

Desktop Menu Specification

Version 1.1

Waldo Bastian <waldo.bastian@intel.com>
Francois Gouget <fgouget@codeweavers.com>
Alex Graveley <alex@ximian.com>
George Lebl <jirka@5z.com>
Havoc Pennington <hp@pobox.com>
Heinrich Wendel <h_wendel@cojobo.net>

20 August 2016

Содержание

1	Расположение файлов	3
2	Расширения формата для ярлыков рабочего стола	4
2.1	Примеры использования категорий и OnlyShowIn	5
3	Формат файлов меню	6
3.1	Декларация типа документа	6
3.2	Элементы	6
3.2.1	<Menu>	6
3.2.2	<AppDir>	7
3.2.3	<DefaultAppDirs>	7
3.2.4	<DirectoryDir>	7
3.2.5	<DefaultDirectoryDirs>	8
3.2.6	<Name>	8
3.2.7	<Directory>	8
3.2.8	<OnlyUnallocated> и <NotOnlyUnallocated>	8
3.2.9	<Deleted> и <NotDeleted>	9
3.2.10	<Include>	9
3.2.11	<Exclude>	9
3.2.12	<Filename>	9

3.2.13	<Category>	9
3.2.14	<All>	9
3.2.15	<And>	10
3.2.16	<Or>	10
3.2.17	<Not>	10
3.2.18	<MergeFile [type="path"parent"] >	10
3.2.19	<MergeDir>	11
3.2.20	<DefaultMergeDirs>	11
3.2.21	<LegacyDir>	11
3.2.22	<KDELegacyDirs>	11
3.2.23	<Move>	12
3.2.24	<Old>	12
3.2.25	<New>	12
3.2.26	<Layout>	12
3.2.27	<DefaultLayout [show_empty="false"] [inline="false"] [inline_limit="4"] [inline_header="true"] [inline_alias="false"]>	12
3.2.28	<MenuName [show_empty="..."] [inline="..."] [inline_limit="..."] [inline_header="..."] [inline_alias="..."]>	13
3.2.29	<Separator>	13
3.2.30	<Merge type="menus"files"all"/>	13
4	Слияние	14
5	Генерация меню	15
6	Иерархии устаревших меню	16
7	Пример файла меню	19
8	Приложения	20
8.1	Приложение А. Зарегистрированные категории	20
8.1.1	Основные категории	20
8.1.2	Дополнительные категории	20
8.1.3	Зарезервированные категории	27

8.2	Приложение Б. Окружения зарегистрированные OnlyShowIn	27
8.3	Приложение В. Интеграция вашего приложения в меню	27
8.3.1	Добавление пунктов меню	27
8.3.2	Места установки	29
8.3.3	Пример	29
8.3.4	Обратная совместимость	31
8.4	Приложение Г. Замечания по реализации	31
8.4.1	Редактирование меню	31

9 Глоссарий 32

Аннотация

Этот документ определяет, как построить видимую пользователем иерархию приложений, обычно отображаемую в виде меню. Такая иерархическая структура позволяет стороннему программному обеспечению добавлять пункты меню, которые работают для всех рабочих столов, и позволяет системным администраторам редактировать меню таким образом, чтобы это влияло на все рабочие столы.

Основная схема очень проста. Информация о каждом приложении (элемент меню) хранится в ярлыке рабочего стола (см. «Стандарт элементов рабочего стола»). Затем файл конфигурации XML определяет иерархическое расположение пунктов меню и какие пункты меню фактически отображаются.

Ситуация несколько усложняется необходимостью поддержки устаревших иерархий записей рабочего стола и необходимостью разрешить третьим сторонам расширять структуру меню. Обе эти проблемы решаются с помощью идеи объединения двух макетов меню.

В дополнение к строгому определению содержания каждого меню эта спецификация также предусматривает ряд советов по макету / представлению. Эта часть спецификации является необязательной, реализации могут предпочесть игнорировать эти подсказки.

1 Расположение файлов

Файлы, включенные в эту спецификацию, расположены в соответствии со «Спецификацией каталогов базового рабочего стола».

Вот файлы, определенные этой спецификацией:

```
$XDG_CONFIG_DIRS/menus/${XDG_MENU_PREFIX}applications.menu
```

Этот файл содержит определение XML макета главного меню приложения. Первый файл, найденный в пути поиска, должен использоваться; другие файлы игнорируются. Это подразумевает, что если пользователь имеет свое собственное *\$XDG_MENU_PREFIX application.menu*, оно заменяет общесистемное. (Хотя пользовательское меню может явно присоединить общесистемное.)

Системы, которые предлагают несколько сред рабочего стола и которые хотят использовать различные макеты меню в разных средах, могут использовать файлы *.menu* с разными префиксами. В этом случае переменная окружения *\$XDG_MENU_PREFIX* должна быть установлена системой чтобы отражать используемый файл меню.

Например, если система содержит среды рабочего стола GNOME и KDE, она может решить использовать *gnome-applications.menu* в качестве макета меню в сеансах GNOME и *kde-applications.menu* в качестве макета меню в сеансах KDE. Чтобы правильно отразить это, следует установить для переменной среды *\$XDG_MENU_PREFIX* значение «gnome-» или «kde-».

Реализации могут использовать файлы *.menu* с другими именами для задач или меню, кроме основного меню приложения. Такое использование не охватывается этой спецификацией.

```
$XDG_CONFIG_DIRS/menus/applications-merged/
```

Каталоги слияния по умолчанию, включены в элемент *<DefaultMergeDirs>*. По соглашению третьей стороны могут добавлять новые файлы *<Menu>* в это место для создания своих собственных подменю.

Обратите внимание, что система, которая использует *gnome-applications.menu* или *kde-applications.menu* в зависимости от используемой среды рабочего стола, должна по-прежнему использовать приложения слияния в качестве каталога слияния по умолчанию в обоих случаях.

Реализации могут использовать файлы *.menu* с именами, отличными от *application.menu*, для задач или меню, кроме основного меню приложения. В этом случае первая часть имени каталога слияния по умолчанию получена из имени файла *.menu*.

Например, в системе, использующей файл *preferences.menu* для описания дополнительного меню, каталоги слияния по умолчанию, включенные в элемент «*DefaultMergeDirs*» в файле *preferences.menu*, станут *\$XDG_CONFIG_DIRS/menus/preferences-merged/*

```
$XDG_DATA_DIRS/applications/
```

Этот каталог содержит файл *.desktop* для каждого возможного пункта меню. Следует использовать каждый каталог в пути поиска *\$XDG_DATA_DIRS* (то есть записи рабочего стола собираются из всех, а не только из первого существующего). Когда две записи рабочего стола имеют одинаковое имя, используется та, которая появилась ранее в пути.

Элемент *<DefaultAppDirs>* в файле меню указывает, что этот список местоположений записей рабочего стола по умолчанию должен быть проверен в этой точке. Если файл меню не содержит *<DefaultAppDirs>*, то эти местоположения не сканируются.

```
$XDG_DATA_DIRS/desktop-directories/
```

Этот каталог содержит записи каталога, которые могут быть связаны с папками в макете меню. Каждый каталог в пути поиска должен быть использован. Используются только файлы, оканчивающиеся на *.directory*; другие файлы игнорируются.

Элемент *<DefaultDirectoryDirs>* в файле меню указывает на то, что этот список местоположений записей каталога по умолчанию следует сканировать в этой точке. Если файл меню не содержит *<DefaultDirectoryDirs>*, то эти местоположения не сканируются.

2 Расширения формата для ярлыков рабочего стола

Эта спецификация добавляет три новых поля для ярлыков рабочего стола: *Categories*, *OnlyShowIn* и *NotShowIn*.

Поле *Categories* представляет собой список строк, используемых для классификации пунктов меню. Например, приложения в категории *AudioVideo* могут оказаться в подменю «Звук и видео». В приложении А 8.1: «Зарегистрированные категории» перечисляет стандартные категории. Категории, не указанные в этом документе, должны начинаться с префикса «X-», указывающего, что они являются расширениями. Категории чувствительны к регистру.

Элементы меню рабочего стола должны перечислять все категории, которые явно применяются. Они не должны перечислять категории, которые применяются только расплывчато или возможно, потому что пользователь в конечном итоге увидит ту же запись на рабочем столе в полдюжине мест. Но часто бывает, что несколько категорий будут применяться к одной записи меню рабочего стола.

Поле *OnlyShowIn* представляет собой список строк, определяющих среды, в которых должен отображаться данный пункт меню. Если присутствует поле *OnlyShowIn*, данная среда должна отображать пункт меню, только если строка, идентифицирующая эту среду. Строки чувствительны к регистру. Приложение Б 8.2, *Registered OnlyShowIn Environments* перечисляет строки для использования в некоторых распространенных средах.

Поле *NotShowIn* представляет собой список строк, определяющих среды, в которых не должен отображаться данный элемент меню. Если присутствует поле *NotShowIn*, данная среда должна отображать пункт меню, только если строка, идентифицирующая эту среду, отсутствует. Строки чувствительны к регистру. Приложение В 8.3, "Окружения, зарегистрированные как *OnlyShowIn*" перечисляет строки для использования в некоторых распространенных средах.

Среды, не указанные в этом документе, должны иметь префикс «X-», указывающий на то, что они являются расширениями. Среды чувствительны к регистру.

2.1 Примеры использования категорий и OnlyShowIn

Ярлык рабочего стола для средства просмотра изображений на основе Qt может содержать следующую строку категорий:

```
Categories=Qt;Graphics;RasterGraphics;Viewer;
```

Ярлык рабочего стола для Octave, математической программы командной строки (которая также будет иметь поле *Terminal=true*), может иметь такие категории:

```
Categories=ConsoleOnly;Math;
```

Ярлык рабочего стола для программы калькулятора для GNOME, которая должна появляться только в GNOME, может содержать:

```
Categories=GNOME;Utility;  
OnlyShowIn=GNOME;
```

Обратите внимание, что поле *OnlyShowIn* является списком и поэтому заканчивается точкой с запятой.

3 Формат файлов меню

Файлы меню должны быть правильно сформированными файлами XML и заканчиваться расширением «.menu». Они также должны соответствовать DTD файла меню, что подразумевает, что специфичные для реализации расширения формата файла не допускаются. Реализации могут остановить обработку, если они сталкиваются с файлом меню, который не соответствует связанному DTD. Обратите внимание, что связанный DTD может отличаться по версии от той, которая определена в этом документе.

Когда реализация обновляет существующий файл меню, может потребоваться обновить идентификатор до более новой версии DTD. Реализации никогда не должны обновлять идентификатор существующего файла меню до более старой версии. Чтобы оставаться совместимым с более новыми версиями, реализации должны игнорировать и сохранять любые элементы XML, атрибуты и значения атрибутов, с которыми он не знает, как обращаться.

3.1 Декларация типа документа

Файлы меню для этой версии спецификации должны использовать следующие идентификаторы пространства имен, общедоступные и системные:

Пространство имен

`http://www.freedesktop.org/standards/menu`

Public идентификатор для версии 1.1

`PUBLIC "-//freedesktop//DTD Menu 1.0//EN"`

Системный идентификатор для 1.1

`http://www.freedesktop.org/standards/menu-spec/menu-1.0.dtd`

Вот пример объявления типа документа:

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

Все файлы меню **ДОЛЖНЫ** включать декларацию типа документа, чтобы реализации могли адаптироваться к различным версиям этой спецификации (и, таким образом, реализации могли проверять файл меню по DTD).

3.2 Элементы

3.2.1 <Menu>

Корневым элементом является <Menu>. Каждый элемент <Menu> может содержать любое количество вложенных элементов <Menu>, указывая подменю.

3.2.2 <AppDir>

Этот элемент может появиться только под <Menu>. Содержимое этого элемента является именем каталога. Ярлыки рабочего стола в этом каталоге сканируются и добавляются в пул ярлыков, которые могут быть включены в это <Menu> и его подменю. Следует использовать только файлы, оканчивающиеся на «.desktop», остальные файлы игнорируются.

Ярлыки рабочего стола в пуле доступных ярлыков идентифицируются по их идентификатору рабочего стола (см. Идентификатор рабочего стола). Идентификатор файла рабочего стола для записи на рабочем столе равен имени файла с удалением всех компонентов пути. Таким образом, при наличии <AppDir>/foo/bar и ярлыка рабочего стола /foo/bar/Hello.desktop ярлык рабочего стола получит идентификатор файла рабочего стола Hello.desktop

Если каталог содержит подкаталоги, то эти подкаталоги также должны (рекурсивно) сканироваться. Имя подкаталога должно быть добавлено в качестве префикса к идентификатору файла рабочего стола вместе с символом тире (“-“). Поэтому с учетом <AppDir>/foo/bar и ярлыка рабочего стола /foo/bar/booz/Hello.desktop ярлык рабочего стола получит идентификатор файла рабочего стола booz-Hello.desktop Ярлык рабочего стола /foo/bar/bo/oz/Hello.desktop приведет к идентификатору файла рабочего стола bo-oz-Hello.desktop

Элементы <AppDir>, появляющиеся позже в файле меню, имеют приоритет в случае коллизий между идентификаторами файлов рабочего стола.

Если имя файла, заданное как <AppDir>, не является абсолютным путем, оно должно быть задано относительно местоположения анализируемого файла меню.

Дублирующиеся элементы <AppDir> (которые указывают один и тот же каталог) следует игнорировать, но последний дубликат в файле следует использовать при определении порядка сканирования каталогов. Это важно при объединении (см. Раздел «Объединение»). Порядок элементов <AppDir> относительно элементов <Include> и <Exclude> не имеет значения, также для облегчения слияния.

3.2.3 <DefaultAppDirs>

Этот элемент может отображаться только под <Меню>. Элемент не имеет содержимого. Элемент должен обрабатываться так, как если бы он был списком элементов <AppDir>, содержащих местоположения папок приложения по умолчанию (*datadir/apps/* и т.д.). При развертывании <DefaultAppDirs> в список <AppDir> местоположения по умолчанию, находящиеся ранее в пути поиска, переходят позже в <Menu>, с приоритетом.

3.2.4 <DirectoryDir>

Этот элемент может отображаться только под <Меню>. Содержимое этого элемента является именем каталога. В каждом каталоге, указанном в элементе <DirectoryDir>, будет производиться поиск записей каталога, которые будут использоваться при определении элемента <Directory> для этого меню и его подменю. Если имя файла, заданное как <DirectoryDir>, не является абсолютным путем, оно должно быть указано относительно местоположения анализируемого файла меню.

Записи каталога в пуле доступных записей идентифицируются по их относительному пути (см. Relative path9).

Если две записи каталога имеют повторяющиеся относительные пути, необходимо использовать элемент из последнего (самого дальнего вниз) элемента в файле меню. Должны быть загружены только файлы, заканчивающиеся расширением «.directory», другие файлы должны игнорироваться.

Дублирующиеся элементы *<DirectoryDir>* (которые указывают один и тот же каталог) обрабатываются как дублирующиеся элементы *<AppDir>* (используется последний дубликат).

3.2.5 *<DefaultDirectoryDirs>*

Этот элемент может отображаться только под *<Меню>*. Элемент не имеет содержимого. Элемент должен обрабатываться так, как если бы он был списком элементов *<DirectoryDir>*, содержащих стандартные каталоги рабочего стола (*datadir/desktop-directory/* и т.д.). Местоположения по умолчанию, которые находятся ранее в пути поиска, переходят позже в *<Menu>*, поэтому они имеют приоритет.

3.2.6 *<Name>*

Каждый элемент *<Menu>* должен иметь один элемент *<Name>*. Содержимое элемента *<Name>* - это имя, которое будет использоваться при обращении к данному меню. Каждое подменю данного *<Menu>* должно иметь уникальное имя. Таким образом, на элементы *<Menu>* можно ссылаться по пути меню, например «Приложения/Графика». Поле *<Name>* не должно содержать косую черту ("/"); реализации должны отбрасывать любое имя, содержащее косую черту. Смотрите также *Menu path*9.

3.2.7 *<Directory>*

Каждый элемент *<Menu>* имеет любое количество элементов *<Directory>*. Содержимое элемента *<Directory>* - это относительный путь к файлу описателя каталога, содержащего метаинформацию о *<Menu>*, такую как его значок и локализованное имя. Если *<Directory>* не указано для *<Menu>*, его поле *<Name>* должно использоваться как видимое пользователем имя меню.

Повторяющиеся элементы *<Directory>* разрешены для упрощения слияния меню и позволяют пользовательским меню переопределять системные меню. Последний элемент *<Directory>*, который появляется в файле меню, «выигрывает», а другие элементы игнорируются, если только последний элемент не указывает на несуществующую запись в каталоге, в этом случае вместо этого следует попробовать предыдущий элемент и т.д.

3.2.8 *<OnlyUnallocated>* и *<NotOnlyUnallocated>*

Каждый элемент *<Menu>* может содержать любое количество элементов *<OnlyUnallocated>* и *<NotOnlyUnallocated>*. Действует только последний такой элемент, который появляется, поскольку он определяет, может ли *<Menu>* содержать какие-либо записи рабочего стола или только те записи рабочего стола, которые не соответствуют другим меню. Если нет ни элементов *<OnlyUnallocated>*, ни *<NotOnlyUnallocated>*, по умолчанию используется значение *<NotOnlyUnallocated>*.

Для обработки *<OnlyUnallocated>* файл меню должен быть проанализирован в два концептуальных этапа. При первом прохождении обрабатываются элементы *<Menu>*, которые могут соответствовать любой записи рабочего стола. Во время этого прохода каждая запись рабочего стола помечается как выделенная в соответствии с тем, соответствует ли она правилу *<Include>* в некотором *<Menu>*. Второй этап обрабатывает только элементы *<Menu>*, которые ограничены нераспределенными записями рабочего стола. Во время второго прохода запросы могут соответствовать только тем записям на рабочем столе, которые не были помечены как выделенные во время первого прохода. См. Раздел «Создание меню».

3.2.9 <Deleted> и <NotDeleted>

Каждый элемент <Menu> может содержать любое количество элементов <Deleted> и <NotDeleted>. Актуален только последний такой элемент, который появляется, поскольку он определяет, было ли удалено <Menu>. Если нет ни элементов <Deleted>, ни <NotDeleted>, по умолчанию используется значение <NotDeleted>. Целью этого элемента является поддержка редактирования меню. Если меню содержит элемент <Deleted>, за которым не следует элемент <NotDeleted>, это меню следует игнорировать.

3.2.10 <Include>

Элемент <Include> представляет собой набор правил, пытающихся сопоставить некоторые из известных записей рабочего стола. Элемент <Include> содержит список любого количества соответствующих правил. Правила соответствия указываются с использованием элементов <And>, <Or>, <Not>, <All>, <Filename> и <Category>. Каждое правило в списке правил имеет логическое отношение ИЛИ по отношению к остальным правилам, то есть ярлыкам рабочего стола, которые соответствуют любому правилу, включаются в меню.

Элементы <Include> должны появляться сразу под элементами <Menu>. Записи рабочего стола, которые они соответствуют, включены в меню. Элементы <Include> и <Exclude> для данного <Menu> обрабатываются по порядку, причем запросы в начале файла обрабатываются первыми. Это имеет значение для слияния, см. Раздел «Слияние» 4. См. Раздел «Создание меню» 5 для получения полной информации о том, как обрабатывать элементы <Include> и <Exclude>.

3.2.11 <Exclude>

Любое количество элементов <Exclude> может отображаться под элементом <Menu>. Содержимое элемента <Exclude> представляет собой список соответствующих правил, как и в случае с <Include>. Однако соответствующие ярлыки рабочего стола удаляются из списка включенных в него ярлыков рабочего стола. (Таким образом, элемент <Exclude>, который появляется перед любыми элементами <Include>, не будет иметь никакого эффекта, например, поскольку записи на рабочем столе еще не были включены.)

3.2.12 <Filename>

Элемент <Filename> является наиболее простым правилом соответствия. Соответствует ярлыку рабочего стола, если ярлык рабочего стола имеет заданный идентификатор файла рабочего стола. Смотрите *Desktop-File Id*9.

3.2.13 <Category>

Элемент <Category> - это еще один базовый предикат соответствия. Он соответствует ярлыку рабочего стола, если ярлык рабочего стола имеет данную категорию в своем поле *Category*.

3.2.14 <All>

Элемент <All> - это правило соответствия, которое соответствует всем ярлыкам на рабочем столе.

3.2.15 <And>

Элемент <And> содержит список правил соответствия. Если каждое из правил соответствия внутри элемента <And> соответствует ярлыку рабочего стола, тогда все правило <And> соответствует ярлыку рабочего стола.

3.2.16 <Or>

Элемент <Or> содержит список правил соответствия. Если какое-либо из правил соответствия внутри элемента <Or> соответствует ярлыку рабочего стола, тогда все правило <Or> соответствует этому ярлыку рабочего стола.

3.2.17 <Not>

Элемент <Not> содержит список правил соответствия. Если какое-либо из правил соответствия внутри элемента <Not> соответствует ярлыку рабочего стола, тогда все правило <Not> не соответствует записи рабочего стола. То есть правила соответствия ниже <Not> имеют логическое отношение ИЛИ.

3.2.18 <MergeFile [type="path"parent"] >

Любое количество элементов <MergeFile> может быть указано ниже элемента <Menu>, указывая имя другого файла меню, который будет объединен с этим. Раздел под названием «Merge» определяет, как выполняется слияние. Корень <Menu> объединенного файла будет объединен с непосредственным родителем элемента <MergeFile>. Элемент <Name> корневого <Menu> объединенного файла игнорируется.

Если атрибут `type` отсутствует или имеет значение «*path*», то содержимое элемента <MergeFile> указывает файл, который необходимо присоединить. Если это не абсолютный путь, тогда файл, который нужно присоединить, должен быть расположен относительно местоположения файла меню, содержащего этот элемент <MergeFile>.

Дублирующиеся элементы <MergeFile> (которые указывают один и тот же файл) обрабатываются так же, как дублирующиеся элементы <AppDir> (используется последний дубликат).

Если для атрибута `type` установлено значение «*parent*», а файл, содержащий этот элемент <MergeFile>, расположен под одним из путей, указанных в `$XDG_CONFIG_DIRS`, содержимое элемента следует игнорировать, а искать в остальных путях, указанных в `$XDG_CONFIG_DIRS`, для файла с тем же относительным именем файла. Первый найденный файл должен быть присоединен. Не должно быть никакого слияния вообще, если не найдено ни одного подходящего файла.

Примечание о совместимости: имя файла, указанное внутри элемента <MergeFile>, следует игнорировать, если для атрибута `type` установлено значение «*parent*», однако следует ожидать, что реализации, основанные на предыдущих версиях этой спецификации, будут игнорировать атрибут `type` и что такие реализации будут использовать вместо этого имя файла внутри элемента <MergeFile>.

Пример 1: если `$XDG_CONFIG_HOME` равен `./config/` а `$XDG_CONFIG_DIRS` равен `"/opt/gnome:/etc/xdg"` а файл `./config/menus/Applications.menu` содержит <MergeFile `type="parent"/opt/kde3/etc/xdg/menus/Applications.menu`> то файл `/opt/gnome/menus/applications.menu` должен быть присоединен, если он существует. Если этот файл не существует, вместо него следует присоединить файл `/etc/xdg/menus/applications.menu`.

Пример 2: если `$XDG_CONFIG_HOME` равен `~/.config/` а `$XDG_CONFIG_DIRS` равен `"/opt/gnome:/etc/xdg/"` а файл `/opt/gnome/menus/applications.menu` содержит `<MergeFile type="parent" /opt/kde3/etc/xdg/menus/Applications.menu /etc/xdg/menus/applications.menu` то файл `/etc/xdg/menus/applications.menu` должен быть присоединен, если он существует.

3.2.19 <MergeDir>

Любое количество элементов `<MergeDir>` может быть указано ниже элемента `<Menu>`. `<MergeDir>` содержит имя каталога. Каждый файл в данном каталоге, который заканчивается расширением `.menu` должен быть объединен так же, как и `<MergeFile>`. Если имя файла, заданное как `<MergeDir>`, не является абсолютным путем, оно должно быть расположено относительно местоположения анализируемого файла меню. Файлы внутри объединенного каталога не объединяются в указанном порядке.

Дублирующиеся элементы `<MergeDir>` (которые указывают один и тот же каталог) обрабатываются как с дублирующимися элементами `<AppDir>` (используется последний дубликат).

3.2.20 <DefaultMergeDirs>

Этот элемент может отображаться только под `<Menu>`. Элемент не имеет содержимого. Элемент должен обрабатываться так, как если бы это был список элементов `<MergeDir>`, содержащих местоположения каталога слияния по умолчанию. При развертывании `<DefaultMergeDirs>` в список `<MergeDir>` местоположения по умолчанию, находящиеся ранее в пути поиска, переходят в конец `<Menu>`, чтобы иметь приоритет.

3.2.21 <LegacyDir>

Этот элемент может появиться только под `<Menu>`. Текстовое содержимое этого элемента является именем каталога. Каждый каталог, указанный в элементе `<LegacyDir>`, будет представлять собой устаревшую иерархию ярлыков рабочего стола, о том, как загрузить такую иерархию, см. Раздел «Иерархии устаревшего меню»б. Реализации не должны загружать устаревшие иерархии, которые явно не указаны в файле меню (поскольку, например, файл меню может не быть главным меню). Если имя файла, заданное как `<LegacyDir>`, не является абсолютным путем, оно должно быть расположено относительно местоположения анализируемого файла меню.

Дублирующиеся элементы `<LegacyDir>` (которые указывают один и тот же каталог) обрабатываются так же как дублирующиеся элементы `<AppDir>` (используется последний дубликат).

Элемент `<LegacyDir>` может иметь один атрибут, префикс. Обычно при наличии `<LegacyDir>/foo/bar` и ярлыка рабочего стола `/foo/bar/baz/Hello.desktop` ярлык рабочего стола получает идентификатор файла рабочего стола `Hello.desktop`. Учитывая префикс `boo-`, ему вместо этого будет присвоен идентификатор файла рабочего стола `boo-Hello.desktop`. Префикс не должен содержать символов разделителя пути (`'/'`).

3.2.22 <KDELegacyDirs>

Этот элемент может появиться только под `<Menu>`. Элемент не имеет содержимого. Элемент должен обрабатываться так, как если бы он был списком элементов `<LegacyDir>`, содержащих традиционные расположения файлов рабочего стола, поддерживаемые KDE, с жестко заданным префиксом «`kde-`». При развертывании `<KDELegacyDirs>` в список `<LegacyDir>` местоположения, находящиеся ранее в пути поиска, переходят в конец `<Menu>`, чтобы иметь приоритет. Путь поиска можно получить, запустив `kde-config --path apps`

3.2.23 <Move>

Этот элемент может появиться только под <Menu>. Элемент <Move> содержит пары элементов <Old>/<New>, указывающие, как переименовать потомка текущего <Menu>. Если путь назначения уже существует, перемещенное меню объединяется с меню назначения (подробнее см. Раздел «Объединение»).

<Move> используется в основном для исправления старых каталогов. Предположим, например, что вы объединяете <LegacyDir> с именами папок, которые не соответствуют текущей иерархии; устаревшие имена папок могут быть перемещены в новые имена, где они будут объединены с новыми папками.

<Move> также полезен для осуществления редактирования меню, см. Раздел «Редактирование меню»5.

3.2.24 <Old>

Этот элемент может отображаться только под <Move> и должен сопровождаться элементом <New>. Содержимое как <Old>, так и <New> должно быть путем к меню (объединенная косая черта полей <Name>, см. Путь к меню). Пути интерпретируются относительно меню, содержащего элемент <Move>.

3.2.25 <New>

Этот элемент может появляться только под <Move>, и перед ним должен стоять элемент <Old>. Элемент <New> указывает новый путь для предыдущего элемента <Old>.

3.2.26 <Layout>

Элемент <Layout> является необязательной частью этой спецификации. Реализации, которые не поддерживают элемент <Layout>, должны максимально сохранить любые элементы <Layout> и их содержимое. Каждый <Menu> может дополнительно содержать элемент <Layout>. Если появляется несколько элементов, то релевантным является только последний такой элемент. Целью этого элемента является предложение по представлению меню. Если в меню нет элемента <Layout> или если он содержит пустой элемент <Layout>, то следует использовать макет по умолчанию. Элемент <Layout> может содержать элементы <Filename>, <MenuName>, <Separator> и <Merge>. Элемент <Layout> определяет предложенный макет для меню, начиная сверху вниз. Ссылки на ярлыки рабочего стола, которые не содержатся в этом меню, как это определено элементами <Include> и <Exclude>, следует игнорировать. Ссылки на подменю, которые не содержатся непосредственно в этом меню, как определено элементами <Menu>, должны игнорироваться.

3.2.27 <DefaultLayout [show_empty="false"] [inline="false"] [inline_limit="4"] [inline_header=[inline_alias="false"]>

Элемент <DefaultLayout> является необязательной частью этой спецификации. Реализации, которые не поддерживают элемент <DefaultLayout>, должны максимально сохранять любые элементы <DefaultLayout> и их содержимое. Каждый элемент <Menu> может дополнительно содержать элемент <DefaultLayout>, который определяет макет по умолчанию для текущего меню и всех его подменю. Если в меню есть элемент <DefaultLayout>, это переопределит любой макет по умолчанию, указанный родительским меню. Макет по умолчанию определяет предлагаемый макет, если элемент <Menu> не

имеет элемента `<Layout>` или если он имеет пустой элемент `<Layout>`. Для объяснения различных атрибутов смотрите элемент `<MenuName>`. Если макет по умолчанию не был указан, то следует предполагать макет, заданный следующими элементами:

```
<DefaultLayout show_empty = "false"
    inline = "false" inline_limit = "4"
    inline_header = "true"
    inline_alias = "false">
  <Merge type = "menus" />
  <Merge type = "files" />
</DefaultLayout>
```

3.2.28 `<MenuName [show_empty="..."] [inline="..."] [inline_limit="..."] [inline_header="..."] [inline_alias="..."]>`

Этот элемент может появляться только как дочерний элемент меню `<Layout>` или `<DefaultLayout>`. Его содержимое ссылается на непосредственное подменю текущего меню, как определено с элементом `<Menu>`, поэтому оно никогда не должно содержать косую черту. Если такого подменю не существует, элемент следует игнорировать. Этот элемент может иметь различные атрибуты, значения по умолчанию берутся из ключа `DefaultLayout`. Атрибут `show_empty` определяет, должно ли вообще отображаться меню, которое не содержит ярлыков рабочего стола и никаких подменю. Атрибут `show_empty` может иметь значение «true» или «false». Он может иметь встроенный атрибут, который может иметь значение «true» или «false». Если встроенный атрибут имеет значение «true», то упомянутое меню может быть скопировано в текущее меню в текущей точке, а не вставлено как подменю текущего меню. Необязательный атрибут `inline_limit` определяет максимальное количество записей, которые могут быть встроены. Если подменю содержит больше записей, чем `inline_limit`, подменю не будет встроено. Если `inline_limit` равен 0 (ноль), то ограничения нет. Необязательный атрибут `inline_header` определяет, должна ли встроенному меню предшествовать запись заголовка, содержащая заголовок подменю. Атрибут `inline_header` может иметь значение «true» или «false». Необязательный атрибут `inline_alias` определяет, должна ли одна встроенная запись принимать заголовок встроенного меню. В этом случае никакая дополнительная запись заголовка не будет добавлена независимо от значения атрибута `inline_header`. Атрибут `inline_alias` может иметь значение «true» или «false». Пример: если в меню есть подменю под названием «WordProcessor» с одной записью «OpenOffice 4.2», и указаны оба значения `inline = «true»` и `inline_alias = «true»`, то это приведет к тому, что запись «OpenOffice 4.2» будет встроено в текущее меню, но заголовок «OpenOffice 4.2» будет заменен на «WordProcessor».

3.2.29 `<Separator>`

Этот элемент может появляться только как дочерний элемент меню `<Layout>` или `<DefaultLayout>`. Он указывает на предложение нарисовать визуальный разделитель в этой точке меню. Элементы `<Separator>` в начале меню, в конце меню или непосредственно после других элементов `<Separator>` могут игнорироваться.

3.2.30 `<Merge type="menus"files"all"/>`

Этот элемент может появляться только как дочерний элемент меню `<Layout>` или `<DefaultLayout>`. Он указывает точку, куда должны быть вставлены ярлыки и подменю рабочего стола, которые явно не упомянуты в элементе `<Layout>` или `<DefaultLayout>`. Он имеет атрибут `type`, который указывает, какие элементы должны быть вставлены: `type = "menus"` означает, что все подменю, которые не упомянуты явно, должны быть вставлены в алфавитном порядке их визуального заголовка на этом

этапе. *type = "files"* означает, что все элементы рабочего стола, содержащиеся в этом меню, которые явно не упомянуты, должны быть вставлены в алфавитном порядке их визуального заголовка на этом этапе. *type = "all"* означает, что на данном этапе смесь всех подменю и всех записей рабочего стола, которые явно не упомянуты, должны быть вставлены в алфавитном порядке их визуального заголовка. Каждый элемент *<Layout>* или *<DefaultLayout>* должен иметь ровно один элемент *<Merge type = "all">* или он должен содержать ровно один элемент *<Merge type = "files">* и ровно один элемент *<Merge type = "menus">*. Исключение сделано для полностью пустого элемента *<Layout>*, который может использоваться для указания того, что вместо этого следует использовать макет по умолчанию.

4 Слияние

Иногда необходимо объединить две схемы меню. Это делается при сворачивании в иерархиях устаревшего меню (см. Раздел «Иерархии устаревшего меню»), а также для файлов, указанных в элементах *<MergeFile>*. Распространенным случаем является то, что файлы меню для пользователя могут объединять файл системного меню. Объединение также используется, чтобы избежать вырезания и вставки, например, для включения общего подменю в несколько файлов меню.

Объединение включает в себя базу *<Меню>* и объединенное *<Menu>*. Основой является «целевое» меню, и к нему добавляется объединенное *<Menu>*. Результат слияния называется «комбинированным меню».

Цель подготовительного шага состоит в том, чтобы преобразовать все файлы в элементы XML. Для этого пройдите по всему дереву меню. Для каждого элемента *<MergeFile>*, *<MergeDir>* или *<LegacyDir>* замените элемент *<MergeFile>*, *<MergeDir>* или *<LegacyDir>* дочерними элементами корневого *<Menu>* объединяемых файлов. В качестве специального исключения удалите элемент *<Name>* из корневого элемента каждого объединяемого файла. Чтобы создать *<Menu>* на основе *<LegacyDir>*, см. Раздел «Иерархии устаревшего меню» 6.

Продолжайте обработку, пока больше не останется элементов *<MergeFile>*, *<MergeDir>* или *<LegacyDir>*, стараясь избежать бесконечных циклов, вызванных файлами, которые ссылаются друг на друга.

После того как все файлы были загружены в одно дерево, отсканируйте дерево рекурсивно, выполнив эти шаги для удаления дубликатов:

1. Объедините дочерние меню. Каждая группа дочерних *<Menu>* с одинаковыми именами должна быть объединена в одно дочернее меню с таким именем. Объедините дочерние элементы всех меню с одинаковыми именами в порядке их появления и вставьте эти элементы в качестве дочерних элементов последнего меню с этим именем. Удалите все ещё пустые элементы *<Menu>*, оставив последний.
2. Разверните элементы *<DefaultAppDirs>* и *<DefaultDirectoryDirs>* до элементов *<AppDir>* и *<Directory>*. Объедините дубликаты элементов *<AppDir>*, *<DirectoryDir>* и *<Directory>*, сохранив последний. Для элементов *<Directory>*, которые ссылаются на отдельные записи каталога, все они должны быть сохранены - если последний указывает на несуществующий файл, вместо него можно использовать предыдущий, и так далее.
3. Вернитесь к каждому ребенку *<Menu>*, выполнив этот список шагов для каждого ребенка по порядку.

После однократного повторения для удаления дубликатов повторяем удаление, чтобы разрешить элементы *<Move>* для каждого меню, начиная с любого дочернего меню, прежде чем обрабатывать меню более высокого уровня. Поэтому в самых глубоких меню сначала выполняются операции *<Move>*. Внутри каждого *<Menu>* выполните операции *<Move>* в порядке их появления. Если путь назначения

не существует, просто переместите исходный элемент `<Menu>` и измените его поле `<Name>`, чтобы оно соответствовало пути назначения. Если исходный путь не существует, ничего не делать. Если оба пути существуют, возьмите исходный элемент `<Menu>`, удалите его элемент `<Name>` и добавьте оставшиеся дочерние элементы к целевому элементу `<Menu>`.

Если какие-либо операции `<Move>` влияют на меню, повторите шаги для устранения дубликатов, если дубликаты были созданы.

Наконец, для каждого `<Menu>`, содержащего элемент `<Deleted>`, за которым не следует элемент `<NotDeleted>`, удалите это меню и все его дочерние меню.

Объединенные элементы меню сохраняются по порядку, потому что элементы `<Include>` и `<Exclude>` встретившиеся в файле позже переопределяют элементы `<Include>` и `<Exclude>` встретившиеся в файле ранее. Это означает, что если файл меню пользователя объединяется с файлом системного меню, то пользователь всегда может переопределить то, что указывает системное меню, помещая элементы после `<MergeFile>`, который включает системный файл.

Во избежание ситуации, когда ярлык рабочего стола от одной стороны непреднамеренно отменяет ярлык рабочего стола от другой стороны, так как оба имеют один и тот же идентификатор файла рабочего стола, рекомендуется, чтобы поставщики файлов рабочего стола гарантировали, что все идентификаторы файлов рабочего стола начинаются с префикса поставщика. Префикс поставщика состоит из $[a - zA - Z]$ и заканчивается тире (“-“). Проектам с открытым исходным кодом и коммерческим сторонам рекомендуется использовать слово или фразу, предпочтительно их имя, в качестве префикса, для которого они имеют товарный знак. Приложения с открытым исходным кодом могут также запросить использование префикса поставщика другого проекта с открытым исходным кодом (такого как GNOME или KDE), с которым они считают себя связанным, по усмотрению этих проектов.

Например, чтобы обеспечить запуск приложений GNOME с префиксом поставщика «gnome-», он может либо добавить «gnome-» ко всем файлам рабочего стола, которые он устанавливает в `datadir/apps/`, либо установить файлы рабочего стола в подкаталог `datadir/apps/gnome`. При включении устаревших иерархий меню аргумент префикса элемента `<LegacyDir>` может использоваться для указания префикса.

5 Генерация меню

После объединения меню, результатом должно стать описание макета одного меню. Для каждого `<Menu>` у нас есть список каталогов, в которых можно найти ярлыки рабочего стола, список каталогов, в которых можно найти описатели каталога, и ряд директив `<Include>` и `<Exclude>`.

Для каждого элемента `<Menu>` создайте пул ярлыков рабочего стола, собрав ярлыки, найденные в каждом `<AppDir>` для элемента меню. Если два ярлыка имеют одинаковый идентификатор файла рабочего стола, запись для более раннего (ближе к началу файла) `<AppDir>` должна быть отброшена. Затем добавьте в пул ярлыки из всех `<AppDir>`, определенных элементами `<Menu>` предка. Если родительское меню имеет повторяющиеся ярлыки (тот же идентификатор файла рабочего стола), ярлык для дочернего меню имеет приоритет.

Затем пройдите по всем операторам `<Include>` и `<Exclude>`. Для каждого `<Include>` сопоставьте правила с пулом всех ярлыков рабочего стола. Для каждого ярлыка рабочего стола, который соответствует одному из правил, добавьте его в отображаемое меню и пометьте как выделенный. Для каждого `<Exclude>` сопоставьте правила с включенными в данный момент ярлыками рабочего стола. Для каждого соответствующего ярлыка на рабочем столе, удалите его из меню. Обратите внимание, что ярлык, который включена в меню, но снова исключен последующим `<Exclude>`, все еще считается выделенным (для целей `<OnlyUnallocated>`), даже если этот ярлык больше не отображается в меню.

Необходимы два прохода, один раз для обычных меню, где может быть сопоставлена любой ярлык, и один раз для меню `<OnlyUnallocated>`, где могут быть сопоставлены только ярлыки, которые не были помечены как выделенные.

Результатом, конечно же, является дерево ярлыков рабочего стола.

6 Иерархии устаревших меню

Традиционно, меню определялись как иерархия каталогов и файлов в файловой системе, причем каждый каталог файловой системы соответствовал подменю. Реализации этой спецификации должны иметь возможность загружать эти иерархии старого стиля, как указано в этом разделе.

Общий подход таков: прежняя иерархия преобразуется в `<Menu>`, а затем этот макет меню объединяется с меню, в котором указан `<LegacyDir>`.

Ярлыки рабочего стола в устаревшей иерархии должны быть добавлены в пул ярлыков рабочего стола, как если бы `<LegacyDir>` был `<AppDir>`. Записи каталога в устаревшей иерархии должны быть добавлены в пул записей каталога, как если бы `<LegacyDir>` был `<DirectoryDir>`. Это можно реализовать тривиально, добавив соответствующие операторы `<AppDir>` и `<DirectoryDir>` в корень устаревшее `<Menu>`. Есть одна небольшая сложность, а именно атрибут «prefix» в `<LegacyDir>`.

Структура меню концептуально соответствует следующему описанию, хотя на самом деле генерировать XML не нужно:

- Для каждого каталога в устаревшей иерархии создается `<Menu>` с тем же `<Name>`, что и каталог на диске.
- Это меню содержит элемент `<Include>`, который включает каждый ярлык рабочего стола в этом каталоге. То есть он должен иметь `<Filename>Foo/Bar/foo.desktop</Filename>` для каждого ярлыка рабочего стола в каталоге.

В качестве особого исключения, если ярлык рабочего стола в каталоге содержит поле «Категории», то эта запись рабочего стола не должна включаться в устаревшее меню. То есть, элемент `<Include>` не должен генерироваться для этого ярлыка. Это позволяет установить ярлык рабочего стола в устаревшем месте, но при этом оптимально работать с системой меню, указанной в этом документе.

- Если унаследованный каталог содержит файл «.directory», то должен быть создан элемент `<Directory>`, который указывает на указанный файл «.directory».
- Устаревшим ярлыкам рабочего стола не следует присваивать никакие поля категорий, если у них их еще не было, за исключением того, что всем устаревшим записям должна быть добавлена категория «Устаревшие», чтобы файлы меню могли обрабатывать их особым образом.

(Если один и тот же каталог указан как `<LegacyDir>` и `<AppDir>`, его записи на рабочем столе должны быть помечены как «Legacy», только если `<LegacyDir>` появится в файле позже, чем `<AppDir>`.)

Например, скажем, у нас есть следующая устаревшая иерархия каталогов:


```
/usr/share/applnk
  /usr/share/applnk/.directory
  /usr/share/applnk/bar.desktop
  /usr/share/applnk/System
    /usr/share/applnk/System/.directory
    /usr/share/applnk/System/foo.desktop
```

Концептуально это преобразуется в следующее <Меню>:

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

<Menu>
  <Name>Applications</Name>
  <AppDir>/usr/share/applnk</AppDir>
  <DirectoryDir>/usr/share/applnk</DirectoryDir>
  <Directory>.directory</Directory>
  <Include>
    <Filename>bar.desktop</Filename>
  </Include>
  <Menu>
    <Name>System</Name>
    <AppDir>/usr/share/applnk/System</AppDir>
    <DirectoryDir>/usr/share/applnk/System</DirectoryDir>
    <Directory>.directory</Directory>
    <Include>
      <Filename>foo.desktop</Filename>
    </Include>
  </Menu>
</Menu>
```

Это <Menu> затем объединяется, как если бы оно было в файле, и загружается с помощью <MergeFile>.

7 Пример файла меню

```
<Menu>
<Name>Applications</Name>
<Directory>Applications.directory</Directory>

<!-- Search the default locations -->
<DefaultAppDirs/>
<DefaultDirectoryDirs/>

<!-- Merge third-party submenus -->
<MergeDir>applications-merged</MergeDir>

<!-- Merge legacy hierarchy -->
<LegacyDir>/usr/share/applnk</LegacyDir>

<!-- Define default layout -->
<DefaultLayout>
  <Merge type="menus"/>
  <Merge type="files"/>
  <Separator/>
  <Menuname>More</Menuname>
</DefaultLayout>

<!-- some random moves, maybe to clean up legacy dirs,
      maybe from menu editing -->
<Move>
  <Old>Foo</Old>
  <New>Bar</New>
  <Old>Foo2</Old>
  <New>Bar2</New>
</Move>

<!-- A preferences submenu, kept in a separate file
      so it can also be used standalone -->
<Menu>
  <Name>Preferences</Name>
  <Directory>Preferences.directory</Directory>
  <MergeFile>preferences.menu</MergeFile>
</Menu>

<!-- An Office submenu, specified inline -->
<Menu>
  <Name>Office</Name>
  <Directory>Office.directory</Directory>
  <Include>
    <Category>Office</Category>
  </Include>
  <Exclude>
    <Filename>foo.desktop</Filename>
  </Exclude>
</Menu>

</Menu>
```

8 Приложения

8.1 Приложение А. Зарегистрированные категории

Этот раздел содержит ряд общеизвестных категорий и предложений по их использованию:

- Список главных категорий состоит из тех категорий, которые ДОЛЖНА поддерживать каждая соответствующая среда рабочего стола.
- Список дополнительных категорий содержит категории, которые можно использовать для предоставления более подробной информации о приложении.
- Список зарезервированных категорий содержит категории, имеющие значение для рабочего стола.

Обратите внимание, что имена категорий чувствительны к регистру.

8.1.1 Основные категории

Включив одну из главных категорий в файл ярлыка приложения на рабочем столе, приложение будет отображаться в разделе меню приложения, посвященном этой категории. Если несколько основных категорий включены в один файл ярлыка рабочего стола, запись может появляться в меню более одного раза.

Основанные на категориях меню, основанные на Основных категориях, перечисленных в этой спецификации, не обеспечивают полную онтологию для всех доступных приложений. Поэтому реализациям меню на основе категорий СЛЕДУЕТ предоставлять подменю «всеохватывающего» для приложений, которые нельзя разместить в других местах.

В таблице ниже перечислены все основные категории.

8.1.2 Дополнительные категории

В столбце «Связанные категории» перечислены одна или несколько категорий, которые предлагается использовать вместе с «Дополнительной категорией». Если столбец Связанные категории пуст, дополнительная категория может использоваться с любой основной категорией.

В таблице ниже описаны дополнительные категории.

Таблица 2: Дополнительные категории

Дополнительная категория	Описание	Связанные категории
Building	Инструменты для разработки приложений	Разработка
Debugger	Инструмент для отладки приложений	Разработка
IDE	Разработка приложений IDE	Разработка
GUIDesigner	Приложения для GUI дизайна	Разработка

Profiling	Инструмент профилирования	Разработка
RevisionControl	Приложения, такие как cvs или subversion	Разработка
Translation	Инструменты трансляции	Разработка
Calendar	Календарь	Приложения для офиса
ContactManagement	Например адресная книга	Офис
Database	Приложение для работы с базами данных	Office или Development или AudioVideo
Dictionary	Словарь	Office или TextTools
Chart	Приложения для построения диаграмм	Office
Email	Приложение электронной почты	Офис или Сеть
Finance	Финансовое приложение для управления вашими финансами	Офис
FlowChart	Приложения для построения блок-схем	Office
PDA	Инструмент для управления вашим PDA	Office
ProjectManagement	Приложение для управления проектами	Office или Development
Presentation	Приложения для построения презентаций	Office
Spreadsheet	Электронная таблица	Office
WordProcessor	Текстовый процессор	Office
2DGraphics	2D графическое приложение	Graphics
VectorGraphics	Приложение для просмотра, создания или обработки векторной графики	Graphics; 2DGraphics
RasterGraphics	Приложение для просмотра, создания или обработки растровой (растровой) графики	Graphics; 2DGraphics
3DGraphics	Приложения для просмотра, создания или обработки трехмерной графики.	Graphics

Scanning	Инструменты для сканирования файла / текста	Graphics
OCR	Приложения для оптического распознавания символов	Graphics; Scanning
Photography	Инструменты для фотоаппарата и т.д.	Graphics; Office
Publishing Desktop	Издательские приложения и инструменты управления цветом	Graphics или Office
Viewer	Инструмент для просмотра, например, графических или PDF-файлов	Graphics; Office
TextTools	Текстовые утилиты	Utility
DesktopSettings	Инструмент настройки графического интерфейса	Settings
HardwareSettings	Инструмент для управления аппаратными компонентами, такими как звуковые карты, видеокарты или принтеры.	Settings
Printing	Инструмент для управления принтерами	HardwareSettings; Settings
PackageManager	Приложения менеджера пакетов	Settings
Dialup	dial-up программа	Network
InstantMessaging	Клиент мгновенных сообщений	Network
Chat	Клиент чата	Network
IRCClient	IRC клиент	Network
Feed	Канал RSS, подкаст и другое содержимое на основе подписки	Network
FileTransfer	Инструменты, такие как FTP или P2P программы	Network
HamRadio	НАМ радио программное обеспечение	Network или Audio
News	Читалка новостей или новостной ленты	Network
P2P	P2P программа	Network
RemoteAccess	Инструмент для удаленного управления ПК	Network

Телефония	Телефония через Network	сеть
TelephonyTools	Инструменты теле- Утилиты	фонии, для набора номера, управления УАТС, ...
VideoConference	Программное обес- Network	печение для видео-конференций
WebBrowser	веб-браузера Network	
WebDevelopment	Инструмент для веб- Network или Development	разработчиков
Midi	Приложение, свя- AudioVideo; Audio	занное с MIDI
Mixer	Просто микшер AudioVideo; Audio	
Sequencer	Секвенсор AudioVideo; Audio	
Tuner	Тюнер AudioVideo; Audio	
TV	ТВ-приложение AudioVideo; Audio	
AudioVideoEditing	Приложение для ре- Audio или Video или AudioVideo	дактирования аудио / видео файлов
Player	Приложение для Audio или Video или AudioVideo	воспроизведения аудио / видео фай-лов
Recorder	Приложение для за- Audio или Video или AudioVideo	писи аудио / видео файлов
DiscBurning	Приложение для за- AudioVideo	писи дисков
ActionGame	Экшн-игра Game	
AdventureGame	Game	
Приключенческая игра		
АркадаGame	Аркада стиль игры Game	
BoardGame	Настольная игра Game	
BlocksGame	Игра падающие бло- Game	ки
CardGame	Карточная игра Game	
KidsGame	Игра для детей Game	
LogicGame	Логические игры, Game	такие как голово-ломки и т. Д.
RolePlaying	Ролевая игра Game	
Shooter	Шутер Game	
Simulation	Симулятор Game	
SportsGame	Спортивная игра Game	
StrategyGame	Стратегическая иг- Game	ра
Art	Программы для обу- Education или Science	чения искусству
Construction	Образование Education или Science	
Music	Музыкальное про- AudioVideo or Education	граммное обеспече-ние

Languages	Программное обеспечение для изучения иностранных языков	Education или Science
ArtificialIntelligence	Программное обеспечение искусственного интеллекта	Education или Science
Astronomy	Астрономия программного обеспечения	Education или Science
Biology	Программное обеспечение для биологии	Education или Science
Chemistry	Программное обеспечение Химия	Education или Science
ComputerScience	Программное обеспечение для изучения информатики	Education или Science
DataVisualization	Программное обеспечение для визуализации данных	Education или Science
Economy	Экономика программного обеспечения	Education или Science
Electricity	Программное обеспечение для электричества	Education или Science
Geography	Программное обеспечения по географии	Education или Science
Geology	Программное обеспечение геологии	Education или Science
Geoscience	Программное обеспечение наук о Земле, GIS	Education или Science
History	Программное обеспечения по истории	Education или Science
Humanities	Программное обеспечение для философии, психологии и других гуманитарных наук	Education или Science
ImageProcessing	Программное обеспечение для обработки изображений	Education или Science
Literature	Программное обеспечение по литературе	Education или Science
Maps	Программное обеспечение для карт для просмотра карт, навигации, картографии, GPS или	Education или Science или Education или Utility

Math	Математика Программное обеспечение	Education или Science
NumericAnalysis	Программное обеспечение для численного анализа	Education или Science или Наука или Math
MedicalSoftware	Медицинское программное обеспечение	Education или Science
Physics	Программное обеспечение по физике	Education или Science
Robotics	Программное обеспечение робототехники	Education или Science
Spirituality	Религиозное и духовное программное обеспечение	Education или Science или Utility
Sports	Спортивное ПО	Education или Science
ParallelComputing	Программное обеспечение для параллельных вычислений	ComputerScience или Science; ComputerScience
Amusement	Образование: Простое развлечение	
Archiving	Инструмент для архивирования / резервного копирования данных	Utility
Compression	Инструмент для управления сжатыми данными / архивами	Utility; Archiving
Electronics	Программное обеспечение для электроники, например схемотехники	
Emulator	Эмулятор другой платформы, такой как DOS эмулятор	System или Game
Engineering	Инженерное программное обеспечение, например CAD программы	
FileTools	Файловые утилиты	Utility or System
FileManager	Файловый менеджер	System; FileTools
TerminalEmulator	Приложения эмуляции терминала	System
Filesystem	Инструмент файловой системы	System

Monitor	Приложение / апплет, который контролирует неко- торый ресурс или активность	System или Network
Security	Инструмент без- опасности	Settings или System
Accessibility	Специальные воз- можности Доступ- ность	Settings или Utility
Calculator	Калькулятор	Utility
Clock	Приложение часов / апплет	Utility
TextEditor	Текстовый редактор	Utility
Documentation	Help или документа- ция	
Adult	Приложение для взрослых управ- ляет материалами для взрослых или явные заданными материалами	
Core	Основное приложе- ние, важное для яд- ра рабочего стола, такое как файловый менеджер или спра- вочный браузер	
KDE	Приложение KDE на основе библиотек KDE	QT
GNOME	Приложение GNOME на ос- нове библиотек GNOME	GTK
XFCE	Приложение XFCE на основе библиотек XFCE	GTK
GTK	Приложение GTK на основе библиотек GTK+	
Qt	Приложение на ос- нове библиотек Qt	
Motif	Приложение на основе библиотек Motif	
Java	Приложение, осно- ванное на библиоте- ках Java GUI, таких как AWT или Swing	

ConsoleOnly	Приложение , которое работает только внутри терминала (текстовое приложение или приложение командной строки)
-------------	--

8.1.3 Зарезервированные категории

Зарезервированные категории имеют специфичное для рабочего стола значение, которое не было стандартизировано (пока). Файлы записей рабочего стола, которые используют зарезервированную категорию, ДОЛЖНЫ также включать соответствующую запись *OnlyShowIn* =, чтобы ограничить себя теми средами, которые должным образом поддерживают зарезервированную категорию в том виде, в котором она используется.

В таблице ниже описаны зарезервированные категории.

8.2 Приложение Б. Окружения зарегистрированные OnlyShowIn

Включив одну из главных категорий в файл ярлыка приложения на рабочем столе, приложение будет отображаться в разделе меню приложения, посвященном этой категории. Если несколько основных категорий включены в один файл ярлыка рабочего стола, запись может появляться в меню более одного раза.

Категоризированные элементы меню, основанные на Основных категориях, перечисленных в этой спецификации, не обеспечивают полную онтологию для всех доступных приложений. Поэтому реализациям меню на основе категорий СЛЕДУЕТ предоставлять подменю «всеохватывающего» для приложений, которые нельзя разместить в других местах.

В таблице ниже перечислены все основные категории.

Запомните, имена чувствительны к регистру. "KDE"это не "kde".

8.3 Приложение В. Интеграция вашего приложения в меню

8.3.1 Добавление пунктов меню

Следующие шаги описывают, как стороннее приложение может добавлять пункты меню в систему меню:

- Задайте ярлыки рабочего стола в *datadir/apps/* для каждого пункта меню. Пожалуйста, укажите имя файла, например, "*vendor-foo.desktop*" или используйте подкаталог *datadir/Applications/*, чтобы у вас был "*vendor/foo.desktop*". Убедитесь, что все ярлыки на рабочем столе верны (см. Пакет *desktop-file-utils* для утилиты проверки).
- Установите файл меню XML в *sysconfdir /menus/apps-merged/*, чтобы добавить любые подменю, если записи рабочего стола еще не включены в некоторые общие категории.
- Установите любые записи каталога, необходимые для вашего подменю, в *datadir/desktop-directoryies/*, следя за пространством имен и проверяя записи каталога

Таблица 1: Основные категории

Основная категория	Описание	Примечания
AudioVideo	Приложение для представления, создания или обработки мультимедиа (аудио / видео)	
Audio	Аудио приложение	Запись на рабочем столе также должна включать AudioVideo
Video	Видео приложение	Запись на рабочем столе также должно содержать AudioVideo.
Development	Приложение для разработки	
Education	Образовательное программное обеспечение	
Game	Игры	
Graphics	Графическое приложение для просмотра, создания или обработки графики	
Network	Сетевое приложение, такое как веб-браузер	
Office	Приложение офисного типа	
Science	Научное программное обеспечение	
Settings	Настройки приложений	Элементы меню могут появляться в отдельном меню или как часть «Центра управления»
System	Системное приложение, «Системные инструменты», например, программа просмотра журнала или сетевой монитор.	
Utility	Небольшое служебное приложение "Аксессуары"	

Таблица 3: Зарезервированные категории

Зарезервированная категория	Описание
Screensaver	Заставка (запуск этой иконки на рабочем столе должен активировать заставку)
TrayIcon	Приложение, которое в первую очередь является значком для «системного трей» или «области уведомлений» (приложения, которые открывают обычное окно и даже имеют значок в трее, не должны перечислять эту категорию)
Applet	Апплет, который будет запускаться внутри панели или другого такого приложения, вероятно, для конкретного рабочего стола
Shell	Оболочка (фактическая специфическая оболочка, такая как bash или tcsh, а не TerminalEmulator)

Таблица 4: Окружения зарегистрированные OnlyShowIn

OnlyShowIn	Значение	Окружение
GNOME	GNOME	Desktop
GNOME-Classic	GNOME Classic	Desktop
GNOME-Flashback	GNOME	Flashback Desktop
KDE	KDE	Desktop
LXDE	LXDE	Desktop
LXQt	LXQt	Desktop
MATE	MATE	Desktop
Razor	Razor-qt	Desktop
ROX	ROX	Desktop
TDE	Trinity	Desktop
Unity	Unity	Shell
XFCE	XFCE	Desktop
EDE	EDE	Desktop
Cinnamon	Cinnamon	Desktop
Pantheon	Pantheon	Desktop
Old	Legacy menu systems	

8.3.2 Места установки

Если приложение предназначено для установки пользователем root в масштабе всей системы, то рекомендуется использовать `/usr/share` в качестве значения для `datadir`, а `/etc/xdg` - в качестве значения для `sysconfdir`. В случае, если иерархия `/usr/share` недоступна для записи, рекомендуется использовать `/usr/local/share` в качестве значения для `datadir`.

Если приложение предназначено для установки непривилегированным пользователем для исключительного использования только этим пользователем, то в качестве значения для `datadir` следует использовать `$XDG_DATA_HOME`, а в качестве значения для `sysconfdir` следует использовать `$XDG_CONFIG_HOME`. Если `$XDG_DATA_HOME` не установлен, для него следует использовать значение по умолчанию `$HOME/.local/share`. Если `$XDG_CONFIG_HOME` не установлен, для него следует использовать значение по умолчанию `$HOME/.config`.

8.3.3 Пример

Компания *ShinyThings Inc.* разработала приложение с именем *WebMirror 1.0* и хотела бы добавить свое собственное подменю в системные меню, состоящее из пункта меню *WebMirror* и пункта меню *WebMirror Admin Tool*. Компания будет использовать "shinythings" в качестве идентификатора своего поставщика. Для целей данного примера все пункты меню будут доступны на двух языках: английском и голландском. Код языка для голландского языка `nl`.

Сначала компании необходимо создать два файла `.desktop`, которые описывают два пункта меню:

```
datadir/applications/shinythings-webmirror.desktop
```

```
[Desktop Entry]
Encoding=UTF-8
Type=Application

Exec=webmirror
Icon=webmirror

Name=WebMirror
Name[nl]=WebSpiegel
```

И

```
datadir/applications/shinythings-webmirror-admin.desktop
```

```
[Desktop Entry]
Encoding=UTF-8
Type=Application

Exec=webmirror-admintool
Icon=webmirror-admintool

Name=WebMirror Admin Tool
Name[nl]=WebSpiegel Administratie Tool
```

Файл `.directory` должен быть установлен, чтобы предоставить заголовок и значок для самого под-меню:

```
datadir/desktop-directories/shinythings-webmirror.directory
```

```
[Desktop Entry]
Encoding=UTF-8

Icon=webmirror

Name=WebMirror
Name[nl]=WebSpiegel
```

И, наконец, необходимо предоставить файл `.menu`, который связывает все это вместе:

```

<!DOCTYPE Menu PUBLIC "-//freedesktop//DTD Menu 1.0//EN"
"http://www.freedesktop.org/standards/menu-spec/menu-1.0.dtd">
<Menu>
<Name>Applications</Name>
<Menu>
  <Name>WebMirror</Name>
  <Directory>shinythings-webmirror.directory</Directory>
  <Include>
    <Filename>shinythings-webmirror.desktop</Filename>
    <Filename>shinythings-webmirror-admin.desktop</Filename>
  </Include>
</Menu>

```

8.3.4 Обратная совместимость

В течение ограниченного времени установка иерархии каталогов в старых местах GNOME и KDE, таких как `/usr/share/applications` и `/usr/share/gnome/apps`, будет продолжать работать как способ добавления приложения в систему меню. Существует два способа поддержки как старой, так и новой систем меню одновременно:

- Если вы добавите строку «Категории» к записям рабочего стола в устаревшей иерархии, реализации этой спецификации будут игнорировать их расположение в устаревшей иерархии и вместо этого упорядочат их в соответствии с категориями. Это позволяет установить один файл рабочего стола, который работает во всех случаях, хотя, с другой стороны, он находится в устаревшем месте.
- Если вы добавите строку `OnlyShowIn = Old;` до записи рабочего стола, тогда старые унаследованные реализации, которые игнорируют `OnlyShowIn`, все равно будут отображать запись рабочего стола, но реализации этой спецификации не будут. Таким образом, вы можете добавить «`OnlyShowIn = Old;`» запись в унаследованную иерархию и запись рабочего стола в новом стиле в `datadir/Applications/`, и все же получить только одну запись в меню.

8.4 Приложение Г. Замечания по реализации

8.4.1 Редактирование меню

Для реализации редактирования меню предполагается, что создается файл для каждого пользователя. Файл для пользователя должен указывать `<MergeFile>` с файлом всей системы, чтобы системные изменения наследовались. Когда пользователь удаляет элемент меню, вы добавляете `<Exclude>` `<Filename> foo.desktop </Filename>` `</Exclude>`.

Если пользователь перемещает папку, вы можете использовать элементы `<Move>` для представления перемещения. Элементы `<Move>`, используемые для редактирования меню, всегда должны добавляться в меню самого верхнего уровня, чтобы гарантировать, что перемещения выполняются в порядке, в котором они указаны; ходы, указанные в дочерних меню, всегда выполняются перед ходами, указанными в меню более высокого уровня, независимо от их расположения в файле меню.

Чтобы удалить папку, просто добавьте элемент `<Deleted>`.

При добавлении новой папки или перемещении существующей папки реализациям редактирования меню рекомендуется не использовать путь меню ранее удаленной папки.

Редакторы меню, вероятно, должны выполнить некоторую консолидацию / сжатие, чтобы избежать бесконечного роста дерева XML.

9 Глоссарий

Этот глоссарий определяет некоторые термины, используемые в данной спецификации.

Desktop entry Ярлык рабочего стола - это файл с именем, оканчивающимся расширением «*.desktop*», которое соответствует спецификации ярлыка рабочего стола с *Type = Application*. Он описывает пункт меню, включая имя, значок и что делать, когда этот элемент выбран. Ярлыки рабочего стола также называются «*.desktop files*».

Desktop-File Id ID для идентификации ярлыка рабочего стола. Для файлов рабочего стола, установленных в *\$XDG_DATA_DIRS/apps*, это то же самое, что и идентификатор файла рабочего стола, определенный в спецификации записи рабочего стола. Кроме того, если */opt/ude* определен как *<LegacyDir prefix="foo ->*, то */opt/ude/Settings/bar.desktop* имеет идентификатор файла рабочего стола *foo-bar.desktop*

Directory entry *Directory entry* - это файл с именем, оканчивающимся расширением «*.directory*», которое соответствует спецификации ярлыка рабочего стола с *Type=Directory*. Он предоставляет локализованное имя и значок для подменю. Записи каталога также называются «*.directory files*».

Menu path «Menu path» - это путь к определенному меню. Пути к меню всегда «относительны», поэтому никогда не начинайте с косой черты. Путь к меню - это просто *<Имя>* каждого родителя меню, за которым следует *<Имя>* самого меню. Например, «*Foo/Bar/Baz*» является допустимым путем меню.

Relative path Канонический путь к записи каталога, относительно *<DirectoryDir>*, содержащего запись. Например, если каталог */usr/share/desktop-directory* указан как *<DirectoryDir>*, относительный путь к */usr/share/desktop-directory/foo/bar.directory* будет *foo/bar.directory*.