

Desktop Application Autostart Specification

By John Palmieri <johnp@redhat.com>

Archived: 2026-04-06 15:47:37 UTC

Authors:

Kévin Ottens <ervin@ipsquad.net>

Renatoaldas <seventhguardian_@hotmail.com>

Rodrigo Moya <rodrigo@gnome-db.org>

Waldo Bastian <bastian@kde.org>

Publication Date: 13 February 2006, Version: Version 0.5

1 Introduction

This DRAFT document defines a method for automatically starting applications during the startup of a desktop environment and after mounting a removable medium.

Some of the file locations in this specification are specified based on the "[desktop base directory specification](http://standards.freedesktop.org/basedir-spec/)" (<http://standards.freedesktop.org/basedir-spec/>).

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119.

2 Autostart Of Applications During Startup

By placing an application's `.desktop` file in one of the Autostart directories the application will be automatically launched during startup of the user's desktop environment after the user has logged in.

2.2 Application `.desktop` Files

An application `.desktop` file must have the format as defined in the "[Desktop Entry Specification](http://standards.freedesktop.org/desktop-entry-spec/)" (<http://standards.freedesktop.org/desktop-entry-spec/>). All keys should be interpreted as defined with the following exceptions in order to take into account that the `.desktop` files in an autostart directory are not shown in a menu.

2.2.1 Hidden Key

When the `.desktop` file has the `Hidden` key set to true, the `.desktop` file MUST be ignored. When multiple `.desktop` files with the same name exists in multiple directories then only the `Hidden` key in the most important `.desktop` file must be considered: If it is set to true all `.desktop` files with the same name in the other directories MUST be ignored as well.

2.2.2 `OnlyShowIn` and `NotShowIn` Keys

The `OnlyShowIn` entry may contain a list of strings identifying the desktop environments that MUST autostart this application, all other desktop environments MUST NOT autostart this application.

The `NotShowIn` entry may contain a list of strings identifying the desktop environments that MUST NOT autostart this application, all other desktop environments MUST autostart this application.

Only one of these keys, either `OnlyShowIn` or `NotShowIn`, may appear in a single `.desktop` file.

2.2.3 `TryExec` Key

A `.desktop` file with a non-empty `TryExec` field MUST NOT be autostarted if the value of the `TryExec` key does NOT match with an installed executable program. The value of the `TryExec` field may either be an absolute path or the name of an executable without any path components. If the name of an executable is specified without any path components then the `$PATH` environment is searched to find a matching executable program.

2.3 Implementation Notes

If an application autostarts by having a `.desktop` file installed in the system wide autostart directory, an individual user can disable the automatic start of this application by placing a `.desktop` file of the same name in their personal autostart directory which contains the key `Hidden=true`.

3 Autostart Of Applications After Mount

When a desktop environment mounts a new medium, the medium may contain an Autostart file that can suggest to start an application or an Autoopen file that can suggest to open a specific file located on the medium.

3.1 Autostart Files

When a new medium is mounted the root directory of the medium should be checked for the following Autostart files in order of precedence:

- `.autorun`
- `autorun`
- `autorun.sh`

Only the first file that is present should be considered.

The desktop environment MAY ignore Autostart files altogether based on policy set by the user, system administrator or vendor.

The desktop environment MUST prompt the user for confirmation before automatically starting an application.

When an Autostart file has been detected and the user has confirmed its execution the autostart file MUST be executed with the current working directory (CWD) set to the root directory of the medium.

3.2 Autoopen Files

When a new medium is mounted and a) the medium does not contain an Autostart file or b) a policy to ignore Autostart files is in effect then the root directory of the medium should be checked for the following Autoopen files in order of precedence:

- `.autoopen`
- `autoopen`

Only the first file that is present should be considered.

The desktop environment MAY ignore Autoopen files altogether based on policy set by the user, system administrator or vendor.

An Autoopen file MUST contain a single relative path that points to a non-executable file contained on the medium. If the file contains a newline or carriage return character then the newline or carriage return character itself and all characters that follow MUST be ignored.

The relative path MUST NOT contain path components that refer to a parent directory (`../`)

The relative path MUST NOT point to an executable file.

The desktop environment MUST verify that the relative path points to a file that is actually located on the medium, taking into account any symbolic or other links and MUST ignore any relative path that points to a file location outside the medium itself.

If the relative path points to an executable file then the desktop environment MUST NOT execute the file.

The desktop environment MUST prompt the user for confirmation before opening the file.

When an Autoopen file has been detected and the user has confirmed that the file indicated in the Autoopen file should be opened then the file indicated in the Autoopen file MUST be opened in the application normally preferred by the user for files of its kind UNLESS the user instructed otherwise.