



Dolby Voice Electron Framework patch for Microsoft Windows

Software Guide

22 July 2020

1 Introduction to the Dolby Voice Electron Framework patch software guide

This documentation describes how to implement the Dolby Voice Electron Framework patch, and how to verify the DVC-2 codec usage. It also contains the license that describes its terms of use.

2 About the Dolby Voice Electron Framework patch

The Dolby Voice Electron Framework patch offers an improved audio connection quality for the applications created with the Electron framework.

The Dolby Voice Electron Framework patch modifies the Electron to enable its integration with the Dolby Voice Client and usage of the DVC-2 codec in WebRTC conferences. This codec provides features such as noise and echo reduction, voice leveling, and spatial audio (for three-dimensional, realistic sound).

3 System requirements

Before implementing the Dolby Voice Electron Framework patch, ensure that your system complies with the minimum requirements.

The minimum requirements are listed in the following table.

Table 1: Dolby Voice Electron Framework patch requirements

Requirement	Version
Operating system	Windows 10 or later
Server version of the operating system	Windows Server 2012 R2 or later
Visual Studio	15.7.2 or later
Python	2.7.10 or later, with the support of Transport Layer Security (TLS) 1.2
Python for Windows extension (pywin32)	NA
Git	NA
Node.js	NA
Windows SDK	10.0.15063.468
Debugging tools for Windows	NA
Any text editor for building Electron	NA
Dolby Voice Client SDK	Check the version in the release notes
100 GB available disk space	NA

Additionally, set the following system variables:

- DEPOT_TOOLS_WIN_TOOLCHAIN=0
- GYP_MSVS_VERSION=2019

To set the variables, use this procedure:

1. Use the Windows key + R keyboard shortcut to open the run window.

2. In the run window, paste the following command.

```
rundll32 sysdm.cpl,EditEnvironmentVariables
```

3. For each variable, navigate to the **user environment variables** and select **New** option.
4. Fill in the form for each variable (for example, name: `DEPOT_TOOLS_WIN_TOOLCHAIN`, value: 0).
5. Select **OK** to confirm.

Additional information:

- We recommend installing Visual Studio in the default directory. Otherwise, set proper variables to point the tool chains to the installation path. You may need to change the following elements: 2019, Community, and `DRIVE:` in these paths:

```
vs2019_install = DRIVE:\path\to\Microsoft Visual Studio\2019\Community
```

```
WINDOWSSDKDIR = DRIVE:\path\to\Windows Kits\10
```

- To install the Windows SDK, follow these steps:
 1. Open Visual Studio Installer.
 2. Select **Modify** ▸ **Individual Components**.
 3. Select the proper Windows SDK to install.
- To install debugging tools for Windows, follow these steps:
 1. Go to: **Control panel** ▸ **Programs** ▸ **Programs and Features**.
 2. Select **Windows Software Development Kit**.
 3. Select **Change** ▸ **Check Debugging Tools For Windows** ▸ **Change**.

For more information about Electron prerequisites, see: <https://www.electronjs.org/docs/development/build-instructions-windows#prerequisites>

4 Implementing the Dolby Voice Electron Framework patch


Apply the Dolby Voice Electron Framework patch on Electron to integrate it with the Dolby Voice Client.

Prerequisites

Make sure that you meet the system requirements.

About this task

- Use Git Bash with administrator privileges to perform this procedure.
- Use a command line to build Electron. A compilation using Visual Studio is currently not supported.
- Be prepared that the entire building process takes several hours.

 **Note:** This procedure mentions the Electron 8.3.4 and Dolby Voice Client DVC_3.0.1_r5245647 as examples; refer to the release notes to find the proper version numbers.

Procedure

1. Open Git Bash program with administrator privileges.
2. Create an empty working directory, such as `C:\electron:`

```
mkdir -p /c/electron && cd /c/electron
```

3. Place the following items in the empty working directory you created:


- The Dolby Voice Client SDK extracted from the *DVC_3.0.1_r5245647.zip* file
- The *dvc_electron.patch* Dolby Voice Electron Framework patch file

4. Download Electron dependencies:

```
git clone https://chromium.googlesource.com/chromium/tools/depot_tools.git && export
PATH=`pwd`/depot_tools:$PATH
```

5. Download the Electron source code to a new, dedicated folder called *electron-gn*:

```
mkdir electron-gn && cd electron-gn && gclient config --name "src/electron" --
unmanaged https://github.com/electron/electron && gclient sync --with_branch_heads --
with_tags
```

 **Note:** Your working directory should now contain:

- The *dvc_electron.patch* Dolby Voice Electron Framework patch file
- The *DVC_3.0.1_r5245647* folder with the *client* directory inside it
- The *electron-gn* folder with the *src* folder inside it

6. Download Electron 8.3.4:

```
cd src/electron && git checkout v8.3.4 && gclient sync -f && gclient sync -D
```

7. Apply the Dolby Voice Electron Framework patch:

```
cd .. && patch -p1 < ../../dvc_electron.patch
```

8. Prepare the build environment:

```
export CHROMIUM_BUILDTOOLS_PATH=`pwd`/buildtools
export DVC_LIBRARY_PATH=../../../../../DVC_3.0.1_r5245647
```

9. Generate the Electron build configuration:

```
gn gen out/Release --args="import(\"//electron/build/args/release.gn\") dvmc_sdk_root=
\"$DVC_LIBRARY_PATH\""
```

10. Build Electron:

```
ninja -C out/Release electron
```

11. Copy the Dolby Voice Client library to the newly built Electron application:

```
cp $DVC_LIBRARY_PATH/client/lib/windows/win64/dvclient.dll out/Release
```

12. Run Electron with Dolby Voice Client:

```
./out/Release/electron.exe
```

Results

Electron is integrated with the Dolby Voice Client, and it is capable of using the DVC-2 codec.

5 Verifying DVC-2 usage

Check the usage of the DVC-2 codec in WebRTC conferences to confirm that it works correctly.

Prerequisites

Configure the Dolby Voice Electron Framework patch as described in the configuration procedure.


About this task

We recommend performing this procedure before using the product, to confirm that it works correctly. In case of any problems with DVC-2 support in created applications, use this procedure to verify that this codec is still supported in WebRTC conferences. Doing so makes it easier to find the root cause of the problem.

This verification requires checking the presence of the specific elements during the Session Description Protocol (SDP) negotiation.

Procedure

1. Open the Electron application.
2. Using the application, go to the [WebRTC](#) web page to test your connection.
3. Select: **Get media** ▶ **Create peer connection** ▶ **Create offer**.

 **Note:** This step creates a peer connection, which turns on the camera. Notice that after performing this step, connection details appear in the **Offer SDP** window visible on the left side of the page.

4. Check whether the content of the **Offer SDP** window includes the following entries:

```
a=rtpmap:96 DVC-2/8000
```

5. Additionally, in the content of the **Offer SDP** window, look for the list of the supported codecs, similar to this example:

```
m=audio 9 UDP/TLS/RTP/SAVPF 96 111 9 0 8 13 110 126
```

Check whether the 96 ID is mentioned at the beginning of the list.

Results

The presence of the mentioned elements signals that during this connection, the DVC-2 codec works properly. If there is a different result, contact Dolby.

6 License agreement

The Dolby Voice Electron Framework patch can be utilized under conditions described in the license agreement.

Copyright 2020 Dolby Laboratories and Dolby International AB.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Notices

Trademarks

Dolby and the double-D symbol are registered trademarks of Dolby Laboratories.

The following are trademarks of Dolby Laboratories:

Dialogue Intelligence™	Dolby Theatre®
Dolby®	Dolby Vision®
Dolby Advanced Audio™	Dolby Vision IQ™
Dolby Atmos®	Dolby Voice®
Dolby Audio™	Feel Every Dimension™
Dolby Cinema®	Feel Every Dimension in Dolby™
Dolby Digital Plus™	Feel Every Dimension in Dolby Atmos™
Dolby Digital Plus Advanced Audio™	MLP Lossless™
Dolby Digital Plus Home Theater™	Pro Logic®
Dolby Home Theater®	Surround EX™

All other trademarks remain the property of their respective owners.

Patents

THIS PRODUCT MAY BE PROTECTED BY PATENTS AND PENDING PATENT APPLICATIONS IN THE UNITED STATES AND ELSEWHERE. FOR MORE INFORMATION, INCLUDING A SPECIFIC LIST OF PATENTS PROTECTING THIS PRODUCT, PLEASE VISIT <http://www.dolby.com/patents>.

Confidential information

Confidential information for Dolby Laboratories Licensees only. Unauthorized use, sale, or duplication is prohibited.