

This document is licensed by Red Hat under the [Creative Commons Attribution-ShareAlike 3.0 Unported License](https://creativecommons.org/licenses/by-sa/3.0/). If you distribute this document, or a modified version of it, you must provide attribution to Red Hat, Inc. and provide a link to the original. If the document is modified, all Red Hat trademarks must be removed.

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, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

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.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## 摘要

《桌面迁移与管理指南》记录了对红帽 Linux 7 企业版之 GNOME 3 桌面系统的迁移规划、部署、配置和管理。目标读者为对红帽 Linux 系统企业版有基本了解的系统管理员。

# 目录

<b>第 1 章 介绍 GNOME 3 桌面系统</b>	<b>3</b>
1.1. GNOME 3 是什么?	3
1.2. 什么是 GNOME shell ?	4
1.3. 什么是 GNOME Classic ?	5
1.4. 获取更多信息	7
<b>部分 I. 迁移规划</b>	<b>8</b>
<b>第 2 章 logind</b>	<b>9</b>
2.1. 获取更多信息	10
<b>第 3 章 GSettings 和 dconf</b>	<b>11</b>
3.1. 获取更多信息	11
<b>第 4 章 PolicyKit</b>	<b>12</b>
4.1. 策略配置	12
4.2. 默认策略	12
4.3. 脚本语言的检查权限	13
4.4. 扩展 polkit 配置	13
<b>第 5 章 GDM</b>	<b>14</b>
5.1. 获取更多信息	14
<b>第 6 章 GNOME Shell 扩展</b>	<b>15</b>
6.1. 时钟小程序的置换	15
<b>第 7 章 gnome-session</b>	<b>16</b>
7.1. 获取更多信息	16
<b>第 8 章 国际化</b>	<b>17</b>
8.1. 输入法	17
8.2. 文件位置的改变	17
<b>部分 II. 配置和管理</b>	<b>18</b>
<b>第 9 章 用 GSettings 和 dconf 来配置桌面</b>	<b>19</b>
9.1. 术语解释：GSettings、gsettings，以及 dconf	19
9.2. 用户和系统设置	19
9.3. 浏览桌面应用的 GSettings 值	19
9.4. 何为 dconf 配置文件？	20
9.5. 设置自定义默认值	21
9.6. GSettings 键值属性	23
9.7. 存储用户设置于 NFS	23
<b>第 10 章 默认外观</b>	<b>25</b>
10.1. 在 Anaconda 中置入徽标	25
10.2. 在 Firstboot 中置入徽标	25
10.3. 引导装载程序屏幕	26
10.4. Plymouth	27
10.5. 自定义登录屏幕	30
10.6. 自定义桌面背景	32
10.7. 设置字体	35
<b>第 11 章 GNOME Shell 扩展</b>	<b>41</b>
11.1. GNOME Shell 扩展是什么？	41

11.2. 启用计算机范围的扩展程序	42
11.3. 锁定被启用的应用程序	43
11.4. 设置强制性扩展程序	43
<b>第 12 章 整合应用程序</b>	<b>45</b>
12.1. 自定义菜单	45
12.2. 自定义默认的最热门程序	48
12.3. 配置文件关联	50
<b>第 13 章 自定义 GNOME 系统的桌面功能</b>	<b>56</b>
13.1. 启用 Ctrl+Alt+Backspace 快捷方式	56
13.2. 启用构成键	56
13.3. 禁用命令行	57
<b>第 14 章 会话管理</b>	<b>59</b>
14.1. 何为 GDM ?	59
14.2. 认证	59
14.3. 用户会话	61
<b>第 15 章 虚拟文件系统和磁盘管理</b>	<b>69</b>
15.1. GVFS	69
15.2. GVFS 后端	69
15.3. 装载、卸载和弹出	70
15.4. Nautilus 和【打开】或【保存】文件对话书签	71
15.5. 设置默认服务器列表	71
15.6. 向所有其它应用程序显示 GNOME 虚拟文件系统	72
15.7. GVFS 装载的密码管理	73
15.8. GNOME 中的 GVFS 工具和 xdg-utils	73
15.9. 使用 GVFS 元数据	74
15.10. 疑难解答	75
<b>第 16 章 硬件配置</b>	<b>78</b>
16.1. 数位屏	78
<b>附录 A. 访问红帽文档</b>	<b>80</b>
<b>附录 B. 修订历史</b>	<b>81</b>

# 第 1 章 介绍 GNOME 3 桌面系统

## 1.1. GNOME 3 是什么？

GNOME 3 系统是红帽 Linux 7 企业版默认的桌面环境。它是新一代 GNOME 桌面系统版本，与之前和红帽企业版 Linux 5、Linux 6 一起推出的 GNOME 2 相比而言，GNOME 3 引进了新的使用者界面，以及大量对特性的改进。

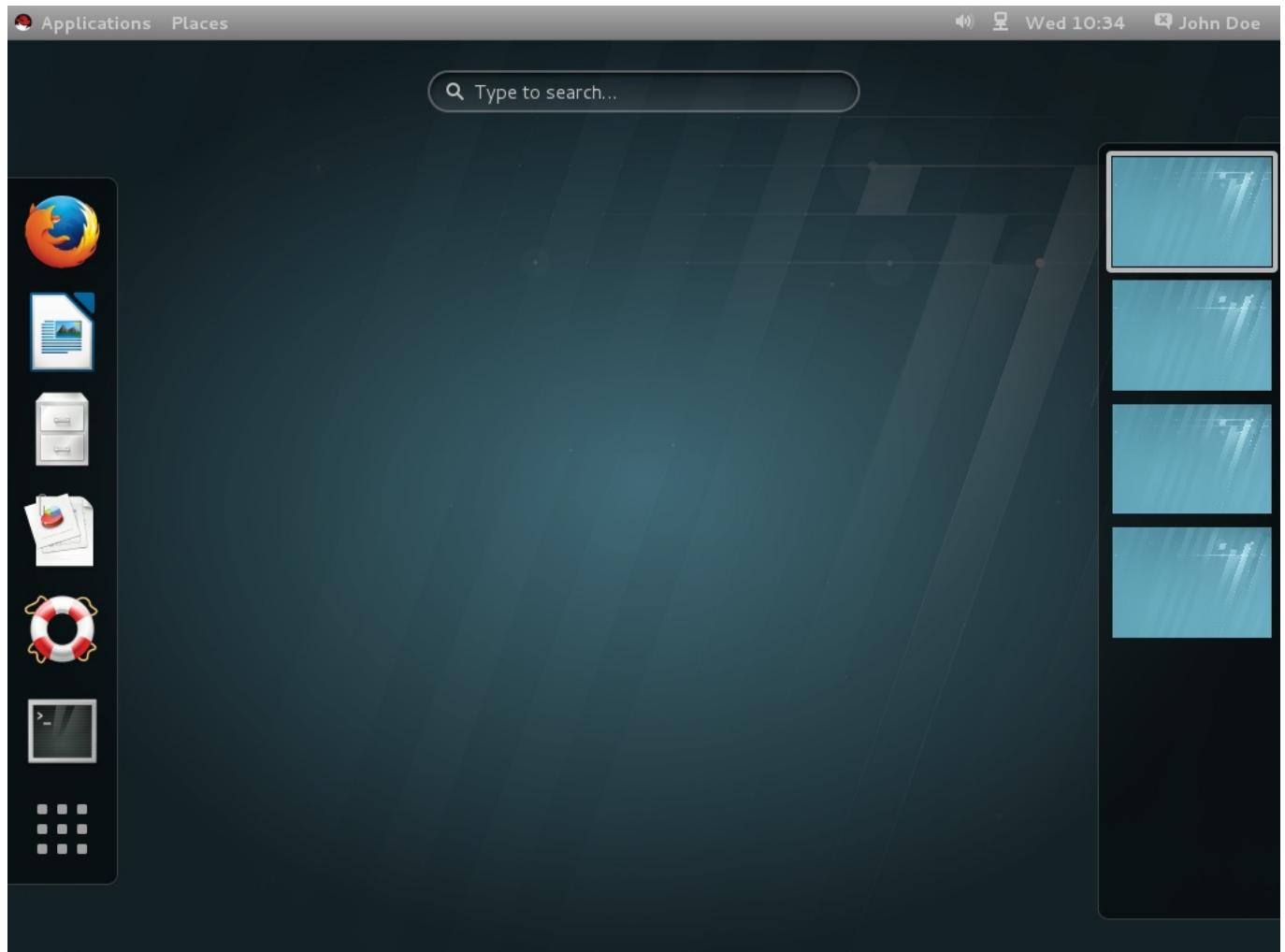


图 1.1. GNOME 3 桌面系统 (GNOME 经典版)

GNOME 3 提供了一个提升生产力的纯工作环境。强大的搜索功能让您可以在同一个工作地点访问到所有工作内容。比如，当您需专注于手边的工作任务时，您可以关闭通知功能。GNOME 3 建立在很多有效的组件上：

### GNOME Shell

GNOME Shell 系统是图像直观的现代用户界面。它提供了高质量的用户体验，其中包含视觉效果和硬件加速支持。详细信息请参考 [〈第 1.2 节 “什么是 GNOME shell？”〉](#)。

### GNOME 经典版

GNOME 经典版是新旧版本的结合；它保持了 GNOME 2 熟悉的外观和感受，但增加了 GNOME shell 系统强大的新特点和 3-D 功能。GNOME 经典版是红帽企业版 Linux 7 系统的默认 GNOME 会话和 GNOME shell 模式。详细信息请阅读章节 [〈第 1.3 节 “什么是 GNOME Classic？”〉](#)。

### GSetting

**GSettings** 是一个配置存储系统，它代替了旧 GNOME 版本中的“**GConf**”。更多有关“**GConf**”到 **GSettings** 转变的信息，请参见〈[第 3 章 GSettings 和 dconf](#)〉。更多关于如何使用 **GSettings** 配置桌面的信息，请阅读〈[第 9 章 用 GSettings 和 dconf 来配置桌面](#)〉。

## GVFS

总体来说，“**GVFS**”为 GNOME 桌面系统提供了完整的虚拟文件系统设施并对储存器进行管理。GNOME 3 通过“**GVFS**”与在线文件存储服务、日历和通讯录很好地结合，因此您可以在同一个地方对所有数据进行访问。更多内容请阅读〈[第 15 章 虚拟文件系统和磁盘管理“GVFS”](#)〉。

## GTK+

“**GTK+**”是用来创建图形用户界面的多平台工具包，它提供了高可用性、功能多样化的应用程序界面。有了“**GTK+**”，GNOME 3 才能够改变应用程序外观，或者提供平滑的图像外观。除此之外，“**GTK+**”还包含了大量的特点，比如支持面向对象编程（GObject）、充分支持国际字符集和文本布局（Pango），或者包含一组访问接口（ATK）。

## 1.2. 什么是 GNOME shell ?

GNOME shell 是 GNOME 桌面的用户界面，是 GNOME 3 的关键技术。它提供了一些基本的用户界面功能，比如切换窗口，启动应用程序或者显示通知。

GNOME shell 引进了创新的用户界面概念，提供了高质量的用户体验，其中包括现代图形硬件对系统硬件的加速。

GNOME shell 用户界面的一些主要组件有：

### 顶部菜单栏。

屏幕顶部的水平导航栏提供对一些 GNOME shell 基本功能的访问路径，比如“**活动概览**”，时钟和日历、系统状态图标，以及屏幕左上方的系统菜单。

### 活动概览。

视窗和应用视图是“**活动概览**”的主要特点，可以使用户运行应用程序和视窗，并在这两者之间进行切换。

顶部的“**检索入口**”允许用户搜索桌面上多个可用的项目，包括应用程序、文档、文件和配置工具。

左侧的垂直条被称为“**dash**”，它包含了收藏的和正在运行中的应用程序列表。

“**工作区列表**”显示在右侧，它允许用户在多个工作区间进行转换，或者将应用程序和视窗从一个工作区转移到另一个工作区。

### 讯息匣。

“**讯息匣**”是靠近屏幕底部的水平栏，它会在用户键入 **Super+M** 时显示。它提供对搁置通知的访问。

### GNOME Classic 的特定组件。

*GNOME Classic* 是红帽 Linux 7 系统企业版 GNOME shell 模式的默认设置。它改变了 GNOME Shell 某些方面的行为以及 GNOME shell 的外观。其中包括底部栏的窗口列表，和顶部栏中的“**应用程序**”以及“**位置**”。更多有关 GNOME Classic 的详细信息，请参见〈[第 1.3 节 “什么是 GNOME Classic ?”](#)〉。

### 1.2.1. 硬件加速和软件渲染

GNOME Shell 的特点是它的视觉效果，同时它利用了“**Clutter**”提供的硬件加速支持，一个以 OpenGL 为基



础的图形库。

为了使硬件加速器能够正常工作，显卡驱动必须能够支持 GL 1.2 和多纹理扩展，或者能够支持 GL 1.3。另一种选择是驱动必须支持 GLES 1.1 或者 GLES 2.0。要记住，很多处理器型号和驱动并不能为 GL 或者 GLES 提供恰当的支持，因此硬件加速器在包含那些处理器和驱动的系统是不可用的。

对无法满足处理器和硬件要求的系统，以及虚拟计算机来说，软件渲染则可以通过可支持的硬件加速来提供与 GNOME 3 相同的用户体验。软件渲染是由“**llvmpipe**”驱动提供的。

您可以运行 **glxinfo** 命令，来确定系统是否使用软件渲染和“**llvmpipe**”驱动。

```
$ glxinfo | grep renderer
OpenGL renderer string: Gallium 0.4 on llvmpipe (LVVM 3.3, 128 bits)
```

要注意的是，因为软件渲染不能够提供并实现完全一致的 OpenGL，那么，如果一些系统在应用程序中需要依靠和 GLX 视图状态一致的 X server，则它们有可能不能正常工作。此时，请考虑升级您的硬件，或者运行主机和驱动能够完全支持硬件加速的系统程序。

### 1.3. 什么是 GNOME Classic ?

GNOME Classic 是 GNOME Shell 的一个特点和模式，提供给那些倾向于更加传统的桌面体验的用户。GNOME Classic 则以 GNOME 3 的技术为基础，提供了大量的用户界面更改。

**“应用程序”和“位置”菜单。**

**“应用程序”**菜单显示在屏幕的左上方。它准许用户使用按类别分组的应用程序。用户也可以在此菜单上打开**“活动概览”**。

**“位置”**菜单在**“应用程序”**菜单的旁边，并显示在**“顶部菜单栏”**上。它允许用户快速访问重要的文件夹，比如**“下载”**或者**“图片”**。

**任务栏。**

任务栏显示在屏幕的底部，特点有：

- ✧ 窗口列表、
- ✧ 在窗口列表旁，显示的提示图标、
- ✧ 在提示图标旁，现实的当前工作空间短识别符，以及可用空间的总量。

**四个可用的工作空间。**

在 GNOME Classic 中，用户可用的默认工作空间量的默认设置为四个。

**最小化、最大化按钮。**

GNOME Classic 窗口标题栏的特点是最小化和最大化按钮，它们可以让用户快速地将窗口最小化到窗口列表中，或者将窗口最大化到桌面大小。

**传统的 Super+Tab 窗口切换器。**

在 GNOME Classic 中，**Super+Tab** 窗口切换器中的窗口不在应用程序分组中。

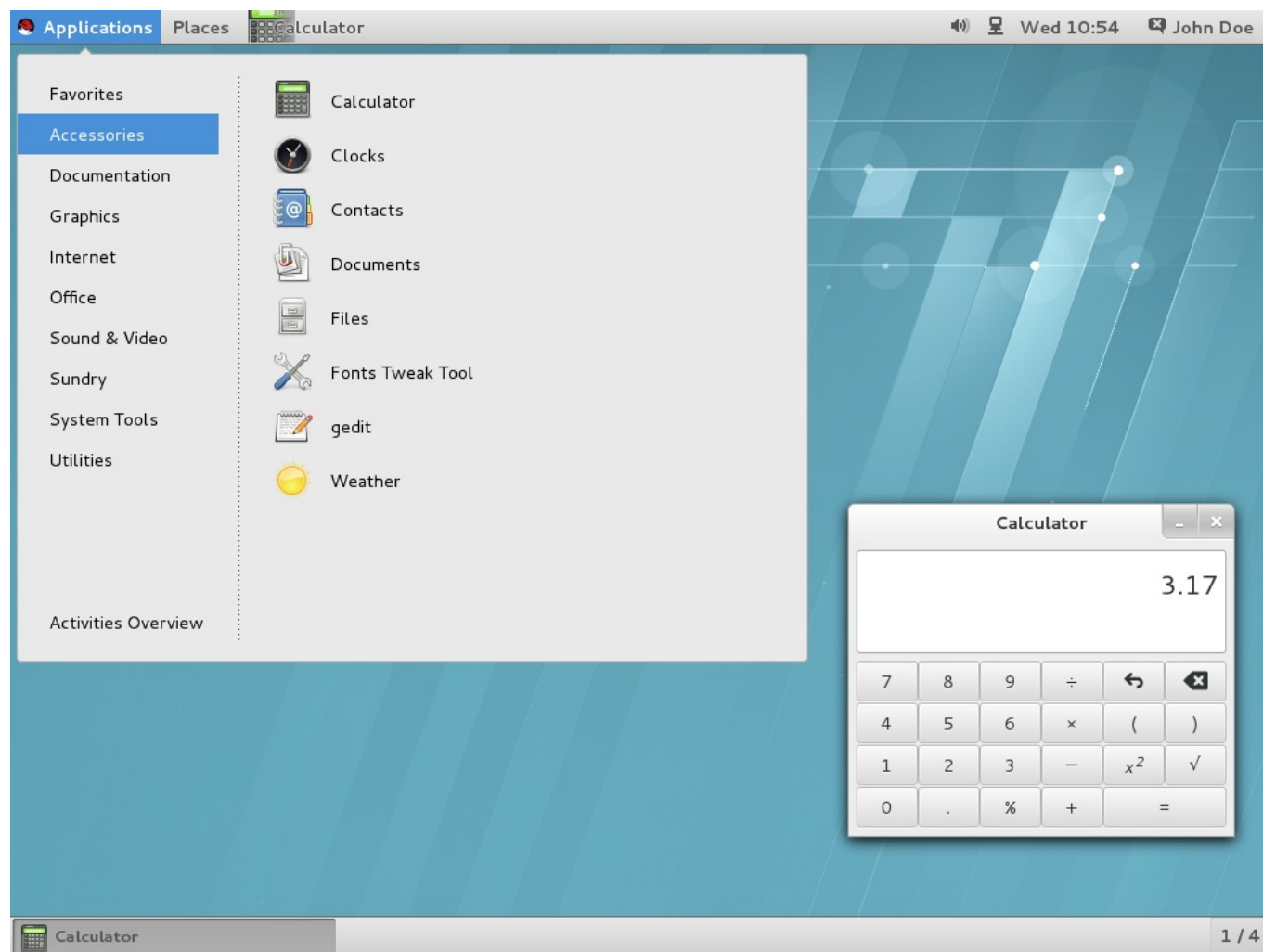


图 1.2. 配备计算机应用和应用程序菜单配件子菜单的 GNOME Classic

### 1.3.1. GNOME Classic 扩展

GNOME Classic 扩展被分配为一组〈[GNOME Shell extensions](#)〉。GNOME Classic 扩展被安装为 *gnome-classic-session* 软件包的从属品，它为启用 GNOME Classic 会话提供了所需组件。因为 GNOME Classic 扩展在红帽企业版 Linux 7 中是默认启用的，所以 GNOME Classic 是红帽企业版 Linux 7 的默认桌面用户界面。

- ✧ AlternateTab (“[alternate-tab@gnome-shell-extensions.gcampax.github.com](#)”)、
- ✧ 应用程序菜单 (“[apps-menu@gnome-shell-extensions.gcampax.github.com](#)”)、
- ✧ 启用新实例 (“[launch-new-instance@gnome-shell-extensions.gcampax.github.com](#)”)、
- ✧ 放置状态显示器 (“[places-menu@gnome-shell-extensions.gcampax.github.com](#)”)、
- ✧ 窗口列表 (“[window-list@gnome-shell-extensions.gcampax.github.com](#)”)。

### 1.3.2. GNOME Classic 与 GNOME 的相互转换

用户可以通过退出系统，在登录界面中的“会话”列表中选择 GNOME，从而将 GNOME Classic 切换到 GNOME。

执行以下命令，从而将 GNOME Classic 在用户会话中切换回 GNOME：

```
$ gnome-shell --mode=user -r &
```

执行以下命令，从而在同一个用户会话中切回到 GNOME Classic：

```
$ gnome-shell --mode=classic -r &
```

### 1.3.3. 禁用 GNOME Classic 为默认会话

对所有红帽企业版 Linux 7 中新创建的用户而言，GNOME Classic 是设定的默认会话。您需要在 `/var/lib/AccountsService/users/username` 文件中修改用户的账户服务来为某个特定用户覆盖这个默认设置。如何修改的细节内容请参见〈[第 14.3.2 节“配置用户默认会话”](#)〉。

## 1.4. 获取更多信息

用户可以在 GNOME 帮助中获得更多使用 GNOME 3、GNOME Shell 或者 GNOME Classic 的信息，这些信息是由 `gnome-user-docs` 软件包提供的。如需进入 GNOME 帮助，请按 **Super** 键进入“活动概览”，输入 **help**，然后按 **Enter** 键。

## 部分 I. 迁移规划

迁移规划主要集中于红帽企业版桌面默认环境到 GNOME2 桌面系统的迁移，以及红帽企业版 Linux 5 和 Linux 6 到 GNOME 3 的迁移。这个部分的指南逐一概述了某些成分的变化，并描述了具备的新特点。

这本指南仅适用于 GNOME 系统桌面环境的变化。关于红帽企业版Linux 7系统的变化，请参见：

- ✦ 关于诸如“**GRUB 2**”引导装载程序、软件包管理、“**systemd**”，或者计算机配置的部分，请参见《*红帽企业版Linux 7 系统管理员指南*》。
- ✦ 《*红帽企业版Linux 7系统迁移规划指南*》是对红帽企业版Linux 6系统和红帽企业版Linux 7系统两者之间的运行状况和兼容性的主要变化概览，同时也介绍了由红帽提供辅助升级至红帽企业版Linux 7 系统所需要的工具。
- ✦ 关于安装红帽企业版Linux 7 和使用**Anaconda**安装程序的详细信息，请参考《*红帽企业版Linux 7 系统安装指南*》。

这些文件可以在 [http://access.redhat.com/site/documentation/Red\\_Hat\\_Enterprise\\_Linux/](http://access.redhat.com/site/documentation/Red_Hat_Enterprise_Linux/) 网站查找到。

## 第 2 章 logind

“**logind**”（更确切的说，是“**systemd-logind**”）是管理用户登录的系统服务。这项服务将负责以下内容：

- ✧ 追踪用户和会话，其过程以及它们的空闲状态，
- ✧ 为用户进程创建控制组，
- ✧ 为用户提供以 PolicyKit 为基础的操作访问，比如系统关闭或休眠操作，
- ✧ 执行应用程序的关机／休眠抑制逻辑，
- ✧ 开机／休眠硬件钥匙的处理，
- ✧ 为用户提供 multi-seat（多座位）管理，会话切换管理，以及设备访问管理，
- ✧ 在虚拟终端（控制台）激活自动大量生成文本登录（getty），管理用户运行时目录。

“**logind**”服务是红帽企业版 Linux 7 中新的初始化系统，它取代了红帽企业版 Linux 6 中的“**upstart**”初始化系统，并与“**systemd**”紧密结合。这个变化带来了许多新特点和新功能。以下是对那些最重要的特点与功能的概述：

### ConsoleKit

“**ConsoleKit**”框架在红帽企业版 Linux 7 中是禁用的。“**systemd**”提供与之相同的功能。  
“**ConsoleKit**”和“**logind**”都是追踪当前运行用户会话的服务。



### 注意

“**ConsoleKit**”之前能够在系统中的活动会话改变时，运行任意的外壳脚本（使用虚拟终端切换）。现在将不再提供这个功能。

### /var/log/ConsoleKit/history 文件

“**ConsoleKit**”之前可以向 `/var/log/ConsoleKit/history` 发送日志文件，而现在的“**logind**”则不支持这个功能。这个文件被传统的追踪系统中所有登录和登出的 **wtmp** 文件和 **utmp** 文件所取代。`/var/log/ConsoleKit/history` 所提供的信息与 **wtmp** 文件所提供的信息基本相同，只是格式不同而已。考虑到功能上的重叠，“**logind**”仅仅扮演了“**wtmp**”文件的角色。

### seat.d 脚本

但是既然已不再使用“**ConsoleKit**”，则“**seat.d**”脚本语言也不再是“**ConsoleKit**”框架中的一部分，并已由“**systemd-logind**”替代。

### ck-list-sessions 命令

“**ConsoleKit**”提供了返回最近用户扩展信息的 **ck-list-sessions** 命令，不仅是普通用户而且包含对“**GDM**”进行 GUI 访问。要达到相似结果也可以运行“**loginctl**”命令。

```
$ loginctl list-sessions
```

### multi-seat 支持（多座位支持）

“**logind**”和“**GDM**”提供了使用户可以将其他显示器，鼠标，或者键盘添加到 **multi-seat** 的功能。这么做的话，就会出现一个额外的登录界面，用户在使用其他的设备时也可进行登录。

运行以下命令可以列出系统中可用的座位：

```
$ loginctl list-seats
```

运行以下命令可以显示系统中某个特定座位的状态：

```
$ loginctl seat-status seat
```

其中 *seat* 是座位的名称，比如“**seat0**”。

运行以下命令，为某个特别的座位指派特定的硬件：

```
# loginctl attach seat device
```

其中 *seat* 是座位的名称，比如“**seat1**”，同时 *device* 是用 **/sys** 设备路径指定的设备名称，比如 **/sys/devices/pci0000:00/0000:00:02.0/drm/card0**。

如果要更改任务，将硬件指定到另一个座位，或者使用 **loginctl flush-devices** 命令。

## 2.1. 获取更多信息

- ✦ **systemd-logind.service** (8) –“**logind**”的手册页提供了更多关于“**logind**”用量和特征的信息。它同时涵盖了“**systemd-logind**”提供的 API（logind D-Bus API 文件）。
- ✦ **logind.conf**(5) – 讨论登录管理员配置文件的 **logind.conf** 的手册页。
- ✦ **loginctl** (1) –“**systemd**”登录管理员的手册页包含了更多 multi-seat 特点的信息。

## 第 3 章 GSettings 和 dconf

红帽企业版 Linux 7 系统的主要改变之一是将“GConf”（作储存用户偏好用）转变为“GSettings”高级设置系统和“dconf”后端的相结合。除了作为后端，“dconf”同时也是将系统硬件和软件配置细节以单一、紧致、二进制格式储存起来的程序。

### GConf

如上所述，“GConf”配置系统由以下两个系统代替：“GSettings”API 和低级别配置系统“dconf”后端。

“gsettings”命令行工具和“dconf”实用工具两者都用来查看和变更用户设置。“dconf”实用工具使用“dconf-editor”GUI 来编辑配置数据库，而“gsettings”实用工具则会直接作用在终端。更多关于“dconf-editor”和“gsettings”实用工具的信息，请参见〈[第 9 章 用 GSettings 和 dconf 来配置桌面](#)〉。

### gconftool

“gconftool-2”工具已经由“gsettings”和“dconf”替代。同样，“gconf-editor”也已由“dconf-editor”替代。

### 重写

*keyfile*（关键文件）的概念在红帽 Linux 7 企业版中已有引用：**dconf** 实用工具允许程序管理员通过直接安装 *default overrides*（重写默认设置）来覆盖默认设置。例如，所有的用户现在可以通过执行关键文件目录，比如 `/etc/dconf/db/local.d/` 中的覆盖应用 **dconf** 来设置默认背景。想要了解更多有关默认值和覆盖设置的信息，请参见〈[第 9.5 节 “设置自定义默认值”](#)〉。

### 锁定设置

“dconf”系统现在允许锁定个人或者整个设置子路径，来防止用户自定义。更多如何锁定设置的信息，请参见〈[第 9.5.1 节 “锁定特定设置”](#)〉。

### NFS 和 dconf

使用在家目录中通过“NFS”共享的“dconf”实用程序，需要进行额外的设置。有关本专题的信息，请参见〈[第 9.7 节 “存储用户设置于 NFS”](#)〉。

## 3.1. 获取更多信息

更多如何使用 GSettings 和“dconf”配置用户设置的信息，请参见〈[第 9 章 用 GSettings 和 dconf 来配置桌面](#)〉。



## 第 4 章 PolicyKit

“**PolicyKit**”实用工具是一个框架，它提供了一个为无特权程序 *subjects* 提供服务的特权程序 (*mechanisms*) 使用的授权 API。“**PolicyKit**”的相关变化或者它的系统名称“**polkit**”所历经的细节改变如下。

### 4.1. 策略配置

就这些新的特点而言，它们的授权规则现在是在 *JavaScript.rules* 文件中定义的。这就意味着同样的文件被用来同时定义规则以及管理员状态。这些信息之前被储存在两个不同的文件类型—**.pkla** 和 **.conf** 中，它们通过使用键值对，来定义额外的本地授权。

这些新的 *.rules* 文件储存在两个位置；但是本地配置的“**polkit**”规则是储存在 **/etc/polkit-1/rules.d/** 目录下的，第三方软件包则储存在 **/usr/share/polkit-1/rules.d/** 中的。

目前的 **.conf** 和 **.pkla** 配置文件与 *.rules* 文件保存在一起，并同时存在。考虑到兼容性的问题，红帽企业版 Linux 7 中的“**polkit**”进行了升级。

规则的逻辑优先级也发生了变化。“**polkitd**”现在按照字母顺序从 **/etc/polkit-1/rules.d** 和 **/usr/share/polkit-1/rules.d** 目录中读取 *.rules* 文件。如果有两个名字相同的文件，相较于 **/usr** 中的文件，**/etc** 中的文件则将优先进行处理。除此之外，**/etc/polkit-1/rules.d/49-polkit-pkla-compatible.rules** 文件应用现有规则。因此，只要文件名在字母顺序中排在 49-polkit-pkla-compatible 之前，它们就可以由 *.rules* 在 **/usr** 或者 **/etc** 中重写。确定您的旧规则没被改写的最简单的方法就是命名所有其它 *.rules* 时，起始数字都大于 49。

以下是一个 *.rules* 文件的例子。它为 *storage* 群组创建了一个允许在系统设备中安装文件系统的规则。这个规则存储在 **/etc/polkit-1/rules.d/10-enable-mount.rules** 文件中：

#### 例 4.1. 允许在系统设备中安装文件系统

```
polkit.addRule(function(action, subject) {
    if (action.id == "org.freedesktop.udisks2.filesystem-mount-system" &&
        subject.isInGroup("storage")) {
        return polkit.Result.YES;
    }
});
```

更多信息请参见：

- ✧ **polkit(8)** – 描述 JavaScript 规则和优先规则的手册页。
- ✧ **pkla-admin-identities(8)** 和 **pkla-check-authorization(8)** – 分别是 **.conf** 和 **.pkla** 文件格式的资料说明手册页。

### 4.2. 默认策略

默认设置现在允许 *wheel* 组的成员使用它们自己的口令而不用请求 *root* 口令来对管理操作进行身份验证。默认策略被定义在 **/etc/polkit-1/rules.d/50-default.rules** 中。

在 GNOME Setting 的用户面板中，您可以为“**管理员**”配置一个账户。当您第一次在 **GNOME 初始设置** 中创建用户时，在默认情况下，您会创建一个“**管理员**”账户（*wheel* 组的一员）。





### 注意

当“**sudo**”用户被允许使用其他用户的安全权限去运行程序时，“**管理员**”将成为 **wheel** 组的一员去提供额外授权用户执行受限命令的特殊系统权限。

## 4.3. 脚本语言的检查权限

检查一个程序是否被授权操作的“**pkcheck**”实用工具，现在为 **--process** 选项提供新选项，以新的参数来指定程序，从而使命令更加安全。新格式如下所示：

```
--process pid,start-time,uid
```

**--process** 选项本身并不是新的，只有它的参数是新的：*pid*、*start\_time*，和 *uid* 可使用确切的值进行替换。在 **pkcheck(1)** 手册页提及的其它值依然存在，但被认为是不安全的。



### 重要

调用“**pkcheck**”的脚本语言需要使用新格式，来避免竞争条件的出现。

## 4.4. 扩展 polkit 配置

替换后端授权实现的支持已被删除。可以通过编写一个 JavaScript **.rules** 文件，来调用外部程序从而达到类似的作用。

红帽企业版 Linux 7 的“**polkit**”取消了对替换“**PolkitBackendActionLookup**”实现（用来为验证对话框提供数据的界面）的支持。

更多关于“**polkit**”的信息，请参见 **polkit (8)** 手册页。

## 第 5 章 GDM

“GDM”是“GNOME 显示管理器”，它提供了一个图形登录环境。在 GNOME 2 向 GNOME 3 的过渡之后，只有通过“systemd”才可以设置“GDM”，因为 GNOME 3 不再支持其他初始化系统。

### gdm 软件包

*gdm* 软件包取代了 *xorg-x11-xdm* 软件包，它为 X Window 系统提供了旧版的显示登录管理器。如前所述，*gdm* 软件包提供了图形登录屏幕，它在启动、登出，和用户切换不久之后则会显示。

### GDM 和 logind

“GDM”现在使用“logind”来定义和追踪用户。更多相关信息，请参见〈[第 2 章 logind](#)〉。系统管理员也可以在 GDM 自定义配置文件 `/etc/gdm/custom.conf` 中手动设置自动登录。

### custom.conf

GDM 设置现在可以在文件 `/etc/gdm/custom.conf` 中找到。但就反向兼容性而言，一旦发现 `/etc/gdm/custom.conf` 文件的话，`custom.conf` 文件就会被它取代。升级时，红帽建议您移除旧的 `gdm.conf` 文件并将所有的自定义设置迁移到 `custom.conf` 文件中。

## 5.1. 获取更多信息

更多关于“GDM”的信息，请参见〈[第 14.1 节 “何为 GDM？”](#)〉。

更多关于设置和管理用户会话的信息，请参见〈[第 14.3 节 “用户会话”](#)〉。

更多关于自定义登录屏幕外观的信息，请参见〈[第 10.5 节 “自定义登录屏幕”](#)〉。

## 第 6 章 GNOME Shell 扩展

红帽 Linux 7 企业版中的 GNOME Shell 不支持在红帽 Linux 5 和 6 企业版中用来自定义 GNOME 2 默认界面的小应用程序。GNOME 3 将小应用程序替换成了 *GNOME Shell 扩展*。此扩展可以更改 GNOME Shell 的默认界面，以及一些诸如窗口管理和程序启动的部分。

### 6.1. 时钟小程序的置换

红帽 Linux 5 和 6 企业版的 GNOME 2 的特点是 **时钟** 小应用程序，它在 GNOME 2 面板上提供对日期、时间和日历功能的访问。在红帽 Linux 7 企业版中，**时钟** 是由 *gnome-clocks* 软件包提供的，并替代了之前的那个小应用程序。用户可以通过点击 GNOME Shell 顶部菜单栏中的日历，选择“**打开时钟**”，从而对时钟进行访问。

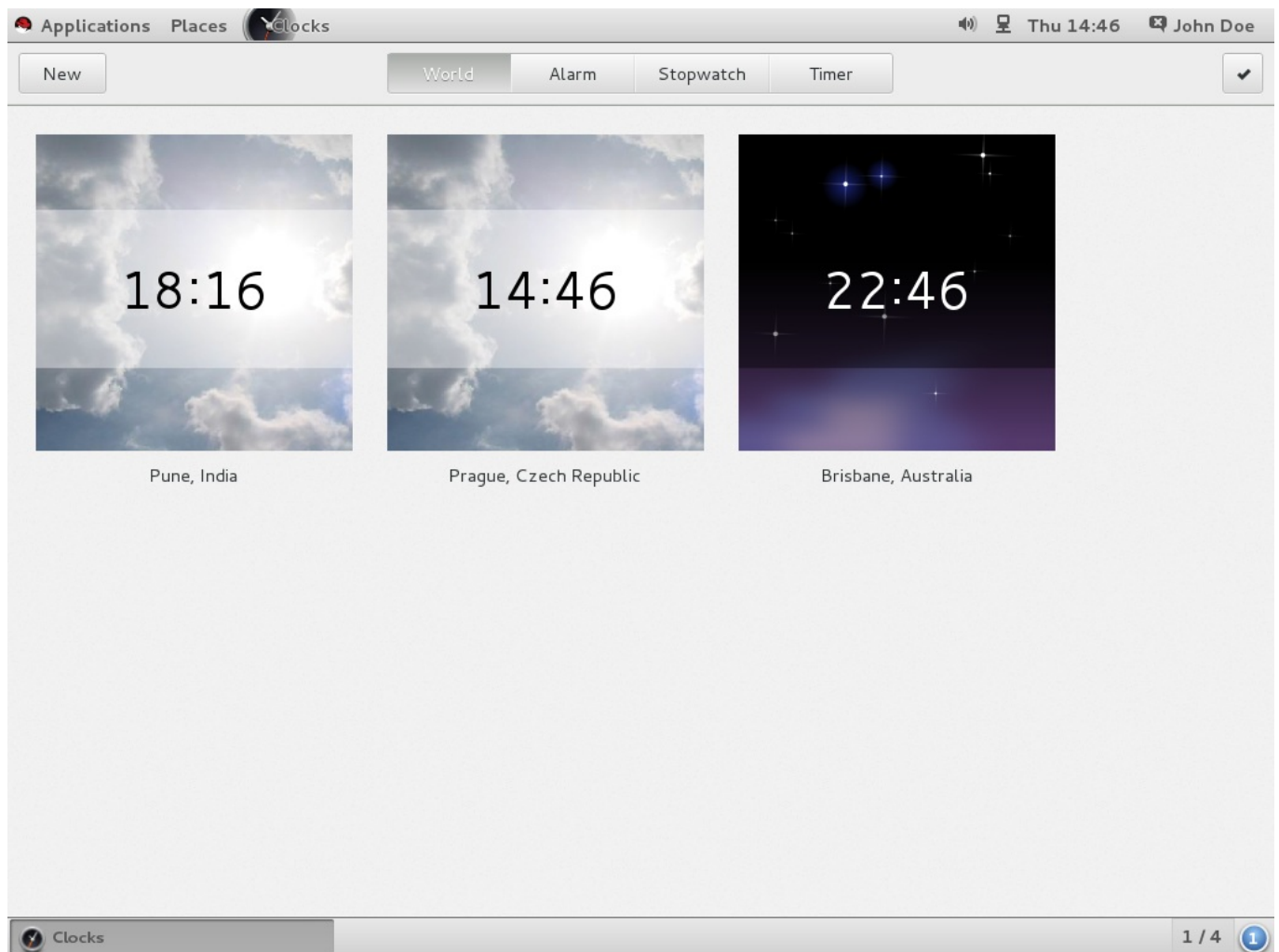


图 6.1. 打开时钟

#### 6.1.1. 获取更多信息

更多关于 GNOME Shell 扩展是什么，及如何对它们进行设置管理的信息，请参见 [〈第 11.1 节“GNOME Shell 扩展是什么？”〉](#)。

## 第 7 章 gnome-session

红帽企业版 Linux 7 中的“**gnome-session**”程序也进行了升级。它与之前启动 GNOME 桌面的方式一样，但它的某些部分发生了变化。

### gnome-session-properties

**gnome-session-properties** 应用依然是 *gnome-session* 软件包的一部分。但是，它的功能被限制为，为个人用户管理启动程序，以及在登出时保存当前运行的应用程序。后者在红帽企业版 Linux 6 中保留了下来。

### 被命名的会话

“**现在保存**”键能够在特定时间保存会话，并为之命名。保存的会话在登录时恢复。您在 **gnome-会话-特性** 中点击“**登出时自动记住正在运行的程序**”后，保存的应用列表也会在登录时显示出来。

有了这项更新，就可以创建多个布局并进行重命名，或者为一个用户账号选择多个用户会话。

## 7.1. 获取更多信息

有关会话管理的详细信息，请参见 [〈第 14 章 会话管理〉](#)。

关于如何为用户管理启动（自动运行）应用程序的信息，请参见 [〈第 14.3.5 节 “为所有用户添加自动启动的应用程序”〉](#)。

## 第 8 章 国际化

### 8.1. 输入法

红帽 Linux 7 企业版的中 GNOME 桌面默认的输入法框架是 *IBus*（智能输入总线）。它与 GNOME 3 结合并包含一个供用户选择的输入法界面。

#### 8.1.1. 输入法的配置与转换

用户可以使用 GNOME 设置中的“**区域 & 语言**”来配置输入法。更多使用输入法的信息可以在 GNOME 帮助中找到。具体方法如下，点击 **Super** 键，输入“**活动概览**”，键入 **help**，然后按 **Enter** 键。

对于非 GNOME 会话而言，IBus 可以同时配置 **ibus-setup** 工具中的 XKB 布局和输入法，并通过捷径在两者间进行切换。

转换输入源的默认捷径是 **Super+Space**。红帽企业版 Linux 6 中的捷径是 **Ctrl+Space**。

#### 8.1.2. IBus 预测输入法

“**ibus-typing-booster**”是 IBus 平台的预测输入法。它会根据部分输入的内容预测完整的词句，从而提供了更加快速更加准确的文本输入。用户可以在一系列建议列表中选择需要的字。“**ibus-typing-booster**”也可以使用 Hunspell 拼写检查器字典来为某个语种提供输入建议。

#### 8.1.3. GNOME 桌面中的 IBus 已取代 im-chooser

由于 IBus 现在已与 GNOME 桌面相结合，“**im-chooser**”仅在使用非 IBus 输入法时是可用的。

### 8.2. 文件位置的改变

在红帽企业版 Linux 7 中，对输入法和字体设置文件夹、目录的位置做了以下变更：

- ✧ **.xinputrc** 文件已从用户家目录转移到 **~/.config/imsettings/** 目录中。
- ✧ 已将 **.imsettings.log** 文件从用户家目录中转移至 **~/.cache/imsettings/log**。
- ✧ **~/.fonts.conf** 文件已弃用。建议用户将此文件转移到 **~/.config/fontconfig/** 文件目录下。
- ✧ **~/.fonts.conf.d** 目录已弃用。建议用户将此目录转移到 **~/.config/fontconfig/** 目录下。
- ✧ **/etc/fonts/conf.avail/** 目录中所有被禁用的“**fontconfig**”设置文件都被转移到 **/usr/share/fontconfig/conf.avail/** 目录下。如果您有任何指向原先位置的本地符号链接，请记得更新这些链接。

## 部分 II. 配置和管理

《红帽企业版 Linux 7 桌面系统迁移管理指南》的第二个部分描述并解释配置管理GNOME桌面系统的多种方法。

## 第 9 章 用 GSettings 和 dconf 来配置桌面

### 9.1. 术语解释：GSettings、gsettings，以及 dconf

本小节定义了一些容易混淆的术语。

#### dconf

“**dconf**”是管理用户设置，基于键值的配置系统。它是红帽企业版 Linux 7 使用的“**GSettings**”的后端。“**dconf**”管理了一系列不同的设置，包括“**GDM**”、应用程序，以及代理设置。

#### dconf

“**dconf**”命令行实用程序用来从“**dconf**”数据库中读取单个值或整个目录，以及将单独值或整个目录写入“**dconf**”。

#### GSettings

**GSettings** 是应用程序设置的高级 API，“**dconf**”的前端。

#### gsettings

使用“**gsettings**”命令行工具查看以及更改用户设置。

### 9.2. 用户和系统设置

“**dconf**”给予系统管理员和用户几个控制配置的级别。

- ✧ 管理员可以定义适用于所有用户的默认设置。
- ✧ 用户可以用他们自己的设置覆盖默认值。
- ✧ 管理员也可以选择性地锁定设置，以防止用户重写设置。更多信息请参见 [〈第 9.5.1 节 “锁定特定设置”〉](#)。

### 9.3. 浏览桌面应用的 GSettings 值

您可以使用两个工具来查看和编辑“**GSettings**”值：

- ✧ **dconf-editor** GUI 工具。
- ✧ **gsettings** 命令行实用程序。

**dconf-editor** 和 **gsettings** 实用程序可以让您浏览，以及更改系统和应用程序偏好的选项。您甚至可以用它们来更改那些图形用户界面中不存在的偏好设定。

**dconf-editor** 为浏览和编辑设置提供了一个 GUI。它以树视图的形式展现了设置的不同等级，并且显示了每一个设置的附加信息，包括简介、类型和默认值。**gsettings** 可以用来显示以及设置“**dconf**”值。它还包含命令和设置的 Bash 完成。**gsettings** 可以用来自动化 shell 脚本中的配置。

请注意，**dconf-editor** 和 **gsettings** 实用程序是为了浏览和更改当前用户的 **GSettings** 数据库。也就是说您始终都应应以普通用户的身份运行这些工具。

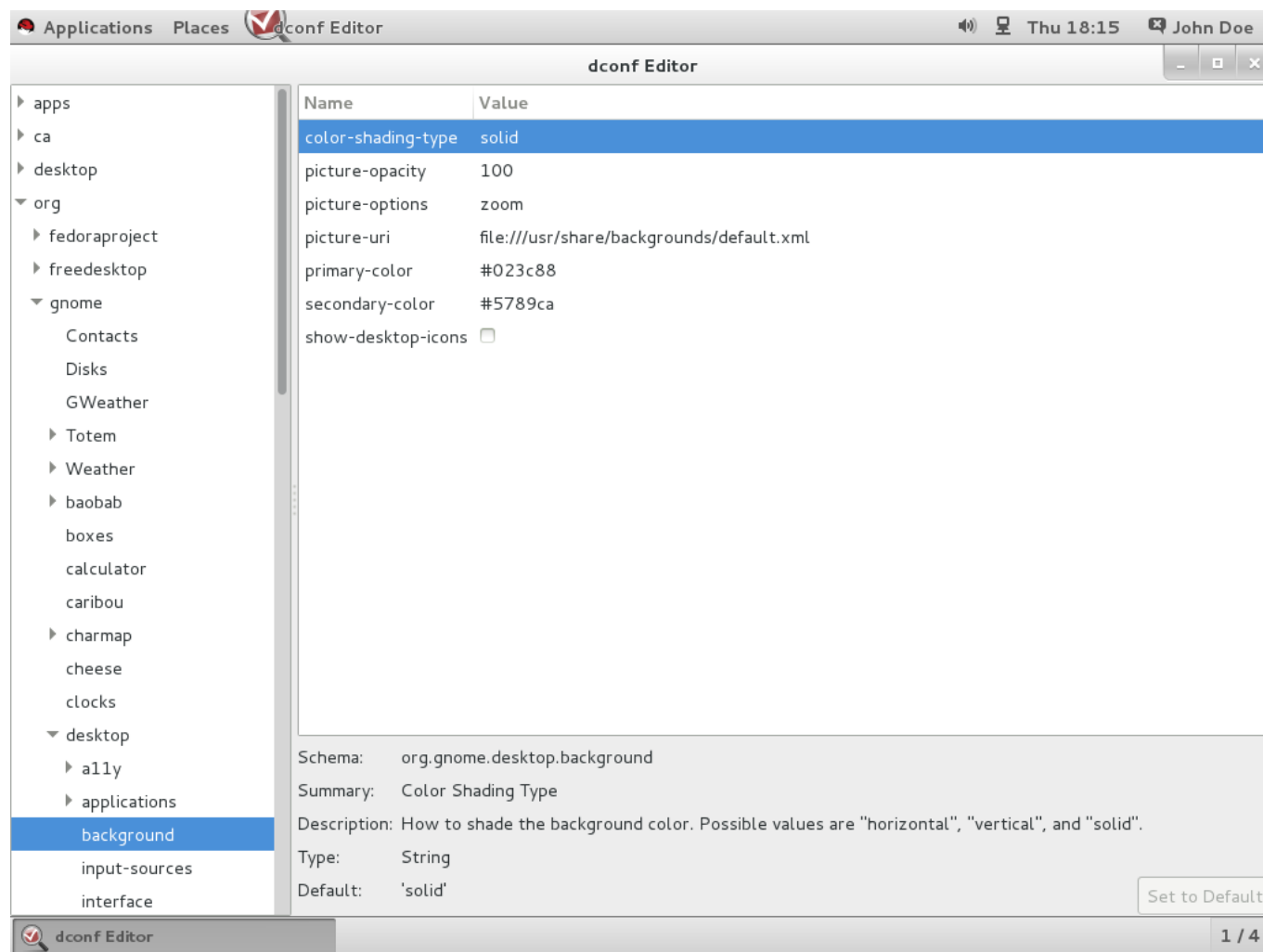


图 9.1. dconf-editor 显示 org.gnome.desktop.background GSettings 键值

您的系统中可能没有默认安装 **dconf-editor**。欲安装此应用，请运行以下命令：

```
# yum install dconf-editor
```

### 9.3.1. 获取更多信息

更多有关 **dconf-editor** 工具的信息，请参见 **dconf-editor(1)** 手册页。

更多有关 **gsettings** 实用程序的信息，请参见 **gsettings(1)** 手册页。

## 9.4. 何为 dconf 配置文件？

配置文件是系统软硬件配置的资料库列表，由“**dconf**”系统搜集而来。“**dconf**”配置文件允许您通过比较相同的系统来排除硬件或软件的故障问题。

“**dconf**”系统将它的配置文件存储在文本文件中。**\$DCONF\_PROFILE** 环境变量可以从 **/etc/dconf/profile/** 目录中，或者从绝对路径，比如用户家目录中给文件指定一个相对路径。

除非您设定的值出现了问题，否则设置于“**dconf**”“配置文件”中的键值对将会覆盖默认设置。

### 9.4.1. 选择 dconf 配置文件



启动时，“dconf”将访问环境变量 `$DCONF_PROFILE` 的变量是否已被设定。如果是的话，“dconf”将尝试打开被命名的配置文件，如果这一步失败了则将放弃此尝试。

只要环境变量未被设定，“dconf”就会尝试打开名为 `user` 的配置文件。如果这一步不成功，“dconf”将会退回到内部的硬连接配置中。

配置文件的每一列线路都指定了一个“dconf”数据库。第一列线路指示了写过更改的数据库，而其他的线路则只显示只读数据库。以下是存储在 `/etc/dconf/profile/user` 中的一个样本配置文件：

```
user-db:user
system-db:local
system-db:site
```

这个样本配置文件指定了三个数据库：`user` 是通常可在 `~/.config/dconf` 中找到的用户数据库的名称，`local` 和 `site` 是位于 `/etc/dconf/db/` 中的两个系统数据库。



### 重要

会话的“dconf”配置文件在登录时就已确定，因此用户需登出并重新登录，来应用一个新的“dconf”用户配置文件到他们的会话上。

## 9.5. 设置自定义默认值

计算机范围的默认设置可以通过给“dconf”配置文件中的键值提供默认值的方式来设定。用户可以重写这些默认值。

如要给键值设置默认值，“user”配置文件必须存在，并且必须在“dconf”数据库中为这个键值添加一个值。

### 例 9.1. 设置默认背景

1. 在 `/etc/dconf/profile/user` 中创建一个 `user` 配置文件。

```
user-db:user
system-db:local
```

`local` 是一个“dconf”数据库的名称。

2. 在 `/etc/dconf/db/local.d/01-background` 中为本地数据库创建一个包含以下默认设置的 `keyfile`：

```
# dconf path
[org/gnome/desktop/background]

# GSettings key names and their corresponding values
picture-uri='file:///usr/local/share/backgrounds/wallpaper.jpg'
picture-options='scaled'
primary-color='000000'
secondary-color='FFFFFF'
```

在默认设置“关键文件”的过程中，会使用到以下 GSettings 键值：

表 9.1. org.gnome.desktop.background schemas GSettings 键值

键值名	可能值	描述
picture-options (图片选项)	“none”、“wallpaper”、“centered”、“scaled”、“stretched”、“zoom”、“spanned”	决定如何渲染由 <i>wallpaper_filename</i> 设置的图片。
picture-uri (图片路径)	具备路径的文件名	背景图片使用的 URI。请注意，后端仅支持本地 (file://) URI。
primary-color (主要色)	默认值：000000	作渐变梯度或者纯色时的左侧或顶部颜色。
secondary-color (次要色)	默认值：FFFFFF	作渐变梯度时的右侧或底部色彩，不可用于纯色。

- 3. 根据喜好编辑“关键文件”。更多信息请参见〈[第 9.3 节 “浏览桌面应用的 GSettings 值”](#)〉。
- 4. 更新系统数据库：

```
# dconf update
```

 **重要**


当创建或更改“用户”文件后，用户需在应用更改之前登出并重新登录。

如果您想避免创建 **user** 文件，您可以使用 **dconf** 命令行实用工具来向“**dconf**”数据库读取写入单独值或者整个目录。更多信息请参见 **dconf** (1) 手册页。

9.5.1. 锁定特定设置

“**dconf**”中的锁定模式是非常有用的工具，用来防止用户更改特定设置。

如果要锁定一个“**GSettings**”键值，您需要在关键文件目录中创建一个 **locks** 子目录（例如，**/etc/dconf/db/local.d/locks/**）。这个目录中的文件包含了一列需锁定的键值，您可以在此目录中添加多个文件。

 **重要**

如果您不通过使用锁定来加强系统设置，用户就能够很容易地用他们自己的设置覆盖系统设置。除非有一个加强系统设置的锁定，否则的话用户设定的任何设置将会优先于系统设置。

下面的例子演示了如何锁定默认墙纸的设置。您可以按照这个步骤锁定任何需要锁定的设置。

**例 9.2. 如何锁定默认墙纸**

- 1. 设定一张默认墙纸。

2. 创建一个名为 `/etc/dconf/db/local.d/locks/` 的目录。
3. 编辑 `/etc/dconf/db/local.d/locks/00-default-wallpaper`，并在每行列出一个键值：

```
# Prevent users from changing values for the following keys:
/org/gnome/desktop/background/picture-uri
/org/gnome/desktop/background/picture-options
/org/gnome/desktop/background/primary-color
/org/gnome/desktop/background/secondary-color
```

4. 更新系统数据库：

```
# dconf update
```

## 9.6. GSettings 键值属性

您只能在“**dconf**”数据库中设置一次 GSettings 键值。如果您在“**dconf**”数据库中不同的地方为不同的值设定了相同的键值，那么它们中只有一个能生效。换句话说，您将会以这个键值设置覆盖其它的键值设置。

在每一个“**dconf**”系统数据库中，每一个键值只能拥有一个值。但需要注意的是，对于键值而言，值是以数组存在的。拥有数组类型的键值有多个值，它们被指定在如下所示，并以逗号区分的列表中。

```
key=['option1', 'option2']
```

### 例 9.3. org.gnome.desktop.input-sources.xkb-options GSettings 键值

设置 **org.gnome.desktop.input-sources.xkb-options** GSettings 键值的方法如下所示。它只能被设定一次，因此如果您想要一个值中存在两个元素，您需要将它们指定于同一个配置文件中。因为这是一个阵列类型的值，所以它可以有许多元素，从而即使设定多个键值也可以正常工作。

```
[org/gnome/desktop/input-sources]
# Enable Ctrl-Alt-Backspace for all users
# Set the Right Alt key as the Compose key and enable it
xkb-options=['terminate:ctrl_alt_bksp', 'compose:ralt']
```

## 9.7. 存储用户设置于 NFS

为了使“**dconf**”在使用“**网络文件系统**”（“**NFS**”）家目录时能够正常运行，就必须使用“**dconf**”“**Keyfile 后端**”。

需注意的是，在使用“**dconf**”“**Keyfile 后端**”前，必须将 *glib2-fam* 软件包安装到系统中。否则有关远程计算机设置更改的通知将无法正常运行。

### 过程 9.1. 设置 dconf 键值文件后端

1. 验证 *glib2-fam* 软件包已安装在系统中。
  - a. 系统必须订阅 **Optional** 通道。想要了解系统如何订阅 **Optional** 通道，请阅读以下资源：

➤ <https://access.redhat.com/site/solutions/70019>

b. 运行以下命令，来安装 *glib2-fam* 软件包：

```
# yum install glib2-fam
```

2. 在每一个客户端创建或编辑 `/etc/dconf/profile/user` 文件。
3. 在这个文件的开头放置一行 `service-db:keyfile/user`。
4. 保存修改。

“**dconf**”“*Keyfile 后端*”只能在用户下次登录时生效。它将轮询关键文件以确定更新是否被执行，这样一来设置可能不会立即更新。

## 第 10 章 默认外观

个人用户和系统管理员都可以为所有用户自定义设置 GNOME 3 桌面的外观和功能。系统管理员可以为所有用户提供自定义的默认桌面配置，甚至可以为了防止用户更改配置而关闭这些特性。本章讲述了如何自定义安装实用程序（“anaconda”）、系统启动实用程序（“firstboot”、“GRUB”，和 “Plymouth”）、登录屏幕、字体、键盘布局、帘栅屏蔽，以及桌面背景。

### 10.1. 在 Anaconda 中置入徽标

如果您要建立自己公司的发行版，那么您可以改变红帽企业版 Linux 7 安装程序，“Anaconda”。默认情况下，“Anaconda”只会显示运行系统版本的标志，如红帽 “Shadow Man” 的标志，以及 RPM 标志。

尽管“Anaconda”的图片相对较少，但也还是有定制化的空间。您可以更改：

- ✧ 安装程序映像。
- ✧ 产品名称。

#### 过程 10.1. 更改安装程序映像

1. 创建一个 “*product.img*” 文件（计算机文件系统映像）。如何在红帽企业版 Linux 7 中创建此文件的详细信息，请参见《[安装指南](#)》。
2. 于此映像中创建 `pixmaps/rnotes/language_code/` 目录，并将您的映像放入其中。如果不支持多方定位（localisation, not location）的话，您还可以使用 `pixmaps/rnotes/` 目录。否则您必须把映像放到正确的 `language_code` 目录下。

安装过程中，您的映像将会沿屏幕下方大约每分钟循环一次。

#### 过程 10.2. 更改产品名称

1. 按照以下设计的基本格式创建一个 `.buildstamp` 文件：

```
[Main]
Product=Name of Product Here
Version=Version Number Here
```

2. 将 `.buildstamp` 文件以最高级别保存在 `product.img` 文件中 – 请参见〈[过程 10.1, “更改安装程序映像”](#)〉。
3. Anaconda 在 `images/` 子目录中的安装源中寻找 `product.img` 并编辑产品名称。

指定的产品名称和版本会在安装过程中显示。

### 10.2. 在 Firstboot 中置入徽标

“Firstboot”会在您首次启动红帽企业版 Linux 7 并登录之前启动。在图像模式中，您可以更改出现在“Firstboot”欢迎屏幕中的徽标。

1. 用 .png 格式的新徽标创建一个图像文件。
2. 将它保存到 `/usr/share/firstboot/themes/default/` 目录下。

另一个方法是，创建一个带有徽标图像的新目录，并使用运行“**Firstboot**”时，指向新创建目录的“**/firstboot/systemd/firstboot-graphical.service**”配置项目的 **--themedir** 选项。

然而，在首次引导系统前创建目录是必要的，比如，“**Kickstart**”中的“**%post**”小节。更多详细信息，请参见《[红帽企业版 Linux 7 安装指南](#) 安装 Kickstart》。

## 10.3. 引导装载程序屏幕

红帽企业版 Linux 7 的引导装载程序是“**GRUB 2**”。您可以更改“**GRUB 2**”外观的几个部分。以下几小节将向您展示如何改变 Linux 发行版名称、菜单颜色，和背景图片。

### 10.3.1. 发行版名称

默认情况下，“**GRUB 2**”显示含有发行版名称的标题。您可以通过自定义 **/etc/default/grub** 文件中的 **GRUB\_DISTRIBUTOR** 变量，从而更改标题。

#### 过程 10.3. 设置发行版名称

1. 以 root 身份，打开 **/etc/default/grub** 文件。
2. 使用 **GRUB\_DISTRIBUTOR** 变量，来指定您自己的发行版名称。以下是 **/etc/default/grub** 文件的摘录。使用 **GRUB\_DISTRIBUTOR** 变量来更新第二行：

```
GRUB_TIMEOUT=5
GRUB_DISTRIBUTOR=Our Corporate Distro V1.2
GRUB_DEFAULT=saved
GRUB_DISABLE_SUBMENU=true
...
```

3. 以 root 身份运行以下命令，以确定更改生效：

```
grub2-mkconfig -o /boot/grub2/grub.cfg
```



#### 注意

您需要在每次做更改后更新 **/etc/default/grub** 目录。

同时，您可以通过更改 **/etc/grub.d/40\_custom** 纯文字档，或者通过在 **/etc/grub.d/** 目录中添加其他文件，来改变“**GRUB 2**”屏幕中显示的文本颜色或字体。您可以从以下指令中选择：

- ✧ **set color\_normal=foreground/background**
- ✧ **set color\_highlight=foreground/background**
- ✧ **set menu\_color\_normal=foreground/background**
- ✧ **set menu\_color\_highlight=foreground/background**

更多有关这些变量的语义和接受值的信息，请参见 **grub(8)** 手册页。

### 10.3.2. GRUB 2 背景

在默认设置中并没有配置“GRUB”背景。然而，您依然可以添加图片到引导程序屏幕。

在您为“GRUB 2”配置背景图片前，必须为“gfxterm”图形终端安装一个“GRUB 2”Unicode 字体。因为没有默认提供的字体，所以前提是将现有的 TTF 或者 OTF 文件转换成“GRUB 2”使用的 PF2 格式：

通过运行 **grub2-mkfont** 命令，来将现有的 TTF 或者 OTF 文件转换成 PF2 格式。为输出文件 **unicode.pf2** 命名，从而使之能与 **grub2-mkconfig** 写入的默认配置正常运行。

### 例 10.1. 将 TTF 文件转换到 PF2 格式

这个例子演示了从 **LiberationSerif-Bold.ttf** 到 .pf2 的转换。新的 .pf2 格式文件被命名为 **unicode2**，从而使它不与现有的 **/grub2/fonts/** 目录下的 **unicode.pf2** 产生混淆。

```
grub2-mkfont --output=/boot/grub2/fonts/unicode2.pf2 --size=24
/usr/share/fonts/liberation/LiberationSerif-Bold.ttf
```

现在，您可以为“GRUB 2”设置背景图片。图片文件可以位于 **boot/** 目录之外。支持的格式有 PNG、JPG、JPEG，以及 TGA。

### 过程 10.4. 为引导装载程序屏幕添加图片

1. 以 root 身份打开 **/etc/default/grub** 文件。
2. 编辑文件中的以下配置变量：

```
GRUB_TERMINAL=gfxterm
```

```
GRUB_BACKGROUND=path_to_the_image.png
```

启动时这个选项的值必须是 GRUB 可读的文件。

3. 运行 **grub2-mkconfig -o /boot/grub2/grub.cfg**，将背景写入配置文件。
4. 重启系统。

必要的话，图片会按比例缩放来适合屏幕的大小。

## 10.4. Plymouth

“Plymouth”是红帽企业版 Linux 7 的图形启动系统和记录器，它使用了基于内核的模式设置（KMS）和直接渲染管理器（DRM）。在启动时，“Plymouth”将对用户交互进行处理。

您可以在多个静态或动态图形主题中选择主题，从而自定义启动画面外观。新的主题可以在现有主题的基础上进行创建。

### 10.4.1. 在主题中置入徽标

“Plymouth”的每一个主题都由一个主题数据文件和一个编译的“启动模块插件”组成。这个数据文件有一个安装于 **/usr/share/plymouth/themes/** 目录下的 **.plymouth** 扩展文件。



配置数据以“键 - 值” (key-value) 的格式指定于 **[Plymouth Theme]** 小节中。这个组的有效键值是“Name (名称) ”、“Description (描述) ”，和“ModuleName (模块名称) ”。其中，前两个键值自我解释的，第三个键值则指定了“Plymouth”启动插件模块的名称。不同的插件提供了开机时不同的动画和不同主题底层的实现：

### 例 10.2. 实例 .plymouth 文件

```
[Plymouth Theme]
Name=Charge
Description=A theme that features the shadowy hull of my logo charge up
and finally burst into full form.
ModuleName=two-step
```

### 过程 10.5. 更改 Plymouth 主题

1. 搜索现有的“Plymouth”主题，选择一个您最满意的。运行以下命令：

```
# yum search plymouth-theme
```

或者运行 **plymouth-set-default-theme --list** 命令，来查看所安装的主题。

您还可以在安装所有的 *plymouth* 软件包的同时，将所有主题进行安装。但与此同时，您也会安装许多不必要的软件包。

```
# yum install plymouth\*
```

2. 使用 **plymouth-set-default-theme theme\_name** 命令，将新主题设置为默认主题。

#### 例 10.3. 将 “spinfinity” 设置为默认主题

您已经选择了 *spinfinity* 主题，接下来请运行：

```
# plymouth-set-default-theme spinfinity
```

3. 编辑完成后，重新生成“**initrd**”守护程序，否则您设置的主题将不会显示在启动画面上。您可以运行以下命令重新生成守护程序：

```
# dracut -f
```

### 10.4.2. 创建一个新的 Plymouth 主题

如果您不想从给出的主题列表中选择主题，那么您可以创建属于您自己的主题。最简单的方法就是复制一个现有的主题并对它进行修改。

### 过程 10.6. 用一个现有的主题来创建您自己的主题

1. 复制 **plymouth/** 目录中的所有内容。作为一个目录模板使用，例如，通过以下设置来使用“**two-step**”启动插件的红帽企业版 Linux 7 的默认主题，**/usr/share/plymouth/themes/charge/charge.plymouth**：



```
[Plymouth Theme]
Name=Charge
Description=A theme that features the shadowy hull of my logo charge
up and finally burst into full form.
ModuleName=two-step

[two-step]
ImageDir=/usr/share/plymouth/themes/charge
HorizontalAlignment=.5
VerticalAlignment=.5
Transition=none
TransitionDuration=0.0
BackgroundStartColor=0x202020
BackgroundEndColor=0x202020
```

2. 将 **charge.plymouth** 文件重命名为以下格式，并保存在 **/usr/share/plymouth/themes/newtheme/** 目录下：

```
newtheme.plymouth
```

3. 根据您的喜好、颜色改变、对齐，或者转换，来更新您的 **/usr/share/plymouth/themes/newtheme/newtheme.plymouth** 文件。
4. 运行以下命令，从而将您的 *newtheme* 设置成默认主题。

```
# plymouth-set-default-theme newtheme
```

5. 改变主题后运行以下命令，从而重建“**initrd**”守护程序：

```
# dracut -f
```

#### 10.4.2.1. 使用品牌标商标

一些插件会将品牌商标作为启动动画的一部分显示。如果您想将您自己的徽标加到您的主题中，请遵循以下简短的步骤。



#### 重要

记住，您的品牌徽标图案的格式必须为 *.png* 格式。

#### 过程 10.7. 将您的徽标添加至主题

1. 为您的徽标创建一个名为 **logo.png** 的图形文件。
2. 通过更新 *ImageDir* 键值来编辑 **/usr/share/plymouth/themes/newtheme.plymouth** 文件，从而使之指向您在步骤一中创建的 **logo.png** 图形文件目录：

```
ImageDir=/usr/share/plymouth/themes/newtheme
```

更多关于“**Plymouth**”的信息，请参见 **plymouth** (8) 手册页。

## 10.5. 自定义登录屏幕

GNOME 登录屏幕有一些可以自定义设置的部分。这些更改只能由系统管理员执行并将作用于所有用户。本小节讲述了如何自定义登入文本、徽标、键盘布局，以及用户列表。要注意的是，登录屏幕背景图片不可进行自定义设置。

### 10.5.1. 添加欢迎程序徽标

登录界面上的欢迎程序徽标由 `org.gnome.login-screen.logo` GSettings 键值管理。由于“GDM”使用它自己的“`dconf`”配置文件，您可以通过在此配置文件中更改设置，来添加欢迎程序徽标。

更多关于 GSetting 和“`dconf`”的信息，请参见 [〈第 9 章 用 GSettings 和 dconf 来配置桌面〉](#)。

当为您的登录界面徽标选择合适的图片时，请将以下图片要求纳入考虑：

- ✱ 支持大多数的格式：ANI、BPM、GIF、ICNS、ICO、JPEG、JPEG 2000、PCX、PNM、PBM、PGM、PPM、GTIFF、RAS、TGA、TIFF、XBM、WBMP、XPM，和 SVG。
- ✱ 图片大小按比例缩放到 48 像素高度。因此，举例来说，如果您将徽标设置成 1920x1080 的大小，它将成为原图 85x48 的缩略图。

#### 过程 10.8. 为登录界面添加徽标

1. 在 `/etc/dconf/db/gdm.d/01-logo` 中以计算机范围设置创建一个 `gdm` 数据库：

```
[org/gnome/login-screen]
logo='/usr/share/pixmaps/logo/greeter-logo.png'
```

将 `/usr/share/pixmaps/logo/greeter-logo.png` 更换成您想要用作欢迎程序徽标的图像文件路径。

2. 更新系统数据库：

```
# dconf update
```

下次登录时，屏幕将显示新的登录徽标。

#### 10.5.1.1. 如果徽标没有更新怎么办？

确保您已经以 `root` 身份运行 `dconf update` 命令，从而更新系统数据库。

如若徽标没有更新，请尝试重启“GDM”。更多信息，请参见 [〈第 14.1.1 节“重启 GDM”〉](#)。

### 10.5.2. 显示文本标语

登录界面的文本条由以下 GSettings 键值文件管理（更多有关 GSettings 的信息，请参见 [〈第 9 章 用 GSettings 和 dconf 来配置桌面〉](#)）：

`org.gnome.login-screen.banner-message-enable`

启用标语信息显示。

`org.gnome.login-screen.banner-message-text`

在登录窗口显示文本标语信息。

需要注意的是，由于“GDM”使用它自己的“dconf”配置文件，因此您可以更改此配置文件中的设置，从而配置文本标语。

### 过程 10.9. 在登录界面显示文本标语

1. 以计算机范围设置在 `/etc/dconf/db/gdm.d/01-banner-message` 中创建一个 `gdm` 数据库：

```
[org/gnome/login-screen]
banner-message-enable=true
banner-message-text='Type the banner message here'
```

2. 更新系统数据库：

```
# dconf update
```

标语文本会在您从用户列表中选择了您自己的用户后，或当您开始在文本框中键入文本时出现。您将会在下次登录输入密码时看到标语文本。



#### 重要

将标语信息控制在合理的长度，因为文本太长屏幕可能装不下。

#### 10.5.2.1. 如果标语信息不更新怎么办？

如果标语信息不能显示的话，请确保您已经运行了 `dconf update` 命令。

如果标语信息没有更新的话，尝试重启“GDM”。更多信息请参见〈[第 14.1.1 节“重启 GDM”](#)〉。

#### 10.5.3. 显示多种键盘布局

您可以在登录屏幕上为用户添加其它以供选择的键盘布局。

这对通常使用默认设置以外的键盘布局的用户，以及想在登录屏幕上使用其它那些键盘布局的用户来说会很有帮助。然而，只有在使用登录界面时，这个选择才可用。一旦登录后，您自己的用户设置将会接替其它设置。

### 过程 10.10. 更改系统键盘布局设置

1. 在源代码中一节名为 `! layout` 中的 `/usr/share/X11/xkb/rules/base.lst` 文件内，找到想要的键盘布局代码。
2. 使用 `localectl` 工具，从而更改以下系统键盘布局：

```
localectl set-x11-keymap layout
```

您可以通过一个以逗号分隔的列表指定多个布局。例如，运行以下命令将 `es` 设置为默认布局，将 `us` 设置成第二默认布局：

```
$ localectl set-x11-keymap es,us
```

3. 登出后，即可在登录屏幕的顶部导航栏找到已定义的可用布局。

值得注意的是，您同时可以使用 **localectl** 工具，来指定计算机范围内的默认键盘模式、变量和选项。更多信息请参见 **localectl(1)** 手册页。

#### 10.5.4. 禁用登录屏幕用户列表

您可以通过设置 **org.gnome.login-screen.disable-user-list** GSettings 键值文件，来禁用显示在登录屏幕上的用户列表。

禁用用户列表后，用户需要在登录时的提示窗口中输入他们的用户名和密码。

##### 过程 10.11. 设置 **org.gnome.login-screen.disable-user-list** 键值文件

1. 在 **/etc/dconf/db/gdm.d/00-login-screen** 中，以计算机范围设置创建一个 **gdm** 数据库：

```
[org/gnome/login-screen]
# Do not show the user list
disable-user-list=true
```

2. 通过更新“**dconf**”实用工具来更新系统数据库：

```
# dconf update
```

## 10.6. 自定义桌面背景

您可以通过使用“**dconf**”实用程序设置默认背景、添加额外背景，或者添加多个背景。

如果不允许系统用户更改这些默认的设置，那么系统管理员需要使用锁目录来锁定设置。否则每一个用户都将能够自定义设置来符合他们的喜好。更多信息请参见 [〈第 9.5.1 节“锁定特定设置”〉](#)。

### 10.6.1. 自定义默认桌面背景

您可以通过在 **org.gnome.desktop.background** 架构中设置相关的 GSettings 键值文件，来设置默认桌面背景。

更多有关 GSettings 的信息，请参见 [〈第 9 章 用 GSettings 和 dconf 来配置桌面〉](#)。

##### 过程 10.12. 设置默认背景

1. 在 **/etc/dconf/db/local.d/filename** 中创建一个用于计算机范围设置的 **local** 数据库：

```
# Specify the dconf path
[org/gnome/desktop/background]

# Specify the path to the desktop background image file
picture-uri='file:///usr/local/share/backgrounds/wallpaper.jpg'
# Specify one of the rendering options for the background image:
# 'none', 'wallpaper', 'centered', 'scaled', 'stretched', 'zoom',
# 'spanned'
picture-options='scaled'
# Specify the left or top color when drawing gradients or the solid
```

```
color
primary-color='000000'
# Specify the right or bottom color when drawing gradients
secondary-color='FFFFFF'
```

2. 覆盖用户设置，以避免用户在 `/etc/dconf/db/local.d/locks/background` 中对此修改：

```
# List the keys used to configure the desktop background
/org/gnome/desktop/background/picture-uri
/org/gnome/desktop/background/picture-options
/org/gnome/desktop/background/primary-color
/org/gnome/desktop/background/secondary-color
```

3. 更新系统数据库：

```
# dconf update
```

在 `/etc/dconf/db/local.d/filename` 文件中配置的桌面为您新的默认桌面。

### 10.6.2. 添加额外背景

您可以为系统上的用户增加额外背景。

1. 用 `org.gnome.desktop.background schemas` 创建一个 `filename.xml` 文件来指定您的额外背景的外观。这是最常用模式的列表：

表 10.1. `org.gnome.desktop.background schemas` GSettings 键值

键值名称	可能值	描述
picture-options	“none”，“wallpaper”，“centered”，“scaled”，“stretched”，“zoom”，“spanned”	决定由 <code>wallpaper_filename</code> 设置的图片如何渲染。
color-shading-type	“horizontal”、“vertical”和“solid”	如何为背景颜色添加底纹。
primary-color	默认：#023c88	当绘制渐变色或者单色时，左侧或者右侧的颜色。
secondary-color	默认：#5789ca	当绘制渐变色时的右侧或者底部色彩，不适用于单色。

所有的图片选项可以在“`dconf-editor`”GUI 或者在“`gsettings`”命令行实用工具中找到。更多信息，请参见〈[第 9.3 节“浏览桌面应用的 GSettings 值”](#)〉。

2. 将 `filename.xml` 文件储存在 `/usr/share/gnome-background-properties/` 目录中。

当用户在右上角点击他们的名字时，选择“设置”，然后在“个人”小节中选择“背景”，用户就可以使用新的背景。

请看实例并了解实际上是如何对 `org.gnome.desktop.background` GSettings 键值文件进行操作的：

#### 例 10.4. 额外背景文件

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE wallpapers SYSTEM "gnome-wp-list.dtd">
```

```
<wallpapers>
  <wallpaper deleted="false">
    <name>Company Background</name>
    <name xml:lang="de">Firmenhintergrund</name>
    <filename>/usr/local/share/backgrounds/company-
wallpaper.jpg</filename>
    <options>zoom</options>
    <shade_type>solid</shade_type>
    <pcolor>#ffffff</pcolor>
    <scolor>#000000</scolor>
  </wallpaper>
</wallpapers>
```

在一个配置文件中，您可以指定多个 **<wallpaper>** 元素来添加更多的背景。

请看以下拥有两个 **<wallpaper>** 元素的 .xml 文件实例，添加两个不同的背景：

#### 例 10.5. 拥有两个墙纸元素的额外背景文件

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE wallpapers SYSTEM "gnome-wp-list.dtd">
<wallpapers>
  <wallpaper deleted="false">
    <name>Company Background</name>
    <name xml:lang="de">Firmenhintergrund</name>
    <filename>/usr/local/share/backgrounds/company-
wallpaper.jpg</filename>
    <options>zoom</options>
    <shade_type>solid</shade_type>
    <pcolor>#ffffff</pcolor>
    <scolor>#000000</scolor>
  </wallpaper>
  <wallpaper deleted="false">
    <name>Company Background 2</name>
    <name xml:lang="de">Firmenhintergrund 2</name>
    <filename>/usr/local/share/backgrounds/company-wallpaper-
2.jpg</filename>
    <options>zoom</options>
    <shade_type>solid</shade_type>
    <pcolor>#ff0000</pcolor>
    <scolor>#00ffff</scolor>
  </wallpaper>
</wallpapers>
```

### 10.6.3. 设置帘栅屏蔽

**帘栅屏蔽** (Screen Shield) 是系统暂停时迅速滑下的屏幕。它由 **org.gnome.desktop.background.picture-uri** GSettings 键值所控制。由于“GDM”使用它自己的“dconf”配置文件，所以您可以通过更改此配置文件的设置，来设置默认背景。

更多有关 GSettings 和“dconf”的信息，请参见 [〈第 9 章 用 GSettings 和 dconf 来配置桌面〉](#)。

### 过程 10.13. 为帘栅屏幕添加徽标

1. 在 `/etc/dconf/db/gdm.d/01-corp-login` 中，以计算机范围设置创建一个 `gdm` 数据库：

```
[org/gnome/desktop/background]
picture-uri='file:///opt/corp/background.jpg'
```

用您想用作帘栅屏幕的图像文件途径去代替 `/opt/corp/background.jpg`。

支持的格式为 PNG、JPG、JPEG，和 TGA。必要时图像将被缩放以适应屏幕。

2. 更新系统数据库：

```
# dconf update
```

您下次登录时，新的帘栅屏幕将出现在背景中。时间、日期和周几这些信息会显示在前景中。

#### 10.6.3.1. 如果帘栅屏幕没有更新怎么办？

请确定您已经以 `root` 身份运行 `dconf update` 命令更新数据库。

如背景未更新，请尝试重启“GDM”。更多信息请参见〈[第 14.1.1 节 “重启 GDM”](#)〉。

## 10.7. 设置字体

作为红帽企业版 Linux 7 的组成部分，“**字体设置**”是设置和自定义字体的资料库。它将字体管理简化，并提供了显示功能，比如抗锯齿功能。“**字体设置**”允许那些与字体资料库兼容的应用程序，去使用管理员，或者个人用户安装在系统中的字体。

“**字体设置**”搜索 `/etc/fonts/fonts.conf` 配置文件中默认列出的目录，来整合成一个系统中可用的列表。

您可以使用 `fc-list` 命令，来列出“**字体设置**”中已知安装在系统中的所有字体。

```
$ fc-list : file
```

有关 `fc-list` 的详情，请参见 `fc-list(1)` 手册页。

有关“**字体设置**”及其设置的详情，请参见 `fonts-conf` (5) 手册页。

#### 10.7.1. 为所有用户添加额外字体

您可以在使用“`fontconfig`”处理字体的应用程序中，为用户安装可用的额外字体。

### 过程 10.14. 安装额外的字体

1. 将字体复制到 `/usr/local/share/fonts/` 目录中来安装字体。如果此目录不存在，请重新创建一个。  
为每一个您安装的字体家族创建子目录，因为一些字体有粗体、斜体等等多个文件夹。
2. 运行以下命令，以确保字体缓存已被更新：

```
$ fc-cache /usr/local/share/fonts/
```



“**fontconfig**”将对新字体进行侦测并确保它们是可用的。不像用户会话，一些应用程序可能需要在允许您使用新字体之前进行重启。

或者，您也可以将字体安装到 `/usr/local/share/fonts/` 之外的，其它列在 `/etc/fonts/fonts.conf` 文件夹的系统目录中。如果这个文件夹不在此文件夹中，那么您需在含有您想使用的目录的 `/etc/fonts/local.conf` 文件夹中，以计算机范围设置创建您自己的配置文件。详细信息请见 **fonts-conf(5)** 手册页。

如果您使用的是其它的目录，记得在运行 **fc-cache** 命令更新字体缓存时指定目录名：

```
$ fc-cache directory_name
```

### 10.7.2. 为个人用户安装额外字体

您可以在使用“**fontconfig**”处理文字的系统应用里，为某个特定用户安装额外字体。

#### 过程 10.15. 安装额外字体

1. 将字体复制到 `~/.local/share/fonts/` 目录，来进行安装。
2. 运行以下命令以确保字体缓存已被更新：

```
$ fc-cache ~/.local/share/fonts
```

“**fontconfig**”会对新字体进行侦测并使之可用。您可能需要重启正在运行的应用才能看到改变。用户会话不需要重启。

### 10.7.3. 更换字体

当某个应用要求使用系统中不存在的字体时，“**fontconfig**”会读取 `/etc/fonts/fonts.conf` 配置文件，来判定与所要求的字体最相似的可用字体，并与之进行替换。

您可以使用 **字体调整工具 (Fonts Tweak Tool)**，来为某个特定字体配置替换字体。需要注意的是，这个工具仅可以用作单一用户配置。



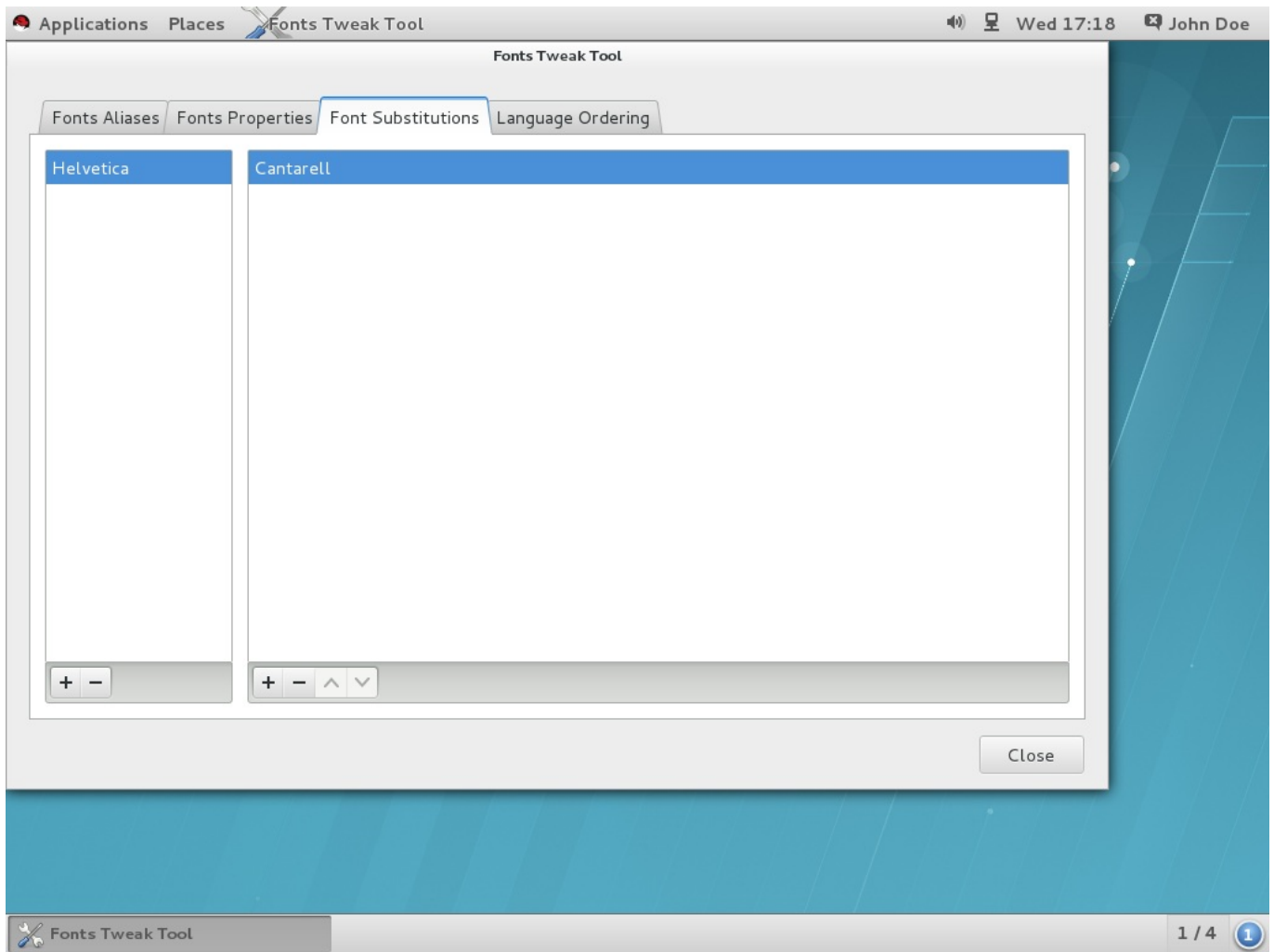


图 10.1. 使用字体调整工具替换字体

字体调整工具 (Fonts Tweak Tool) 可能没有默认安装在系统中，运行以下命令进行安装：

```
# yum install fonts-tweak-tool
```

#### 过程 10.16. 替换字体

1. 按 **Super** 键进入“活动概览”，来启动 **Fonts Tweak Tool**，键入 **Fonts Tweak Tool**，然后按 **Enter** 键。
2. 单击“字体替换”选项卡。
3. 单击左窗格中左下角的“+”按钮，选择或者键入您想要替换的字体名称，然后单击“添加”。
4. 单击右窗格中左下角的“+”按钮，选择您想要用来更换第一个字体的字体名称，然后点击“添加”。
5. 点击“关闭”。

现在，之前的字体已被替换成您想要的字体并可以进行使用。

#### 10.7.4. 配置字体别名

字体调整工具 (Fonts Tweak Tool) 允许个人用户在每个区域配置不同的字体别名：

- » Sans Serif (灯芯体)、

- ✧ Serif（衬线体）、
- ✧ Monospace（等宽字体）、
- ✧ Cursive（草书），以及
- ✧ Fantasy（有衬线的字体）。

这些别名被用来代表一些常见的字体类型，比如衬线体和等宽字体。这样一来，应用程序和用户都可以参考这些别名，而无需指定安装在系统中的某个特定的字体。

用户可以通过为每个字体别名选择一个自定义字体，来覆盖系统默认字体。

**Fonts Tweak Tool** 可能没有默认安装在系统中，运行以下命令以安装此程序：

```
# yum install fonts-tweak-tool
```

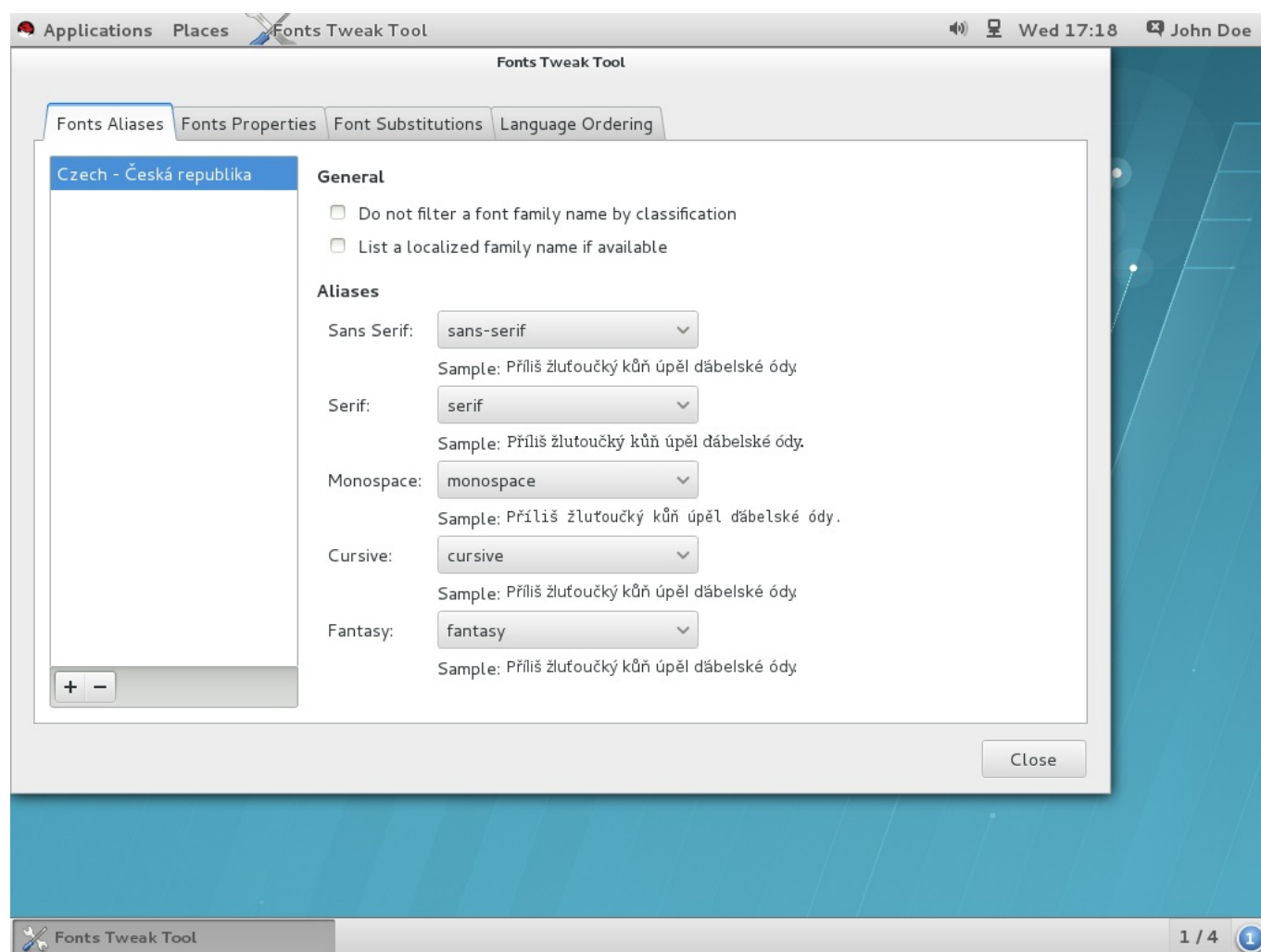


图 10.2. 使用字体调整工具配置字体别名

### 过程 10.17. 配置字体别名

1. 按 **Super** 键进入“活动概览”来启动 **Fonts Tweak Tool**，键入 **Fonts Tweak Tool**，然后按 **Enter** 键。
2. 单击“字体别名”选项卡。
3. 单击左窗格左下角的“+”按钮，选择或者输入您想要配置字体别名的区域名称，然后单击“添加”。

除了使用过的区域，从区域列表中选择默认来配置默认字体别名。

4. 在右侧窗格中选定您想要用来覆盖系统默认设置的字体别名，然后选择下拉列表中的自定义字体。
5. 单击“关闭”。

现在，您已经覆盖了系统默认的别名，并选择了一个自定义字体。

### 10.7.5. 多语言顺序

**字体调整工具 (Fonts Tweak Tool)** 允许在用户界面上配置了多种语言的用户，改变应用程序中显示的语言顺序。这个特征尤其适用于使用以拉丁语字体为基础和以非拉丁语字体为基础的用户，以及那些不想用非拉丁语为基础字体显示拉丁文字的用户。

打个比方，如果您配置的语言是日文和中文，您想避免用非拉丁语为基础的日文字体来显示英语拉丁文字的话，将英文设置成主要语言，日文设置成次要语言。随后，拉丁语为基础的字体用来显示英文字符，而以非拉丁语为基础的字体则会用来显示日文字符。

**Fonts Tweak Tool** 可能没有默认安装到系统中，运行以下命令进行安装：

```
# yum install fonts-tweak-tool
```

#### 过程 10.18. 配置多种语言

1. 按 **Super** 键进入“活动概览”来启动 **Fonts Tweak Tool**，键入 **Fonts Tweak Tool**，然后按 **Enter** 键。
2. 单击“**Language Ordering**”选项卡。
3. 单击窗口左下方的“+”按钮，选择或者键入您想要配置成主要语言的语言名称，然后单击“**添加**”。
4. 单击窗口左下方的“+”按钮，选择您想要配置成次要语言的语言名称，然后单击“**添加**”来添加其他语言。

重复此项操作以增加更多语言。

5. 单击“关闭”。



#### 重要

您已在用户界面上将语言偏好顺序设置成应用程序中语言显示的顺序。

当配置了多个语言时，一些应用程序（比如 **xterm** 和其它“**Xft**”应用程序）可能不会准确显示用户语言的所有字符。这是因为这些应用程序，或者应用程序所使用的渲染库无法支持备用字体。

### 10.7.6. 配置字体属性

**Fonts Tweak Tool** 允许用户更改多个字体属性，并为每个用户配置精细的字体。

**Fonts Tweak Tool** 可能没有默认安装在系统中，运行以下命令进行安装：

```
# yum install fonts-tweak-tool
```

## 过程 10.19. 更改字体属性

1. 按 **Super** 键进入“活动概览”来启动 **Fonts Tweak Tool**，键入 **Fonts Tweak Tool**，然后按 **Enter** 键。
2. 点击“**字体属性**”选项卡。
3. 单击窗口左下方的“+”按钮，选择或者键入您想要更改属性的字体名称，然后单击“**添加**”。  
  
重复此项操作来增加更多字体。
4. 根据需要更改字体属性。
5. 单击“**关闭**”。

根据增加的字体，用户可以在“**字体属性**”选项卡中配置的一些字体属性包括：

**如果可以的话，使用植入式位图。**

这对喜欢点阵字体多于外框字体的用户来说很适用。添加一个合适的字体并单击“**Use embedded bitmap font if any**”来使用植入式点阵字体。

**使用 JIS X 2013:2004 字形。**

使用 JIS X 2013:2004 标准的日文字形，而不是 JIS X 2013:2000 或者更老的版本，添加支持 JIS X 2013:2004 的字体，然后单击“**特性**”列表中的“**jp04**”。

## 第 11 章 GNOME Shell 扩展

本章将介绍 GNOME Shell 扩展的系统范围配置。您将了解到如何查看这些扩展、如何激活这些扩展、如何锁定启用扩展列表，或者如何给系统用户设置一些强制的扩展。

在配置 GNOME Shell 扩展时，您将使用“**dconf**”设置以下两个 GSetting 键值：

- ✦ `org.gnome.shell.enabled-extensions`
- ✦ `org.gnome.shell.development-tools`

更多有关“**dconf**”和 **GSettings** 的信息，请参见〈[第 9 章 用 GSettings 和 dconf 来配置桌面](#)〉。

### 11.1. GNOME Shell 扩展是什么？

GNOME Shell 扩展允许自定义 GNOME Shell 默认用户界面及其某些部分，比如窗口管理和应用程序启动。

每一个 GNOME Shell 扩展都是由唯一标识符 `uuid` 来进行识别的。`uuid` 同时也用来命名扩展安装的目录名称。您可以在 `~/.local/share/gnome-shell/extensions/uuid` 中为每一个用户安装扩展，或者在 `/usr/share/gnome-shell/extensions/uuid` 中进行计算机范围的安装。

`uuid` 标示符（通用唯一标识符）是全局唯一标识符。请记住，在选定 `uuid` 标示符时，它必须包含以下属性从而预防某些攻击：

- ✦ 您的通用唯一标识符必须不包含 Unicode 字符。
- ✦ 您的通用唯一标识符必须不包含以 `gnome.org` 结尾的字串，因为它不能附属于 GNOME 项目。
- ✦ 您的通用唯一标识符必须只包含字母数字字符与下列符号：句号（.）、at 符号（@），和下划线（\_）。



#### 重要

在红帽企业版 Linux 中配置第三方 GNOME Shell 扩展之前，确保您已阅读过以下文件，以了解红帽对第三方软件的支持策略：

- ✦ [How does Red Hat Global Support Services handle third-party software, drivers, and/or uncertified hardware/hypervisors?](#)

您可以使用 GNOME Shell 的集成调试器和检查器工具 **Looking Glass** 查看已安装的扩展。

#### 过程 11.1. 查看已安装的扩展

1. 按 **Alt+F2** 键。
2. 键入 **lg**，然后按 **Enter** 键，来打开 **Looking Glass**。
3. 在 **Looking Glass** 的顶部菜单栏中，点击“**扩展**”，来打开已安装的扩展列表。

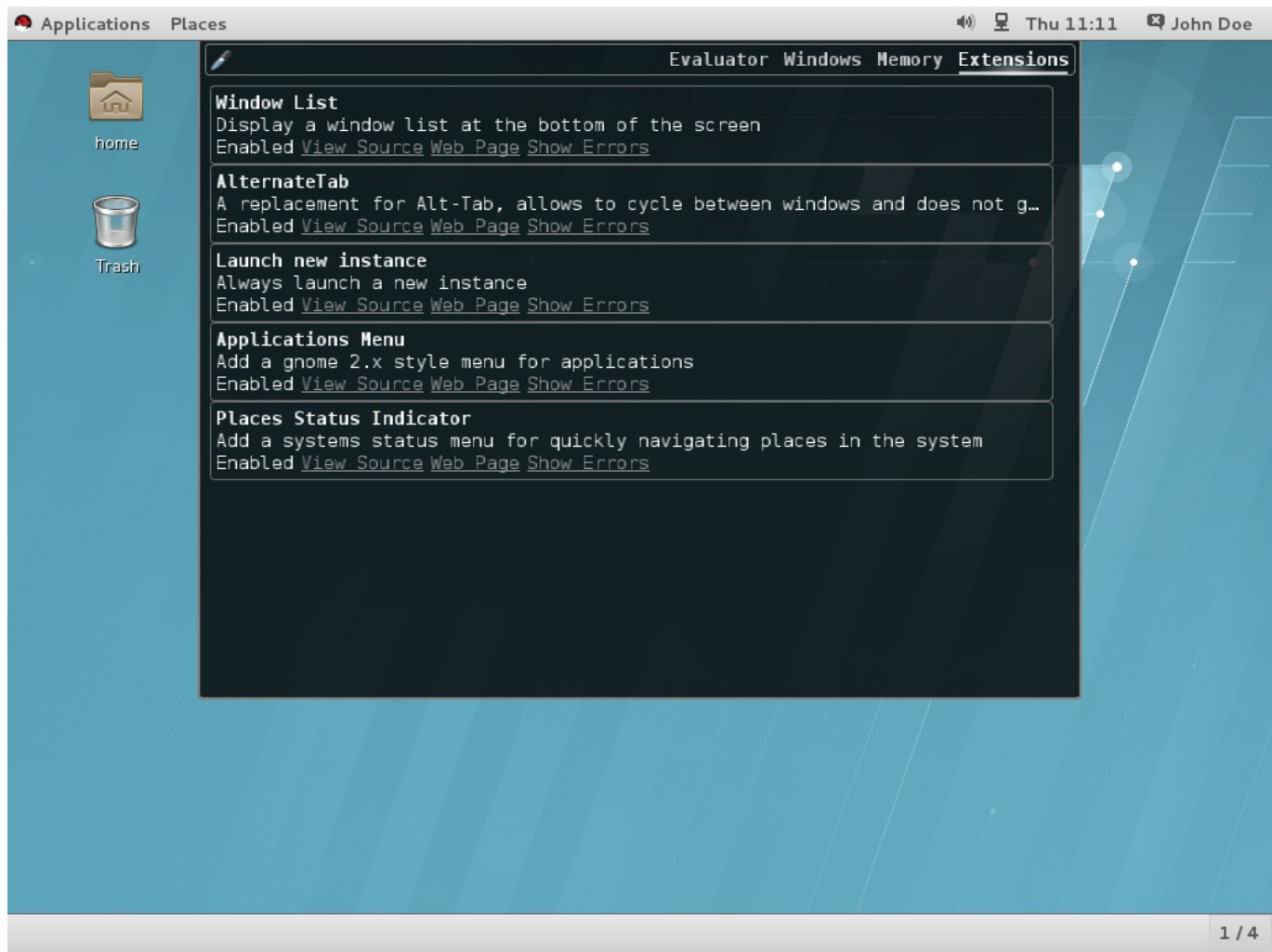


图 11.1. 通过 Looking Glass 来查看已安装的扩展

## 11.2. 启用计算机范围的扩展程序

如需使所有用户能够使用该系统上的扩展程序，请将扩展程序安装在 `/usr/share/gnome-shell/extensions` 目录下。注意新安装的计算机范围的扩展程序是默认不启用的。为了使所有用户可以启用扩展程序，您需要设置 `org.gnome.shell.enabled-extensions` 的键值。

### 过程 11.2. 启用计算机范围的扩展程序

1. 在 `/etc/dconf/db/local.d/00-extensions` 中创建一个用于计算机范围设置的 `local` 数据库文件。

```
[org/gnome/shell]
# List all extensions that you want to have enabled for all users
enabled-extensions=['myextension1@myname.example.com',
'myextension2@myname.example.com']
```

`enabled-extensions` 的键值通过扩展程序的 uuid 规定了被启用的扩展程序。  
(`myextension1@myname.example.com` 和 `myextension2@myname.example.com`)

2. 更新系统数据库：

```
# dconf update
```

### 11.3. 锁定被启用的应用程序

在 GNOME Shell 系统中，您可以通过锁定 `org.gnome.shell.enabled-extensions` 和 `org.gnome.shell.development-tools` 的键值来阻止用户启用或停用扩展程序。

锁定 `org.gnome.shell.development-tools` 的键值可以确保用户无法用 GNOME Shell 的内置调试器和检查工具（Looking Glass）停用任何强制性的扩展程序。

#### 过程 11.3. 锁定被启用的应用程序

1. 在 `/etc/dconf/db/local.d/00-extensions` 中创建一个用于计算机范围设置的 `local` 数据库文件。

```
[org/gnome/shell]
# List all extensions that you want to have enabled for all users
enabled-extensions=['myextension1@myname.example.com',
'myextension2@myname.example.com']
# Disable access to Looking Glass
development-tools=false
```

`enabled-extensions` 的键值通过扩展程序的 uuid 规定了被启用的扩展程序（`myextension1@myname.example.com` 和 `myextension2@myname.example.com`）。

`development-tools` 的键值被设置为假（false），以阻止用户使用应用程序 Looking Glass。

2. 覆盖用户设置，并阻止用户在文件 `/etc/dconf/db/local.d/locks/extensions` 中对此修改：

```
# Lock the list of mandatory extensions and access to Looking Glass
/org/gnome/shell/enabled-extensions
/org/gnome/shell/development-tools
```

3. 更新系统数据库：

```
# dconf update
```

锁定 `org.gnome.shell.enabled-extensions` 和 `org.gnome.shell.development-tools` 的键值之后，GNOME Shell 将不会加载任何安装在 `~/.local/share/gnome-shell/extensions` 或 `/usr/share/gnome-shell/extensions` 中的没有被列在 `org.gnome.shell.enabled-extensions` 键值中的扩展程序，从而阻止用户使用它们。

### 11.4. 设置强制性扩展程序

在 GNOME Shell 系统中，您可以提供一组用户必须使用的扩展程序。为此，在 `/usr/share/gnome-shell/extensions` 目录中安装这些扩展程序，然后锁定 `org.gnome.shell.enabled-extensions` 和 `org.gnome.shell.development-tools` 的键值。

锁定 `org.gnome.shell.development-tools` 的键值可以确保用户无法用 GNOME Shell 的内置调试器和检查工具（Looking Glass）停用任何强制性的扩展程序。

#### 过程 11.4. 设置强制性扩展程序

1. 在 `/etc/dconf/db/local.d/00-extensions-mandatory` 中创建一个用于计算机范围设置的 `local` 数据库文件。

```
[org/gnome/shell]
# List all mandatory extensions
enabled-extensions=['myextension1@myname.example.com',
'myextension2@myname.example.com']
# Disable access to Looking Glass
development-tools=false
```

**enabled-extensions** 的键值通过扩展程序的 uuid 规定了被启用的扩展程序 (*myextension1@myname.example.com* 和 *myextension2@myname.example.com*) 。

**development-tools** 的键值被设置为假 (false) ，以阻止用户使用应用程序 **Looking Glass**。

2. 覆盖用户设置，并阻止用户在 **/etc/dconf/db/local.d/locks/extensions-mandatory** 中对此修改：

```
# Lock the list of mandatory extensions and access to Looking Glass
/org/gnome/shell/enabled-extensions
/org/gnome/shell/development-tools
```

3. 更新系统数据库：

```
# dconf update
```



## 第 12 章 整合应用程序

在 GNOME 桌面上整合应用程序时，系统管理员通常会执行有关定制“应用程序”菜单结构和 MIME 类型的相关任务，例如：

- ✧ 通过创建或修改子菜单为应用程序添加或修改一个菜单项，或自定义“应用程序”的菜单结构。关于菜单定制的更多信息，请参见〈[第 12.1 节“自定义菜单”](#)〉。
- ✧ 在“活动概览”中自定义 GNOME Shell *dash* 上可见的默认最热门程序。关于如何操作的更多信息，请参见〈[第 12.2 节“自定义默认的最热门程序”](#)〉。
- ✧ 添加或修改应用程序的 MIME 类型，并赋予应用程序某一特定的 MIME 类型。关于配置 MIME 类型的更多信息，请参见〈[第 12.3 节“配置文件关联”](#)〉。

### 12.1. 自定义菜单

GNOME 菜单系统基于 *freedesktop.org Desktop Menu Specification*，由三组主要的配置文件和数据文件组成。

#### 桌面项文件（.desktop）

后缀 **.desktop** 的文件提供每个菜单项的名称、运行命令、图标等数据。后缀 **.desktop** 的条目文件也规定了菜单项在菜单结构中的位置，以及在“活动概览”中搜寻应用程序所使用的键值。

**.desktop** 系统文件位于 `/usr/share/applications/` 目录下。针对用户特定的 **.desktop** 文件位于 `~/.local/share/applications/` 目录下。

以下是一个名为 `~/.local/share/applications/myapplication1.desktop` 的 **.desktop** 样本文件：

```
[Desktop Entry]
Type=Application
Name=My Application 1
Icon=myapplication1
Exec=myapplication1
Categories=Network;WebBrowser;
MimeType=application/x-newtype
```

上述文件规定了应用程序的名称（**My Application 1**），应用程序的图标（**myapplication1**），以及运行该应用程序的命令（**myapplication1**）。它还将这个应用程序放置在一个特定的类别目录下（**“Network;WebBrowser;”**），并且将该应用程序与“**application/x-newtype**”MIME 类型关联。

#### 菜单定义文件（.menu）

**.menu** 文件是 XML 配置文件，它们规定了菜单和菜单项的顺序、级别和合并。

计算机范围的 **.menu** 文件位于 `/etc/xdg/menus/` 目录下。针对用户特定的 **.menu** 文件位于 `~/.config/menus/` 目录下，它们可以用来覆盖计算机范围的 **.menu** 文件中规定的值。

特别的一点是，`/etc/xdg/menus/applications.menu` 文件中包括对“应用程序”菜单布局的定义。

#### 目录项文件（.directory）

**.directory** 文件提供每个菜单的名称等数据，它们位于 `/usr/share/desktop-`

**directories/** 中。

## 获得更多信息

关于描述桌面项文件的更多信息，请参见位于 freedesktop.org 网站的《桌面项说明》：

» <http://freedesktop.org/wiki/Specifications/desktop-entry-spec>

关于描述 GNOME 菜单系统执行情况的详细信息，请参见位于 freedesktop.org 网站的《桌面菜单说明》。

» <http://standards.freedesktop.org/menu-spec/latest>

### 12.1.1. 为个别用户移除某个菜单项

针对某个指定用户的“应用程序”菜单定制默认存储于 `~/.config/menus/gnome-applications.menu` 定义文件中。该文件的位置可以通过设置 `$XDG_DATA_HOME` 环境变量被覆盖。

如需覆盖“应用程序”菜单的默认属性，首先需要创建一个 `gnome-applications.menu` 文件。要注意，从“应用程序”菜单移除一个目录项及其子菜单时，也会将它从“活动概览”中的应用程序视图里移除，从而防止用户在“概览”中搜索该项。

#### 过程 12.1. 示例：从附件子菜单中移除计算器菜单项

1. 查阅 `/usr/share/applications/` 目录的内容，并确定您想要移除的菜单项所对应的 `.desktop` 文件：

```
$ grep -r "Name=Calculator" /usr/share/applications/
/usr/share/applications/gcalctool.desktop:Name=Calculator
```

如上所示，“计算器”菜单项对应 `/usr/share/applications/gcalctool.desktop` 文件。

2. 创建一个 `~/.config/menus/gnome-applications.menu` 文件：

```
<!DOCTYPE Menu PUBLIC "-//freedesktop//DTD Menu 1.0//EN"
"http://www.freedesktop.org/standards/menu-spec/1.0/menu.dtd">

<Menu>
  <Name>Applications</Name>
  <MergeFile type="parent">/etc/xdg/menus/gnome-
applications.menu</MergeFile>

  <!-- Removes the Calculator from the Accessories submenu -->
  <Menu>
    <Name>Accessories</Name>
    <Exclude>
      <Filename>gcalctool.desktop</Filename>
    </Exclude>
  </Menu>
  <!-- END of Calculator removal content -->
</Menu>
```

如上所示，文件中包括一个规定子菜单名字（Accessories）的 `<Menu>` 部分、`.desktop` 文件的名字（`gcalctool.desktop`）、以及 `<Exclude>` 元素。

### 12.1.2. 为所有用户移除某个菜单项

面向所有用户的“应用程序”菜单定制默认存储于 `/etc/xdg/menus/applications.menu` 定义文件中。该文件的位置可以通过设置 `$XDG_CONFIG_DIRS` 环境变量被覆盖。

如需覆盖“应用程序”菜单的默认属性，您需要编辑那个 `.menu` 文件。要注意，从“应用程序”菜单移除一个目录项及其子菜单时，也会将它从“活动概览”中的应用程序视图里移除，从而防止用户在“概览”中搜索该项。

#### 过程 12.2. 示例：从附件子菜单中移除计算器菜单项

1. 查阅 `/usr/share/applications/` 目录的内容，并确定您想要移除的菜单项所对应的 `.desktop` 文件：

```
$ grep -r "Name=Calculator" /usr/share/applications/
/usr/share/applications/gcalctool.desktop:Name=Calculator
```

如上所示，“计算器 (Calculator)”菜单项对应 `/usr/share/applications/gcalctool.desktop` 文件。

2. 编辑 `/etc/xdg/menus/applications.menu` 文件，并通过如下所示的 `<Exclude>` 元素在那个 `.menu` 文件末尾处最后的 `</Menu>` 标签前添加一个新的 `<Menu>` 部分。

```
<!-- Removes the Calculator from the Accessories submenu -->

<Menu>
  <Name>Accessories</Name>
  <Exclude>
    <Filename>gcalctool.desktop</Filename>
  </Exclude>
</Menu>

<!-- END of Calculator removal content -->

</Menu> <!-- End Applications -->
```

### 12.1.3. 为个别用户移除某个子菜单

针对某个指定用户的“应用程序”菜单定制默认存储于 `~/.config/menus/gnome-applications.menu` 定义文件中。该文件的位置可以通过设置 `$XDG_DATA_HOME` 环境变量被覆盖。

如需覆盖“应用程序”菜单的默认属性，首先需要创建一个 `gnome-applications.menu` 文件。要注意，从“应用程序”菜单中移除一个子菜单时，也会将它所包括的所有菜单项从“活动概览”中的应用程序视图里移除，从而防止用户在“概览”中搜索这些项目。

#### 例 12.1. 从应用程序菜单中移除系统工具子菜单

创建一个 `~/.config/menus/gnome-applications.menu` 文件：

```
<!DOCTYPE Menu PUBLIC "-//freedesktop//DTD Menu 1.0//EN"
"http://www.freedesktop.org/standards/menu-spec/1.0/menu.dtd">

<Menu>
  <Name>Applications</Name>
  <MergeFile type="parent">/etc/xdg/menus/gnome-
```

```
applications.menu</MergeFile>

<!-- Removes the System Tools submenu from the Applications menu-->

<Menu>
  <Name>System Tools</Name>
  <Deleted/>
</Menu>

<!-- END of System Tools removal content -->

</Menu>
```

如上所示，文件中包括一个规定子菜单名字（System Tools）的 **<Menu>** 部分，还包括 **<Deleted/>** 标签。

#### 12.1.4. 为所有用户移除某个子菜单

面向所有用户的“应用程序”菜单定制默认存储于 `/etc/xdg/menus/applications.menu` 定义文件中。该文件的位置可以通过设置 `$XDG_CONFIG_DIRS` 环境变量被覆盖。

如需覆盖“应用程序”菜单的默认属性，您需要修改那个 `.menu` 文件。要注意，从“应用程序”菜单中移除一个子菜单时，也会将它所包括的所有菜单项从“活动概览”中的应用程序视图里移除，从而防止用户在“概览”中搜索这些项目。

##### 例 12.2. 从应用程序菜单中移除系统工具子菜单

编辑 `/etc/xdg/menus/applications.menu` 文件，并通过如下所示的 **<Deleted/>** 元素在那个 `.menu` 文件末尾处最后的 **</Menu>** 标签前添加一个新的 **<Menu>** 部分。

```
<!-- Removes the System Tools submenu from the Applications menu-->

<Menu>
  <Name>System Tools</Name>
  <Deleted/>
</Menu>

<!-- END of System Tools removal content -->

</Menu>
```

## 12.2. 自定义默认的最热门程序

最热门程序是那些在“活动概览”中的 GNOME Shell *dash* 上可见的程序。可以使用“**dconf**”来为个别用户设置最热门程序，或为所有用户设置相同的最热门程序。

### 12.2.1. 为个别用户设置不同的最热门程序

可以通过修改 `~/.config/dconf/user` 中找到的用户数据库文件来为个别用户设置默认最热门程序。以下样本文件中通过“**dconf**”将 **gedit**，**Terminal** 和 **Nautilus** 设置为某个用户的默认最热门程序。如果用户希望的话，该例子代码可以允许他们以后修改清单。

例 12.3. `/etc/dconf/profile` 文件的内容：

```
# This line allows the user to change the default favorites later
user-db:user
```

例 12.4. `~/.config/dconf/user` 文件的内容：

```
# Set gedit, terminal and nautilus as default favorites
[org/gnome/shell]
favorite-apps = ['gedit.desktop', 'gnome-terminal.desktop',
'nautilus.desktop']
```



### 注意

您也可以锁定上述设置来阻止用户改变它们。更多信息请参见 [〈第 9.5.1 节 “锁定特定设置”〉](#)。

## 12.2.2. 为所有用户设置相同的最热门程序

为了使所有用户拥有相同的最热门程序，必须通过“**dconf**”密钥文件修改系统的数据库文件。以下样本文件修改了“**dconf**”配置文件，然后创建了一个键值文件来设置某个机构内一楼所有员工的默认最热门程序。

例 12.5. `/etc/dconf/profile` 文件的内容：

```
user-db:user

# This line defines a system database called first_floor
system-db:first_floor
```



### 注意

“**用户**”数据库文件中的设置优先于“**first\_floor**”数据库文件中的设置，但“**first\_floor**”数据库文件中引入的锁定优先于出现在“**用户**”中的锁定。关于锁定的更多信息，请参见 [〈第 9.5.1 节 “锁定特定设置”〉](#)。

例 12.6. `/etc/dconf/db/first_floor.d/00_floor1_settings` 文件的内容：

```
# This sample sets gedit, terminal and nautilus as default favorites
# for all users in the first floor
[org/gnome/shell]
favorite-apps = ['gedit.desktop', 'gnome-terminal.desktop',
'nautilus.desktop']
```

通过运行 **dconf update** 命令将您的更改合并到系统数据库中。

## 12.3. 配置文件关联

### 12.3.1. 什么是 MIME 类型？

在 GNOME 系统中，通过 MIME（多用途 Internet 邮件扩展）的类型来识别文件格式。GNOME 桌面使用 MIME 类型来：

- ✧ 确定默认用哪个应用程序打开某种特定的文件格式。
- ✧ 注册同样能打开某种特定文件格式的其它应用程序。
- ✧ 在例如 **文件** 应用程序的文件属性对话框中，提供了描述文件类型的字符串。
- ✧ 在例如 **文件** 应用程序的文件属性对话框中，提供了代表某种特定文件格式的图标。

MIME 类型名字遵循指定的格式：

```
media-type/subtype-identifier
```

#### 例 12.7. MIME 类型格式

“**image/jpeg**”是 MIME 类型的一个例子，其中“**image**”是媒体类型，“**jpeg**”是子类型识别符。

GNOME 系统根据 *freedesktop.org Shared MIME Info* (*freedesktop.org 共享 MIME 信息*) 规范来确定：

- ✧ 存储所有 MIME 类型说明文件的位置，包括计算机范围的位置和用户特定的位置。
- ✧ 如何注册一种 MIME 类型，使得桌面环境知道哪些应用程序可以用来打开某种特定的文件格式。
- ✧ 用户如何更改哪种应用程序打开哪种文件格式。

#### 12.3.1.1. 什么是 MIME 数据库？

MIME 数据库是 GNOME 系统用来存储关于已知 MIME 类型信息的所有 MIME 类型说明文件的集合。

从系统管理员的角度来看，MIME 数据库最重要的部分是 `/usr/share/mime/packages/` 目录，其中存储着对已知 MIME 类型信息进行说明的 MIME 类型相关文件。这种文件的一个例子是 `/usr/share/mime/packages/freedesktop.org.xml`，它对默认设置下系统中可用的标准 MIME 类型的信息作了说明。该文件由 *shared-mime-info* 软件包提供。

#### 12.3.1.2. 获得更多信息

描述 MIME 类型系统的详细信息，请参见位于网站 *freedesktop.org* 上的 *freedesktop.org Shared MIME Info specification* (*freedesktop.org 共享 MIME 信息说明*)。

- ✧ <http://www.freedesktop.org/wiki/Specifications/shared-mime-info-spec/>

### 12.3.2. 为所有用户添加某种自定义的 MIME 类型

如需为系统上的所有用户添加一个自定义的 MIME 类型，并为该 MIME 类型注册一个默认的应用程序，您需要在 `/usr/share/mime/packages/` 目录下创建一个新的 MIME 类型说明文件，在 `/usr/share/applications/` 目录下创建一个 `.desktop` 文件。

过程 12.3. 为所有用户添加自定义的“`application/x-newtype`”MIME 类型。



1. 创建 `/usr/share/mime/packages/application-x-newtype.xml` 文件：

```
<?xml version="1.0" encoding="UTF-8"?>
<mime-info xmlns="http://www.freedesktop.org/standards/shared-mime-info">
  <mime-type type="application/x-newtype">
    <comment>new mime type</comment>
    <glob pattern="*.xyz"/>
  </mime-type>
</mime-info>
```

上述 `application-x-newtype.xml` 样本文件定义了一种新的 MIME 类型“`application/x-newtype`”，并指定扩展名是 `.xyz` 的文件为该 MIME 类型。

2. 创建一个名为例如 `myapplication1.desktop` 的新的 `.desktop` 文件，并将它放置在 `/usr/share/applications/` 目录下：

```
[Desktop Entry]
Type=Application
MimeType=application/x-newtype
Name=My Application 1
Exec=myapplication1
```

上述 `myapplication1.desktop` 样本文件将“`application/x-newtype`”MIME 类型与一个名为 `My Application 1` 的应用程序相关联，该应用程序由 `myapplication1` 命令运行。

3. 请以 root 身份更新 MIME 数据库以使您的更改生效：

```
# update-mime-database /usr/share/mime
```

4. 请以 root 身份更新应用程序数据库：

```
# update-desktop-database /usr/share/applications
```

5. 如需确认您已成功将 `*.xyz` 文件与“`application/x-newtype`”MIME 类型相关联，首先需要创建一个空白文档，例如文件 `test.xyz`：

```
$ touch test.xyz
```

然后运行 `gvfs-info` 命令：

```
$ gvf-info test.xyz | grep "standard::content-type"
standard::content-type: application/x-newtype
```

6. 如需确认 `myapplication1.desktop` 文件已被正确设置为“`application/x-newtype`”MIME 类型的默认注册应用程序，请运行 `gvfs-mime --query` 命令：

```
$ gvf-mime --query application/x-newtype
Default application for 'application/x-newtype':
myapplication1.desktop
Registered applications:
```

```
myapplication1.desktop
Recommended applications:
myapplication1.desktop
```

### 12.3.3. 为个别用户添加自定义的 MIME 类型

如需为个别用户添加自定义的 MIME 类型，并为该 MIME 类型注册一个默认的应用程序，您需要在 `~/.local/share/mime/packages/` 目录下创建一个 新的 MIME 类型说明文件，并在 `~/.local/share/applications/` 目录下创建一个 `.desktop` 文件。

#### 过程 12.4. 为个别用户添加自定义的 `application/x-newtype` MIME 类型

1. 创建 `~/.local/share/mime/packages/application-x-newtype.xml` 文件：

```
<?xml version="1.0" encoding="UTF-8"?>
<mime-info xmlns="http://www.freedesktop.org/standards/shared-mime-
info">
  <mime-type type="application/x-newtype">
    <comment>new mime type</comment>
    <glob pattern="*.xyz"/>
  </mime-type>
</mime-info>
```

上述 `application-x-newtype.xml` 样本文件定义了一种新的 MIME 类型“`application/x-newtype`”，并指定拓展名是 `.xyz` 的文件为该 MIME 类型。

2. 创建一个名为例如 `myapplication1.desktop` 的新的 `.desktop` 文件，并将它放置在 `~/.local/share/applications/` 目录下：

```
[Desktop Entry]
Type=Application
MimeType=application/x-newtype
Name=My Application 1
Exec=myapplication1
```

上述 `myapplication1.desktop` 样本文件将 `application/x-newtype` MIME 类型与一个名为 `My Application 1` 的应用程序相关联，该应用程序由 `myapplication1` 命令运行。

3. 更新 MIME 数据库以使您的更改生效：

```
$ update-mime-database ~/.local/share/mime
```

4. 更新应用程序数据库：

```
$ update-desktop-database ~/.local/share/applications
```

5. 如需确认您已成功将 `*.xyz` 文件与“`application/x-newtype`”MIME 类型相关联，首先需要创建一个空白文档，例如文件 `test.xyz`：

```
$ touch test.xyz
```

然后运行 `gvfs-info` 命令：



```
$ gvfs-info test.xyz | grep "standard::content-type"
standard::content-type: application/x-newtype
```

- 如需确认 **myapplication1.desktop** 文件已被正确设置为“**application/x-newtype**”MIME 类型的注册默认应用程序，请运行 **gvfs-mime --query** 命令：

```
$ gvfs-mime --query application/x-newtype
Default application for 'application/x-newtype':
myapplication1.desktop
Registered applications:
  myapplication1.desktop
Recommended applications:
  myapplication1.desktop
```

### 12.3.4. 为所有用户覆盖默认注册的应用程序

**/usr/share/applications/defaults.list** 文件说明了哪个应用程序被注册默认打开特定的 MIME 类型。如需为系统所有用户覆盖系统默认设置，您需要创建一个 **/usr/share/applications/mimeapps.list** 文件，其中包括了您想要覆盖的默认注册应用程序所对应的 MIME 类型的清单。

#### 过程 12.5. 为所有用户覆盖默认注册的应用程序

- 查阅 **/usr/share/applications/defaults.list** 文件来确定您想要更改的默认注册应用程序所对应的 MIME 类型。例如，如下 **defaults.list** 样本文件中说明了 **text/html** 和 **application/xhtml+xml** MIME 类型所对应的默认注册应用程序：

```
[Default Applications]
text/html=firefox.desktop
application/xhtml+xml=firefox.desktop
```

通过规定对应的 **.desktop** 文件 (**firefox.desktop**) 来定义默认应用程序 (**Firefox**)。其它应用程序的 **.desktop** 文件的默认位置是 **/usr/share/applications/**。

- 创建 **/usr/share/applications/mimeapps.list** 文件。在文件中说明 MIME 类型和它们对应的默认注册应用程序：

```
[Default Applications]
text/html=myapplication1.desktop
application/xhtml+xml=myapplication2.desktop

[Added Associations]
text/html=myapplication1.desktop;
application/xhtml+xml=myapplication2.desktop;
```

这将 **text/html** MIME 类型的默认注册应用程序设置为 **myapplication1.desktop**，将 **application/xhtml+xml** MIME 类型的默认注册应用程序设置为 **myapplication2.desktop**。

为了使这些设置正确起作用，请确保 **myapplication1.desktop** 文件和 **myapplication2.desktop** 文件都放置于 **/usr/share/applications/** 目录下。

- 您可以使用 **gvfs-mime --query** 命令来确认默认注册应用程序已正确设置：

```
$ gvfs-mime --query text/html
Default application for 'text/html': myapplication1.desktop
Registered applications:
  myapplication1.desktop
  firefox.desktop
Recommended applications:
  myapplication1.desktop
  firefox.desktop
```

### 12.3.5. 为个别用户覆盖默认注册的应用程序

**/usr/share/applications/defaults.list** 文件说明了哪个应用程序被注册默认打开特定的 MIME 类型。如需为个别用户覆盖系统默认设置，您需要创建一个 **~/.local/share/applications/mimeapps.list** 文件，其中包括了您想要覆盖的默认注册应用程序所对应的 MIME 类型的清单。

#### 过程 12.6. 为个别用户覆盖默认注册的应用程序

1. 查阅 **/usr/share/applications/defaults.list** 文件来确定您想要更改的默认注册应用程序所对应的 MIME 类型。例如，如下 **defaults.list** 样本文件中说明了 **text/html** 和 **application/xhtml+xml** MIME 类型所对应的默认注册应用程序：

```
[Default Applications]
text/html=firefox.desktop
application/xhtml+xml=firefox.desktop
```

通过规定对应的 **.desktop** 文件 (**firefox.desktop**) 来定义默认应用程序 (**Firefox**)。其它应用程序的 **.desktop** 文件的系统默认位置是 **/usr/share/applications/**。个别用户的 **.desktop** 文件可以储存在 **~/.local/share/applications/** 中。

2. 创建 **~/.local/share/applications/mimeapps.list** 文件。在文件中说明 MIME 类型和它们对应的默认注册应用程序：

```
[Default Applications]
text/html=myapplication1.desktop
application/xhtml+xml=myapplication2.desktop

[Added Associations]
text/html=myapplication1.desktop;
application/xhtml+xml=myapplication2.desktop;
```

这将 **text/html** MIME 类型的默认注册应用程序设置为 **myapplication1.desktop**，将 **application/xhtml+xml** MIME 类型的默认注册应用程序设置为 **myapplication2.desktop**。

为了使这些设置正确起作用，请确保 **myapplication1.desktop** 文件和 **myapplication2.desktop** 文件都放置于 **/usr/share/applications/** 目录下。

3. 您可以使用 **gvfs-mime --query** 命令来确认默认注册应用程序已正确设置：

```
$ gvfs-mime --query text/html
Default application for 'text/html': myapplication1.desktop
Registered applications:
  myapplication1.desktop
```

```
firefox.desktop  
Recommended applications:  
myapplication1.desktop  
firefox.desktop
```

## 第 13 章 自定义 GNOME 系统的桌面功能

这一章提到了三个重要的桌面功能。阅读本章后您将知道如何为所有用户迅速终止默认的 X 服务器，如何启用**构成键**，及如何禁止用户使用命令行。

为确保您所做的更改生效，需要更新“**dconf**”实用工具。用户在退出和再次登录之后将体会到不同之处。

### 13.1. 启用 **Ctrl+Alt+Backspace** 快捷方式

**Ctrl+Alt+Backspace** 快捷方式键的组合用来终止 X 服务器。尤其在以下时候您也许想要终止 X 服务器：

- » 某个程序使 X 服务器停止工作。
- » 您需要从已登录会话中快速切换出来。
- » 您启动的一个程序运行失败。
- » 由于各种原因您无法在当前会话中执行操作。
- » 屏幕冻结

如需启用 **Ctrl+Alt+Backspace** 快捷方式来为所有用户强制终止默认的 X 服务器，您需要设置 **org.gnome.desktop.input-sources.xkb-options** GSettings 键值（关于 GSettings 键值的更多信息，请参见〈[第 9.6 节 “GSettings 键值属性”](#)〉）。

#### 过程 13.1. 启用 **Ctrl-Alt-Backspace** 快捷方式

1. 在 **/etc/dconf/db/local.d/00-input-sources** 中创建一个用于计算机范围设置的 **local** 数据库：

```
[org/gnome/desktop/input-sources]
# Enable Ctrl-Alt-Backspace for all users
xkb-options=['terminate:ctrl_alt_bksp']
```

2. 覆盖用户设置，并阻止用户在 **/etc/dconf/db/local.d/locks/input-sources** 中对此修改：

```
# Lock the list of enabled XKB options
/org/gnome/desktop/input-sources/xkb-options
```

3. 更新系统数据库以使更改生效：

```
# dconf update
```

**Ctrl+Alt+Backspace** 按键组合现已启用。所有用户都可以快速简单地终止 X 服务器，且该操作将把用户带回登录提示符处。

### 13.2. 启用**构成键**

“**构成键**”是一种能使您输入键盘上无法找到的特殊符号或字符的功能。在 GNOME 桌面系统中，您可以将键盘上现有的其中一个键定义为“**构成键**”。“**构成键**”和其它被称为构成键序列的键一起使用，用来输入您经常打的特殊字符。

### 例 13.1. 使用构成键

如果您按住再释放“**构成键**”，然后输入 AE（大写），您将得到“Æ”。如果输入小写字母，您将得到“æ”。

为了启用“**构成键**”，并将键盘上的某特定键设为“**构成键**”，需要设置 `org.gnome.desktop.input-sources.xkb-options` GSettings 键值。这样一来，系统所有用户都将默认启用该设置（关于 GSettings 键值的更多信息，请参见〈[第 9.6 节 “GSettings 键值属性”](#)〉）。

### 过程 13.2. 将右侧的 Alt 键设为构成键

1. 在 `/etc/dconf/db/local.d/00-input-sources` 中创建一个用于计算机范围设置的 `local` 数据库：

```
[org/gnome/desktop/input-sources]
# Set the Right Alt key as the Compose key and enable it
xkb-options=['compose:ralt']
```

如果您想要设置右侧 **Alt** 键以外的其它键，将 `ralt` 替换成那个键的名称，该名称在〈[构成键位置](#)〉一节中的 `xkeyboard-config` (7) 手册页上有说明。

2. 覆盖用户设置，并阻止用户在 `/etc/dconf/db/local.d/locks/input-sources` 中对此修改：

```
# Lock the list of enabled XKB options
/org/gnome/desktop/input-sources/xkb-options
```

3. 更新系统数据库，以使更改生效：

```
# dconf update
```

现在，您可以开始将右侧的 **Alt** 键用作“**构成键**”。只要按住再释放“**构成键**”，然后输入按键的组合，就可以得到特殊符号。



### 注意

作为系统管理员，如果您在编辑本地数据库之后进行锁定，用户将无法添加其它 `xkb` 选项。该锁定阻止用户通过设置诸如 **Alt+Shift** 的按键组合来切换键盘布局。

## 13.3. 禁用命令行

如需彻底禁用命令行，您需要在很多不同环境中更改配置：

- ✦ 设置“`org.gnome.desktop.lockdown.disable-command-line`”GSettings 键值，这将阻止用户进入终端或指定运行某个命令行（**Alt+F2** 命令提示符）。
- ✦ 通过修改 X 服务器配置，禁止用 **Ctrl+Alt+function key** 快捷方式切换至虚拟终端（VT）。
- ✦ 移除**终端**，对于其它任何提供从“应用程序”菜单到终端路径，和提供从 GNOME Shell 系统中“活动概览”到终端路径的应用程序，也一并移除。这可以通过移除这些应用程序的菜单项实现。关于如何移除菜单项的详细信息，请参见〈[第 12.1.2 节 “为所有用户移除某个菜单项”](#)〉。

### 13.3.1. 设置 `org.gnome.desktop.lockdown.disable-command-line` 键值

1. 在 `/etc/dconf/db/local.d/00-lockdown` 中创建一个用于计算机范围设置的 `local` 数据库。

```
[org/gnome/desktop/lockdown]
# Disable command line access
disable-command-line=true
```

2. 覆盖用户设置，并阻止用户在 `/etc/dconf/db/local.d/locks/lockdown` 中对此修改：

```
# Lock the disabled command line access
/org/gnome/desktop/lockdown
```

3. 更新系统数据库：

```
# dconf update
```

### 13.3.2. 禁止虚拟终端切换

通常用户可以用 `Ctrl+Alt+function key` 快捷方式（例如 `Ctrl+Alt+F2`）从 GNOME 桌面和 X 服务器切换到一个虚拟终端。您可以通过向 `/etc/X11/xorg.conf.d/` 目录下 X 配置文件中的 `Serverflags` 部分添加一个 `DontVTSwitch` 选项，来禁止到所有虚拟终端的通路。

#### 过程 13.3. 禁止进入虚拟终端

1. 在 `/etc/X11/xorg.conf.d/` 目录下创建或编辑一个 X 配置文件：

```
Section "Serverflags"

Option "DontVTSwitch" "yes"

EndSection
```

2. 重启 X 服务器，以使您的更改生效。

## 第 14 章 会话管理

### 14.1. 何为 GDM？

“GNOME 显示管理器”（“GDM”）是一个在后台运行的图形登录程序，它运行并管理本地和远程登录的 X 服务器。

“GDM”代替 X 显示管理器“XDM”。然而，“GDM”并不是从“XDM”衍生而来，也不包括“XDM”的任何原始代码。另外，在“GDM”中不支持图形配置工具，因此要更改“GDM”设置，就需要编辑 `/etc/gdm/custom.conf` 配置文件。

#### 14.1.1. 重启 GDM

在更改系统配置时，例如设置登录界面标题消息、登录界面标识或登录界面背景时，您需要重启“GDM”，以使您的更改生效。



#### 警告

请牢记，强制重启服务会使所有已登录的桌面用户当前正在运行的 GNOME 会话中断。这可能导致用户丢失还未保存的数据。

如需重启“GDM”服务，请运行以下命令：

```
# systemctl restart gdm.service
```

关于红帽企业版 Linux 7 管理服务的更多信息，请参见《红帽企业版 Linux 7 系统管理员指南》，可从 [http://access.redhat.com/site/documentation/Red\\_Hat\\_Enterprise\\_Linux/](http://access.redhat.com/site/documentation/Red_Hat_Enterprise_Linux/) 中获得。

### 14.2. 认证

#### 14.2.1. 使用企业凭证登录 GNOME

如果您的网络有一个可用的 Active Directory 或 IPA 域，并且您拥有域账户，那么可以使用域凭证登录 GNOME。

如果计算机已为域账户成功配置，用户可以使用他们的账户登录 GNOME。在登录提示处输入域用户名，后接一个 @ 符号，然后输入域名称。例如，如果您的域名称是 `example.com`，用户名是 `User`，就输入：

```
User@example.com
```

在计算机已经为域账户配置好的情况下，建议您查看描述登录格式的有用提示。

##### 14.2.1.1. 在欢迎界面中选择使用企业凭证

如果您尚未配置好计算机以进行企业身份认证，可以在“欢迎”界面完成此操作，该界面是 GNOME 初始设置程序的一部分。

#### 过程 14.1. 配置企业凭证

1. 在“登录”欢迎界面，选择“使用企业版登录”。

2. 如果“域”信息栏还未填写，请在其中输入您的域名称。
3. 在相关信息栏输入您的域账户用户和密码。
4. 点击“下一步”。

根据域的配置方式，可能会出现一个要求输入域管理员名字和密码的提示符，以便进一步处理。

#### 14.2.1.2. 改为使用企业凭证登录 GNOME

如果您早已完成了初始设置，并希望开启域账户来登录 GNOME，您可以在位于 GNOME 设置中的用户面板上实现该步骤。

#### 过程 14.2. 配置企业凭证

1. 点击顶栏上您的名字，从菜单中选择“设置”。
2. 从项目清单中选择“用户”。
3. 点击“解锁”按钮，并输入计算机管理员的密码。
4. 在窗口左下方点击“[+]”按钮。
5. 选择“企业版登录”窗格。
6. 输入您企业账户的域、用户和密码，然后点击“添加”。

根据您的域配置方式，可能会出现一个要求输入域管理员名字和密码的提示符，以便进一步处理。

#### 14.2.1.3. 疑难解答与高级设置

可以用 `realm` 命令及其各种各样的子命令，来解答（troubleshoot）关于企业版登录功能的疑难问题。例如，如需检查计算机是否已经为企业版登录配置好，运行以下命令：

```
$ realm list
```

鼓励网络管理员提前加入相关域的工作站。为此可以使用重启 `realm join` 命令，或透过脚本自动运行 `realm join`。

#### 14.2.1.4. 获得更多信息

- <http://www.freedesktop.org/software/realmd/docs/> – 该领域的《管理指南》提供了更多关于使用企业版登录功能的详细信息。

#### 14.2.2. 启用指纹认证

如需允许用户使用已登记的指纹登录，请设置 `org.gnome.login-screen.enable-fingerprint-authentication` GSettings 键值。

#### 过程 14.3. 启用指纹认证

1. 在 `/etc/dconf/db/gdm.d/00-fingerprint` 中创建一个用于系统范围设置的 `gdm` 数据库。

```
[org/gnome/login-screen]
# Enable fingerprint authentication
enable-fingerprint-authentication=true
```



2. 更新系统数据库：

```
# dconf update
```

以下 GSettings 键值也需要考虑：

**org.gnome.login-screen.enable-password-authentication**

将该键值设置为假（false），使用户总是使用其它方式登录（指纹和智能卡）。

关于智能卡认证的更多信息，请参见〈[第 14.2.3 节 “启用智能卡认证”](#)〉。

### 14.2.3. 启用智能卡认证

如需允许用户使用智能卡登录，请设置 **org.gnome.login-screen.enable-smartcard-authentication** GSettings 键值。

#### 过程 14.4. 启用智能卡认证

1. 在 `/etc/dconf/db/gdm.d/00-smartcard` 中创建一个用于系统范围设置的 **gdm** 数据库。

```
[org/gnome/login-screen]
# Enable smart card authentication
enable-smartcard-authentication=true
```

2. 更新系统数据库：

```
# dconf update
```

以下 GSettings 键值也需要考虑：

**org.gnome.login-screen.enable-password-authentication**

将该键值设置为假（false），使用户总是使用其它方式登录（指纹和智能卡）。

关于指纹认证的更多信息，请参见〈[第 14.2.2 节 “启用指纹认证”](#)〉。

## 14.3. 用户会话

### 14.3.1. 什么是用户会话中的典型流程？

在一个 GNOME 储备会话中，名为 *守护程序 (daemon)* 的程序在系统上作为后台过程运行。您会发现以下守护程序是默认运行的：

**dbus-daemon**

**dbus-daemon** 提供了信息总线守护程序，它可以用来在程序之间交换消息。**dbus-daemon** 和 D-Bus 库一起执行，后者在任意两个应用程序之间提供一对一的交流。

更多信息请参见 **dbus-daemon** (1) 手册页。

**gnome-keyring-daemon**

通过“**gnome-keyring-daemon**”安全地存储着各类程序和网站的凭据，如用户名和密码。该信息被写入一份名为 **keyring** 文件的加密文件中，并存储于用户的家目录中。

更多信息请参见 **gnome-keyring-daemon** (1) 手册页。

### **gnome-session**

在 **GDM** 等显示管理器的帮助下，**gnome-session** 程序负责运行 GNOME 桌面环境。为用户安排的默认会话在安装系统时由系统管理员设定。一般情况下 **gnome-session** 会加载上一次系统成功运行的会话。

更多信息请参见 **gnome-session** (1) 手册页。

### **gnome-settings-daemon**

“**gnome-settings-daemon**”负责 GNOME 会话设置以及该会话中运行的所有程序。

更多信息请参见 **gnome-settings-daemon** (1) 手册页。

### **gnome-shell**

“**gnome-shell**”提供 GNOME 的核心用户界面功能，例如启动程序、浏览目录、查看文件等等。

更多信息请参见 **gnome-shell** (1) 手册页。

### **pulseaudio**

**PulseAudio** 是红帽企业版 Linux 的声音服务器，它允许程序通过 **Pulseaudio** 守护程序输出声音。

更多信息请参见 **pulseaudio** (1) 手册页。

根据用户的设置，您可能还会见到包括以下这些在内的守护程序：

- » “**dconf-service**”
- » “**ibus**”
- » “**at-spi2-dbus-launcher**”
- » “**at-spi2-registryd**”
- » “**gnome-shell-calendar-server**”
- » “**goa-daemon**”
- » “**gsd-printer**”
- » 各种各样的 **Evolution** 中心进程
- » 各种各样的“**GVFS**”进程

## **14.3.2. 配置用户默认会话**

默认会话从一个名为 **AccountService** 的程序中检索得到。**账户服务**将此信息存储于 **/var/lib/AccountService/users/** 目录下。



## 注意

在 GNOME 2 中，用户家目录里的 `.dmrc` 文件用来创建默认会话。现在 `.dmrc` 文件已经不再使用。

### 过程 14.5. 为用户规定默认会话

1. 请确保您已经通过运行以下命令安装了 `gnome-session-xsession` 软件包：

```
# yum install gnome-session-xsession
```

2. 前往 `/usr/share/xsessions` 目录，在那里您可以找到每个可用会话的 `.desktop` 文件。查阅 `.desktop` 文件的内容，来确定您想要使用的会话。
3. 如需为用户规定一个默认会话，请更新 `/var/lib/AccountsService/users/username file` 中用户的“账户服务”。

```
[User]
Language=
XSession=gnome
```

在此样本文件中，通过 `/usr/share/xsessions/gnome.desktop` 文件将 GNOME 设为默认会话。请注意在红帽企业版 Linux 7 中的默认系统是 GNOME 经典版（`/usr/share/xsessions/gnome-classic.desktop` 文件）。

为用户规定了默认会话之后，除非用户从登录界面选择不同的会话，否则在用户下次登录时会使用该默认会话。

### 14.3.3. 创建自定义会话

如需创建您自己自定义配置的会话，请执行以下步骤：

1. 在 `/etc/X11/sessions/new-session.desktop` 中创建一个 `.desktop` 文件。请确保该文件规定了以下条目：

```
[Desktop Entry]
Encoding=UTF-8
Type=Application
Name=Custom Session
Comment=This is our custom session
Exec=gnome-session --session=new-session
```

**Exec** 项可能通过参数规定了要执行的命令。您可以通过 `gnome-session --session=new-session` 命令运行自定义会话。

要了解更多关于您可以通过 `gnome-session` 使用的参数信息，请参见 `gnome-session (1)` 手册页。

2. 在 `/usr/share/gnome-session/sessions/new-session.session` 中创建一个自定义会话文件，您可以在其中规定会话的名字和所需组件：

```
[GNOME Session]
Name=Custom Session
RequiredComponents=gnome-shell-classic;gnome-settings-daemon;
```

请注意，您在 **RequiredComponents** 中规定的所有项目需要在 `/usr/share/applications/` 中有其对应的 `.desktop` 文件。

配置自定义会话文件之后，可以在“**GDM**”登录界面的会话列表中找到该新的会话。

#### 14.3.4. 查看用户会话日志

如果您想找到关于用户会话问题的更多信息，可以查看“**systemd**”日志。因为红帽企业版 Linux 7 是一个基于“**systemd**”的系统，用户会话日志数据以二进制格式直接存储在“**systemd**”日志中。



#### 注意

在红帽企业版 Linux 6 中，用户会话日志数据存储在 `~/.xsession-errors` 文件中，现在该文件已不再使用。

#### 过程 14.6. 查看用户会话日志

1. 通过运行以下命令确定您的用户 ID (**uid**)：

```
$ id --user
1000
```

2. 查看以上确定的用户 ID 的分类日志：

```
$ journalctl _UID=1000
```

##### 14.3.4.1. 获得更多信息

- ✧ **journalctl** (1) 手册页提供了关于“**systemd**”日志使用情况的更多信息。
- ✧ 关于在红帽企业版 Linux 7 上使用“**systemd**”日志的进一步信息，请参见《红帽企业版 Linux 7 系统管理员指南》，可从 [http://access.redhat.com/site/documentation/Red\\_Hat\\_Enterprise\\_Linux/](http://access.redhat.com/site/documentation/Red_Hat_Enterprise_Linux/) 中获得。

#### 14.3.5. 为所有用户添加自动启动的应用程序

如需在用户登录时自动启动某个应用程序，您需要在 `/etc/xdg/autostart/` 目录下为该应用程序创建一个 `.desktop` 文件。

如需为个别用户管理自动启动的应用程序，请使用 **gnome-session-properties** 应用程序。

#### 过程 14.7. 为所有用户添加一个自动启动的应用程序

1. 在 `/etc/xdg/autostart/` 目录下创建一个 `.desktop` 文件：

```
[Desktop Entry]
Type=Application
Name=Files
Exec=nautilus -n
```

```
OnlyShowIn=GNOME;
AutostartCondition=GSettings org.gnome.desktop.background show-
desktop-icons
```

2. 用应用程序的名字代替 *Files*。
3. 用您想用来运行该应用程序的命令代替 *nautilus -n*。
4. 您可以使用 **AutostartCondition** 键值来查看 GSettings 键值的内容。

如果键值为正确的话，会话管理器将自动运行应用程序。如果在当前运行的会话中键值发生改变，会话管理器会根据之前的键值来判断启动或停止应用程序。

### 14.3.6. 设置自动登录

拥有“管理员”账户类型的用户可以从 GNOME“设置”中的“用户”面板上启用“自动登录”。系统管理员也可以按以下步骤在“GDM”自定义配置文件中手动设置自动登录：

#### 例 14.1. 为用户 john 设置自动登录

编辑 `/etc/gdm/custom.conf` 文件并确保文件中的 **[daemon]** 部分指定了以下内容：

```
[daemon]
AutomaticLoginEnable=True
AutomaticLogin=john
```

用您希望自动登录的用户代替 *john*。

### 14.3.7. 设置自动退出

可自动结束闲置一定时间的用户会话。您可基于计算机是靠电池运行还是靠主电源运行，通过设置对应的 GSettings 键值并锁定，来设定不同的行为。



#### 警告

请牢记，如果自动结束闲置会话，用户可能会丢失尚未存储的数据。

#### 过程 14.8. 连接电源的计算机自动退出的设置

1. 在 `/etc/dconf/db/local.d/00-autologout` 中创建一个用于计算机范围设置的 **local** 数据库。

```
[org/gnome/settings-daemon/plugins/power]
# Set the timeout to 900 seconds when on mains power
sleep-inactive-ac-timeout=900
# Set action after timeout to be logout when on mains power
sleep-inactive-ac-type='logout'
```

2. 覆盖用户设置，并阻止用户在 `/etc/dconf/db/local.d/locks/autologout` 中对此修改：

```
# Lock automatic logout settings
```

```
/org/gnome/settings-daemon/plugins/power/sleep-inactive-ac-timeout
/org/gnome/settings-daemon/plugins/power/sleep-inactive-ac-type
```

### 3. 更新系统数据库：

```
# dconf update
```

以下 GSettings 键值需要考虑：

**org.gnome.settings-daemon.plugins.power.sleep-inactive-ac-timeout**

如果计算机是连接交流电源运行，则它至少需要不活跃这么多秒，才会进入睡眠。

**org.gnome.settings-daemon.plugins.power.sleep-inactive-ac-type**

如果计算机是连接交流电源运行，超时后会如何。

**org.gnome.settings-daemon.plugins.power.sleep-inactive-battery-timeout**

如果计算机是靠电源运行，则它至少需要不活跃这么多秒，才会进入睡眠。

**org.gnome.settings-daemon.plugins.power.sleep-inactive-battery-type**

如果计算机是靠电池运行，超时后会发生什么。

您可以对某个键值运行 **gsettings range** 命令，来查看可以使用的一系列值。例如：

```
$ gsettings range org.gnome.settings-daemon.plugins.power sleep-inactive-ac-
type
enum
'blank'
'suspend'
'shutdown'
'hibernate'
'interactive'
'nothing'
'logout'
```

## 14.3.8. 设置屏幕亮度和闲置时间

通过设置以下 GSettings 键值，您可以配置亮度下降程度，并设置亮度和闲置时间。

### 例 14.2. 设置亮度下降程度

如需设置计算机闲置一定时间后屏幕亮度下降的程度，请在 **/etc/dconf/db/local.d/00-power** 中创建一个用于计算机范围设置的 **local** 数据库，如下例中所示：

```
[org/gnome/settings-daemon/plugins/power]
idle-dim=true
```

### 例 14.3. 设置亮度

如需改变亮度，请如下例中所示，在 **/etc/dconf/db/local.d/00-power** 中创建一个用于计算机范围设置的 **local** 数据库，并用您想要使用的整数值代替 30：

```
[org/gnome/settings-daemon/plugins/power]
idle-brightness=30
```

#### 例 14.4. 设置闲置时间

如需设置触发屏幕空白和显示默认屏保的闲置时间，请如下例中所示，在 `/etc/dconf/db/local.d/00-session` 中创建一个用于计算机范围设置的 **local** 数据库，并用您想要使用的整数值代替 `900`：

```
[org/gnome/desktop/session]
idle-delay=900
```

请以 `root` 身份运行 `dconf update` 命令，将您的更改合并到系统数据库中。



#### 注意

您也可以锁定上述设置，以阻止用户改变它们。请参见 [〈第 9.5.1 节 “锁定特定设置”〉](#)。

### 14.3.9. 在用户闲置计算机时锁定屏幕

如果想在用户闲置计算机时自动锁定屏幕并启用屏保，您需要创建一个 `dconf` 配置文件、设置 `GSettings` 键值对，并对其锁定以防止用户编辑。

#### 过程 14.9. 启用屏保和锁定屏幕

1. 在 `/etc/dconf/db/local.d/00-screensaver` 中创建一个用于系统范围设置的 **local** 数据库：

```
[org/gnome/desktop/session]
# Set the lock time out to 180 seconds before the session is
considered idle
idle-delay=180
[org/gnome/desktop/screensaver]
# Set this to true to lock the screen when the screensaver activates
lock-enabled=true
# Set the lock timeout to 180 seconds after the screensaver has been
activated
lock-delay=180
```

2. 覆盖用户设置，并阻止用户在 `/etc/dconf/db/local.d/locks/screensaver` 中对此修改：

```
# Lock desktop screensaver settings
/org/gnome/desktop/session/idle-delay
/org/gnome/desktop/screensaver/lock-enabled
/org/gnome/desktop/screensaver/lock-delay
```

3. 更新系统数据库：

```
# dconf update
```

### 14.3.10. 屏幕录像记录

GNOME Shell 的一大特色是有一个内置的屏幕录像记录器，因此用户得以在会话中录制桌面或应用程序的活动，并将录制结果归类为 **webm** 格式的高分辨率视频文件。

#### 过程 14.10. 制作屏幕录像

1. 如需开始录制，请按 **Ctrl+Alt+Shift+R**。

在记录器捕捉屏幕活动时，屏幕右下角会显示一个红色的圈。

2. 如需停止录制，请按 **Ctrl+Alt+Shift+R**。屏幕右下角的红圈会消失。
3. 前往 **~/Videos** 文件夹，在那里您可以找到录制的视频，它的文件名以 **Screencast** 开头，并包括录制的日期和时间。

请注意，内置记录器总是捕捉整个屏幕，包括多显示器设置中的所有显示器。



## 第 15 章 虚拟文件系统和磁盘管理

### 15.1. GVFS

“GVFS”（“GNOME 虚拟文件系统”）是对虚拟文件系统界面的拓展，由建立 GNOME 桌面的库提供。“GVFS”提供虚拟文件系统完整的基础结构，并在 GNOME 桌面处理存储空间。

“GVFS”根据 URI（统一资源标识符）标准用地址进行完整的身份认证，它在语法上类似于网页浏览器中使用的 URL 地址。这些 `schema://user@server/path` 格式的地址是确定服务类型的核心信息。

### 15.2. GVFS 后端

“GVFS”中有许多后端，它们提供通往某特定类型资源的路径。以下列表是可用的“GVFS”后端及其说明：

表 15.1. 可用的后端

后端	描述
afc	与 MTP（媒体传输协议（Media Transfer Protocol））相似，在您的苹果电子设备上显示文件（通过 USB 连接）。
afp	Apple 文件协议（AFP）客户端，用于获得 Mac 操作系统 X 和初代 Mac 操作系统的文件服务。
archive	以只读形式处理各种存档文件（ZIP，TAR）。
burn	一个虚拟后端，被刻录程序用来作为新型 CD/DVD/BD 媒体内容的临时存储空间。
cdda	通过不同的波形音频格式（WAV）文件，显示音频 CD。
computer	用于合并活跃装载和物理卷的虚拟后端，它与路标的行为相似。之前 Nautilus 将它用于“计算机”视图。
dav, davs	WebDAV 客户端，包括安全变量。仅可在装载期间进行认证，不支持之后基于每个文件夹的重新认证。
dns-sd	DNS 服务探索 – Avahi 客户端，在浏览网页时使用，为已探索到的服务生成连续的 URI。
ftp	一个极具特色的 FTP（文件传输协议（File Transfer Protocol））客户端，目前没有 FTPS 的支持。默认支持被动传输。
gphoto2	图片传输协议（PTP）客户端，用来访问通过 USB 或 FireWire 连接的相机。
http	处理所有 HTTP 请求；有助于在客户端应用程序中简便地从网站下载文件。
locatest	简单的测试后端代理文件:/// URI；支持错误输入。
mtp	用于访问媒体播放器和智能手机记忆卡的媒体转移协议后端。
network	用于浏览网页，显示附近的 Avahi 和 Samba 服务器。
obexftp	蓝牙客户端
recent	在“GtkFileChooser”中使用的一个后端，用于列出 GNOME 应用程序最近使用的文件。
sftp	一个极具特色的 SFTP（SSH 文件传输协议）客户端。
smb	访问 Samba 和 Windows 共享
trash	一个回收站后端，允许恢复被删除的文件。



#### 注意

有些后端单独打包，并没有默认安装。如需安装其它后端，请使用“yum”软件包管理器。

如需使用后端的的服务，必须形成 URI 字符串。该字符串是“GVFS”中使用的一个基本标识符，它携带了唯一识别所需的所有必要信息，例如服务类型（后端 ID）、绝对路径、用户名（如果需要的话）。您可以在 **Nautilus** 的地址栏与 GTK+ 的【打开】或【保存】文件对话中看到这项信息。

以下示例是 URI 字符串的一个非常基本的形式，它指向一个在 *ftp.myserver.net* 域内运行的 FTP（文件传输协议）服务器的根目录（/）：

#### 例 15.1. 指向根目录的 URI 字符串

```
ftp://ftp.myserver.net/
```

下例通过认证，指向某个特定路径中的文本文件：

#### 例 15.2. 指向文本文件的 URI 字符串

```
ssh://joe@ftp.myserver.net/home/joe/todo.txt
```

## 15.3. 装载、卸载和弹出

在虚拟文件系统中，某些特定资源被设置为自动装载，但最常见的方式是手动触发装载。

### 过程 15.1. 手动装载

1. 在 **Nautilus** 中打开一个文件（即文件应用程序）。
2. 在屏幕顶端的地址栏输入一个结构良好的 URI 字符串。如果地址栏没有显示，请按 **Ctrl+L**。  
或者，**Nautilus** 提供了“连接服务器”对话，您可以通过进入“文件 → 连接服务器”找到它。
3. 当系统要求输入登录凭证时，在相应的输入框中输入您的用户名和密码。
4. 装载过程一结束，您就可以运行文件。

如果您需要卸载这些资源，请遵循以下简单的步骤。

### 过程 15.2. 卸载

1. 点击所选装载的弹出图标。
2. 等待装载消失或显示安全移除的通知。



#### 重要

出于系统运作的原因，数据可能被高速缓存或被缓慢写入背景中。为了在设备或远程资源上安全地传递数据，绝不要拔出或断开资源。

应用程序之间共享装载，并在当前运行的桌面会话中对其进行全域追踪，这意味着即使您退出了触发装载的某个应用程序，其它所有应用程序还是可以访问该装载。通过同样的方式，多个应用程序可以同时访问某个装载，除非有后端限制了该装载（有些协议由于自身设计只能允许单通道）。

**重要**

“GVFS”装载（还有物理卷）仅面向它们的拥有者，其他用户无法侵犯您的隐私。

## 15.4. Nautilus 和【打开】或【保存】文件对话书签

您可以通过添加书签，保存引用地址。

**过程 15.3. 为某个地址添加书签：**

1. 选择您想要添加书签的文件夹或文件。
2. 请按 **Ctrl+D**。

书签首次被激活时，“GVFS”子系统首先寻找现存装载，如果没找到再生成新装载。通过这种方式，您只要在【打开】或【保存】对话中就完成验证。

“书签”被很好地合并并在 GTK+ 和 GNOME 桌面上：每个显示标准 GTK+【打开】或【保存】对话（技术上称为“GtkFileChooser”）的应用程序都会在对话左侧版面列出书签。“Nautilus”及其复制品也会在边栏（或者更普遍的，在“文件”菜单中）显示书签。

**注意**

如果您还没有页面书签，“书签”标签并不显示。

除了“书签”，其它所有可用的“GVFS”卷和装载都列在“GtkFileChooser”的边栏中。有时，书签和“GVFS”卷合并成单独一项，以防止重复和混淆。这样一来，“书签”可以像“GVFS”装载一样拥有弹出图标。

“书签”位于 `~/.config/gtk-3.0/bookmarks` 文件中。在下例中，书签位置是 `~/Music`、`~/Pictures`、`~/Videos`、`~/Downloads` 和 `~/bin`，因此 `~/.config/gtk-3.0/bookmarks` 文件的内容看起来如下所示：

**例 15.3. `~/.config/gtk-3.0/bookmarks` 文件**

```
file:///home/username/Music
file:///home/username/Pictures
file:///home/username/Videos
file:///home/username/Downloads
file:///home/username/bin
```

用您想要使用的用户名代替 `username`。

**过程 15.4. 编辑书签：**

1. 打开顶栏中的文件菜单。
2. 点击“书签”，打开书签编辑器。

## 15.5. 设置默认服务器列表

## 15.5. 从系统管理员的角度看文件共享

从系统管理员的角度来看，不可能一次性为所有用户设置一堆书签。不过，系统管理员可以允许用户访问文件共享。

**Nautilus** 在 `~/.config/nautilus/servers` 文件中以 **XBEL** 格式存储了文件共享服务器的列表。可以向该文件添加文件共享服务器列表，使用户得以轻易访问文件共享。



### 注意

**XBEL** (XML 书签交换语言) 是一种允许您共享 URI (统一资源标识符) 的 XML 标准。在 **GNOME** 中，**XBEL** 用来在诸如 **Nautilus** 的应用程序中共享桌面书签。

在下例中，**Nautilus** 在 `~/.config/nautilus/servers` 文件中以 URI `ftp://ftp.gnome.org/` 创建了一个名为 **GNOME FTP** 的书签。

#### 例 15.4. `~/.config/nautilus/servers` 文件

```
<?xml version="1.0" encoding="UTF-8"?>
<xbel version="1.0"
  xmlns:bookmark="http://www.freedesktop.org/standards/desktop-
bookmarks"
  xmlns:mime="http://www.freedesktop.org/standards/shared-mime-info">
  <bookmark href="<input>ftp://ftp.gnome.org/</input>">
    <title><input>GNOME FTP</input></title>
  </bookmark>
</xbel>
```

## 15.6. 向所有其它应用程序显示 **GNOME** 虚拟文件系统

除了内置 **GIO** 库的应用程序可以访问 **GVFS** 装载之外，“**GVFS**”还提供了一个显示活跃 **GVFS** 装载的“**FUSE**”守护程序。这意味着，任何应用程序都可以使用标准 **POSIX** API 访问活跃的“**GVFS**”装载，就好像它们是常规文件系统一样。

尽管如此，在某些应用程序中，附加库的依赖性和 **VFS** 子系统的新特性可能并不匹配，或者过于复杂。出于这些原因，以及为了提高兼容性，“**GVFS**”提供了一个“**FUSE**”（“**用户空间的文件系统**”）守护程序，它通过自身用于标准 **POSIX**（便携式操作系统界面）访问的装载显示活跃装载。该守护程序透明转换传入请求，来为应用程序模拟一个本地文件系统。



### 重要

来自不同设计的转换并非 100% 功能兼容的，您可能在某些特定应用程序与“**GVFS**”后端的组合中遇到困难。

“**FUSE**”守护程序随着“**GVFS**”主控守护程序自动启动，并将其装载作为后备放置在 `/run/user/UID/gvfs` 文件或 `~/.gvfs` 文件中。手动浏览显示每个“**GVFS**”装载都有单独的目录。当您使用非本地应用程序打开来自“**GVFS**”位置的文件时，经过了一条作为参数的转换路径。请注意，本地 **GIO** 应用程序自动将这条路径转回本地 URI。

## 15.7. GVFS 装载的密码管理

除非资源允许匿名身份验证，否则根本不要求身份验证，典型的“GVFS”装载在激活时会要求身份凭证。在标准的 GTK+ 对话中，用户可以选择是否保存密码。

### 过程 15.5. 示例：已进行身份验证的装载过程

1. 打开文件，并通过按 **Ctrl+L** 激活地址栏。
2. 输入需要身份验证的服务的 URI 字符串，要求结构良好（例如，**sftp://localhost/**）。
3. 出现一个要求用户名、密码，以及密码存储选项的身份凭证对话。
4. 填入身份凭证并确认。

如果选择了永久保存，密码会被存储在用户 keyring 中。“GNOME Keyring”是存储机密的中心位置。它是加密的，并通过登录时默认提供的密码在桌面会话开始时自动解锁。如果有另一个不同的密码保护它，那么该密码是在首次使用时设置。

为了管理存储的密码和“GNOME Keyring”本身，系统提供了应用程序 **Seahorse**，它允许移除个别记录或更改密码。关于 **Seahorse** 的更多信息，请查阅直接安装在桌面上的 **Seahorse** 帮助手册。

## 15.8. GNOME 中的 GVFS 工具和 xdg-utils

“GVFS”与一些可能有助于脚本编制或测试的命令一起推出。提供一组 POSIX 命令的对应命令：

- » **gvfs-cat**
- » **gvfs-less**
- » **gvfs-mkdir**
- » **gvfs-mount**
- » **gvfs-rename**
- » **gvfs-set-attribute**
- » **gvfs-copy**
- » **gvfs-ls**
- » **gvfs-move**
- » **gvfs-rm**
- » **gvfs-trash**
- » **gvfs-info**
- » **gvfs-save**
- » **gvfs-tree**

为了更好地控制“GVFS”的特性，提供了一些附加命令：

- » **gvfs-monitor-dir**

- ✧ **gvfs-monitor-file**
- ✧ **gvfs-mime**
- ✧ **gvfs-open**

所有这些命令都是本机 GIO 客户端，不需要运行后备守护程序“**FUSE**”。它们的目的是不要成为 POSIX 命令的插入式替代，实际上，它们支持很小范围内的交换。在基本形式之下，一条 URI 字符串（而非一条本地路径）看成一个参数。

这些都使得 GNOME 在“**xdg-tools**”（一个 freedesktop.org 互操作性项目）中得到很好的支持。例如，当检测到一个正在运行的 GNOME 会话时，经常使用的“**xdg-open**”实际上会调用“**gvfs-open**”，并从正确的位置读取文件类型关联。

以下是 GVFS 命令使用情况的一些例子：

- ✧ 如需在 **/tmp** 中列出本地文件系统上的所有文件，请执行：

```
$ gvfs-ls file:///tmp
```

- ✧ 以下命令列出了来自远程计算机的文本文件的内容：

```
$ gvfs-cat ssh://joe@ftp.myserver.net/home/joe/todo.txt
```

- ✧ 如需将引用的文本文件复制到本地目录 **/tmp** 中，请运行：

```
$ gvfs-copy ssh://joe@ftp.myserver.net/home/joe/todo.txt /tmp/
```



### 注意

为了方便用户，提供“**bash**”自动完成功能作为软件包的一部分：

## 15.9. 使用 GVFS 元数据

“**GVFS**”将元数据存成一组与特定文件绑定的简单键值/值对信息。因此，用户或应用程序可以使用工具保存一些小数据，它们是为运行时间信息设计的，例如图标位置，最后使用位置，文档、徽标、笔记中的位置等等。

一旦移动文件或目录，相应的元数据也会移动，以保证它和对应文件保持联系。“**GVFS**”私下存储所有元数据，以便它仅在该计算机上可用。但是也会追踪“**GVFS**”装载和可移动媒体。



### 注意

可移动媒体现在安装在 **/run/media/** 目录下，而非 **/media** 目录下。

如需查看和操作元数据，您可以使用：

- ✧ **gvfs-info** 命令；
- ✧ **gvfs-set-attribute** 命令；
- ✧ 或其它任何处理属性的本机 GIO 方式。



下例对某个自定义元数据属性进行了设置。请注意在移动或重命名之后，特定的 **gvfs-info** 调用与数据持久性之间的差异（注意 **gvfs-info** 命令的输出）：

### 例 15.5. 设置自定义的元数据属性

```
$ touch /tmp/myfile
$ gvfs-info -a 'metadata::*' /tmp/myfile
attributes:
$ gvfs-set-attribute -t string /tmp/myfile 'metadata::mynote' 'Please
remember to delete this file!'
$ gvfs-info -a 'metadata::*' /tmp/myfile
attributes:
  metadata::mynote: Please remember to delete this file!
$ mv /tmp/myfile /tmp/newfile
$ gvfs-info -a 'metadata::*' /tmp/newfile
attributes:
  metadata::mynote: Please remember to delete this file!
```

## 15.10. 疑难解答

### 15.10.1. 活跃的 VFS 装载不可见

如果您的活跃 VFS 装载不可见，这意味着您的应用程序不是本机 GIO 客户端。通常，本机 GIO 客户端是指所有使用 GNOME 库的 GNOME 应用程序（glib、gio）。对于非 GIO 客户端，提供“**gvfs-fuse**”服务作为后备。

如需找出 VFS 装载活跃但不可见的原因，请核查“**gvfs-fuse**”过程是否在运行。由于“**gvfs-fuse**”自动运行，我们也不建议您自己启动它，请尝试退出和登录作为首选。

另一种选择是，您可以在终端手动开启 VFS 兼容性装载：

1. 通过运行 **id** 命令找出 **/run/user/UID/gvfs/** 路径的系统用户 ID“**UID**”（守护程序“**gvfsd-fuse**”需要一条用于显示其服务的路径）。

或者，如果 **/run/user/UID/gvfs/** 路径不可用，“**gvfsd-fuse**”会使用您家目录下的 **.gvfs** 路径。

2. 通过运行 **/usr/libexec/gvfsd-fuse -f /run/user/UID/gvfs** 命令，来启动“**gvfsd-fuse**”守护程序。

3. 现在，VFS 装载可供使用，且您可以在应用程序中手动浏览路径。

### 15.10.2. 连接的 USB 磁盘不可见

在某些情况下，当您连接了一个闪存驱动器时，GNOME 桌面并没有将它显示出来。如果驱动器不可见，这意味着：

- ✧ 您无法在**磁盘**应用程序中看到该设备。
- ✧ 您运行了 **udisksctl dump** 命令，它会列出守护程序“**udisks**”的当前状态，并显示除您的闪存驱动器之外所有对象的信息。

- ✱ 您运行了 **dmesg** 命令。在记录末端有一些关于 USB 设备检测的消息，以及一张除您的闪存驱动器之外的已检测到分区的列表。

如果您的闪存驱动器不可见，您可以尝试设置磁盘中的“在用户界面显示”标志：

1. 通过按 **Super** 键进入“活动概览”，并输入 **Disks**，然后按 **Enter**，来打开“磁盘”。
2. 在“卷”操作菜单中，点击“编辑装载选项...”。
3. 点击“在用户界面显示”。
4. 通过点击“OK”确认。

如果仍不显示闪存驱动器，您可以尝试移除驱动器再重新连接。

关于存储空间的更多信息，请参见《红帽企业版 Linux 7 存储空间管理指南》，网址为 [http://access.redhat.com/site/documentation/Red\\_Hat\\_Enterprise\\_Linux/](http://access.redhat.com/site/documentation/Red_Hat_Enterprise_Linux/)。

### 15.10.3. Nautilus 显示未知的或不需要的分区

由于用户界面上没有默认显示设备，请核对 **/etc/fstab** 文件中是否列有该设备。通常，**/etc/fstab** 文件会列出被设计用于操作系统中的磁盘分区，并显示它们是如何装载的。特定的装载选项可能会允许或阻止在用户界面显示该卷。

掩藏卷的办法之一，就是在磁盘应用程序中的“装载选项”窗口，取消勾选“在用户界面显示”：

1. 通过按 **Super** 键进入“活动概览”，并输入 **Disks**，然后按 **Enter**，来打开“磁盘”。
2. 在“卷”操作菜单中，点击“编辑装载选项...”。
3. 取消选中“在用户界面显示”，并通过点击“OK”确认。

### 15.10.4. 无法连接远程文件系统

在许多情况下，客户端会被意外地强制从一个虚拟文件系统（或一个远程磁盘）的装载上断开，并且之后没有自动重新连接上，而是返回了出错消息。若干原因会引发这些情况：

- ✱ 连接被打断（例如，您的手提电脑与 Wi-Fi 断开连接）。
- ✱ 用户停用了一段时间，于是被服务器断开连接（闲置超时）。
- ✱ 计算机从睡眠模式恢复。

解决方法是卸载、再重新装载文件系统，这将重新连接资源。



#### 注意

如果连接被更加频繁地断开，请核对 GNOME“设置”中“网络”面板上的设置。

### 15.10.5. 如果磁盘忙碌该怎么做？

如果收到磁盘忙碌的通知，请确定正在访问磁盘的程序，然后通过常规方法结束正在运行的程序。或者也可使用系统监视器强制终止程序。

在哪里以及如何查看系统进程？



- ✧ 请运行 **lsdf** 命令，来获得被打开文件的列表及其进程。如果 **lsdf** 不可用，请运行 **ps ax** 命令，它也会提供正在运行的进程的列表。
- ✧ 另一种选择是，您可以使用**系统监视器**应用程序，以 GUI 格式显示正在运行的进程。
- ✧ 请确保您已经通过运行以下命令安装了“**iotop**”：

```
# yum install iotop
```

接下来，请以 root 身份运行“**iotop**”，以查看系统进程。

当您已经确定程序时，按如下所示结束或移除程序：

- ✧ 在命令行执行 **kill** 命令。
- ✧ 在**系统监视器**中，右击含有程序进程名称的行，并点击“**结束进程**”或“**移除进程**”下拉菜单项。

## 第 16 章 硬件配置

### 16.1. 数位屏

#### 16.1.1. 我的 Wacom 数位屏没有映射到正确的屏幕

如需正确使用屏幕映射，“libwacom”数据库和“udev”规则文件（`/lib/udev/rules.d/50-libwacom.rules`）中必须包含对您的数位屏的支持。

“libwacom”是一个存储 Wacom 模型数据的数位屏信息客户端库。“gnome-settings-daemon”组件和 GNOME 中的 Wacom 数位屏设置面板均使用该库。

如需向“libwacom”添加对一个新的数位屏的支持，必须创建一个新的数位屏定义文件。数位屏定义文件包括在 `libwacom-data` 软件包中。如果安装了这个软件包，就可以在本地目录 `/usr/share/libwacom/` 中找到可用的数位屏定义文件。

关于“libwacom”的更多信息，请参见以下链接：

» <http://sourceforge.net/apps/mediawiki/linuxwacom/index.php?title=Libwacom>

#### 16.1.2. Wacom 数位屏的配置存储在哪？

Wacom 数位屏的配置存储在 `/org/gnome/settings-daemon/peripherals/wacom/machine-id-device-id` GSettings 键值中，其中 `machine-id` 是 D-Bus 计算机 ID，`device-id` 是数位屏设备 ID。数位屏的配置架构是 `org.gnome.settings-daemon.peripherals.wacom`。

同样的，对于支持专业范围触笔的标识符 `tool-id` 的客户，触笔配置存储在 `/org/gnome/settings-daemon/peripherals/wacom/device-id/tool-id` 键值中。对于不支持 `tool-id` 的客户范围，改为使用一般标识符。触笔的配置架构是 `org.gnome.settings-daemon.peripherals.wacom.stylus`，橡皮擦的配置架构是 `org.gnome.settings-daemon.peripherals.wacom.eraser`。

可通过使用由 `gnome-settings-daemon-devel` 软件包提供的 `gsd-list-wacom` 工具，获得某一特定计算机上所使用的数位屏配置路径的完整列表。

如需验证系统上安装有 `gnome-settings-daemon-devel` 软件包，请确认该系统支持 **Optional** 通道，并运行以下命令：

```
# yum install gnome-settings-daemon-devel
```

如需学习如何使系统支持 **Optional** 通道，请阅读以下资源：

» <https://access.redhat.com/site/solutions/70019>

确认软件包已安装之后，请运行以下命令：

```
$ /usr/libexec/gsd-list-wacom
```

请注意，在配置路径中使用 `machine-id`、`device-id` 和 `tool-id` 会允许每台计算机与独立的数位屏配置共享家目录。

#### 16.1.3. 不同计算机之间共享家目录时，Wacom 设置仅应用于一台计算机

这是因为 `/org/gnome/settings-daemon/peripherals/wacom/machine-id-device-id` GSettings 键值的配置路径中包括 Wacom 数位屏的 D-Bus 计算机 ID (*machine-id*)，该配置路径中存储着您的数位屏配置。

## 附录 A. 访问红帽文档

位于 <https://access.redhat.com/site/documentation/> 的“红帽产品文档”是信息来源中心。目前它被翻译成 22 种语言，并为每一种产品提供了以 HTML、PDF 和 EPUB 格式发行的发行注记和技术注记、安装指南、用户指南和参考指南等不同类型的书。

以下是与本书有着直接或间接关系的文档的概要列表：

- ✦ 《红帽企业版 Linux 7 系统管理员指南》，可从 [http://access.redhat.com/site/documentation/Red\\_Hat\\_Enterprise\\_Linux/](http://access.redhat.com/site/documentation/Red_Hat_Enterprise_Linux/) 获得，它包括了关于各种系统组件的详细信息，如“GRUB 2”启动加载程序、软件包管理、“systemd”或打印机配置。
- ✦ 《红帽企业版 Linux 7 安装指南》，可从 [http://access.redhat.com/site/documentation/Red\\_Hat\\_Enterprise\\_Linux/](http://access.redhat.com/site/documentation/Red_Hat_Enterprise_Linux/) 获得，它包括了关于安装红帽企业版 Linux 7 和使用 Anaconda 安装器的详细信息。
- ✦ 《红帽企业版 Linux 7 迁移计划指南》，可从 [http://access.redhat.com/site/documentation/Red\\_Hat\\_Enterprise\\_Linux/](http://access.redhat.com/site/documentation/Red_Hat_Enterprise_Linux/) 获得，它包括了对红帽企业版 Linux 6 和红帽企业版 Linux 7 在运行状况与兼容性上主要变化的概览。迁移计划指南也介绍了红帽提供的工具，帮助红帽企业版 Linux 7 升级。
- ✦ 《红帽企业版 Linux 7 网络指南》，可从 [http://access.redhat.com/site/documentation/Red\\_Hat\\_Enterprise\\_Linux/](http://access.redhat.com/site/documentation/Red_Hat_Enterprise_Linux/) 获得，它包括了用于红帽企业版 Linux 7 的网络配置和管理的信息。
- ✦ 《红帽企业版 Linux 7 虚拟化部署与管理指南》，可从 [http://access.redhat.com/site/documentation/Red\\_Hat\\_Enterprise\\_Linux/](http://access.redhat.com/site/documentation/Red_Hat_Enterprise_Linux/) 获得，它包括了关于安装、配置和管理红帽企业版 Linux 虚拟化的信息。

## 附录 B. 修订历史

<b>修订 0.1-69.1</b>	<b>Thu Apr 30 2015</b>	<b>Chester Cheng</b>
说明：翻译、校对完成。 翻译、校对：柳叶。（1 至 11 章） 翻译、校对：吴敏峰。（12 至 16 章、附录） 校对、编辑：任浩。 校对、责任编辑：郑中。 附注：本简体中文版来自「Red Hat 全球服务部」与「澳大利亚昆士兰大学笔译暨口译研究所」之产学合作计划。若有疏漏之处，盼各方先进透过以下网址，给予支持指正： <a href="https://bugzilla.redhat.com/">https://bugzilla.redhat.com/</a> 。		
<b>修订 0.1-69</b>	<b>Mon 02 Jun 2014</b>	<b>Petr Kovář</b>
发布红帽企业版 Linux 7.0 指南		
<b>修订 0.1-35</b>	<b>Wed 11 Dec 2013</b>	<b>Petr Kovář</b>
发布测试版红帽企业版 7.0 指南		