



LibreOffice
The Document Foundation

Jiří Burant

Original state

The screenshot shows the 'Options - LibreOfficeDev - OpenCL' dialog box. On the left is a tree view with 'OpenCL' selected. The main area is divided into three sections: 'OpenCL version', 'OpenCL blacklist', and 'OpenCL whitelist'. The 'OpenCL version' section has two checked options. The 'OpenCL blacklist' table lists two Windows entries. The 'OpenCL whitelist' table lists one Linux entry. At the bottom are 'Help', 'OK', 'Cancel', and 'Reset' buttons.

Options - LibreOfficeDev - OpenCL

- LibreOfficeDev
 - User Data
 - General
 - Memory
 - View
 - Print
 - Paths
 - Colors
 - Fonts
 - Security
 - Personalization
 - Application Colors
 - Accessibility
 - Advanced
 - Basic IDE Options
 - OpenCL**
 - Load/Save
 - Language Settings
 - LibreOfficeDev Writer
 - LibreOfficeDev Writer/Web
 - LibreOfficeDev Base
 - Charts
 - Internet

OpenCL version

- Allow use of Software Interpreter (even when OpenCL is not available)
- Allow use of OpenCL

OpenCL blacklist

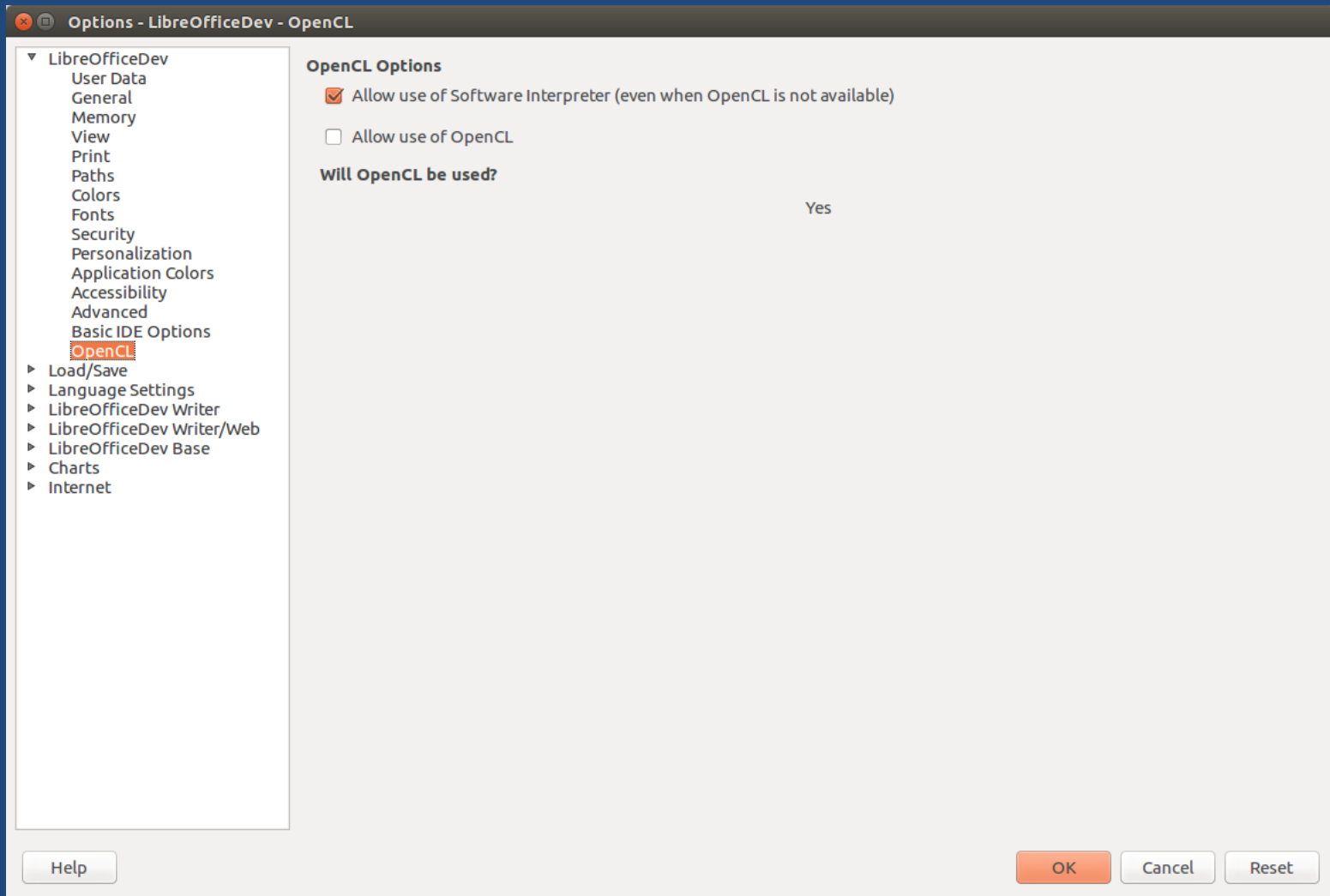
| Operating System | OS Version | Platfor... | Device | Driver version |
|------------------|------------|------------|--------|-------------------|
| Windows | | Intel\{ | | 9\.17\..10\..2884 |
| Windows | | Intel\{ | | 4\.2\..0\..99 |

OpenCL whitelist

| Operating System | OS Version | Platfor... | Device | Driver version |
|------------------|------------|------------|--------|-----------------------|
| | | | NVIDIA | |
| Linux | | Intel\{ | Advan | 1445\..5\{(sse2,avx\} |

Buttons: Help, OK, Cancel, Reset

My contribution



Technicalities of the solution

```
optopencl.cxx (~/libreoffice/cui/source/options) - gedit
Open Save Undo Cut Copy Paste Find
optopencl.cxx x
>SvxTabPage(parent, optopenCLPage, cui/ui/optopenclpage.ui, &rSet),
maConfig(OpenCLConfig::get())
{
    get(mpUseSwInterpreter, "useswinterpreter");
    get(mpUseOpenCL, "useopencl");
    get(cLUsed, "openclused");

    mpUseSwInterpreter->Check(officectg::Office::Common::Misc::UseSwInterpreter::get());

    mpUseOpenCL->Check(maConfig.mbUseOpenCL);
    mpUseOpenCL->SetClickHdl(LINK(this, SvxOpenCLTabPage, EnableOpenCLHdl));

    cL_device_id idTest=opencl::gpuEnv.mpDevID;
    if(!idTest)
    {
        cLUsed->SetText("Yes");
    }else{
        cLUsed->SetText("No");
    }
}

SvxOpenCLTabPage::~SvxOpenCLTabPage()
{
    disposeOnce();
}

void SvxOpenCLTabPage::dispose()
{
    mpUseSwInterpreter.clear();
    mpUseOpenCL.clear();

    SfxTabPage::dispose();
}

VclPtr<SfxTabPage> SvxOpenCLTabPage::Create( vcl::Window* pParent, const SfxItemSet* rAttrSet )
{
```

Cooperation

- At first very good
- Quick response
- Later, worse communication
- Ambiguous demands
- No clear specification

Evaluation

- The task has been completed
- Ambiguous reviews
- No clear output
- Not merged to new version

Thank You
for
Your Attention