Subject: Re: What improvements for Upp on Linux to comply with freedesktop?
Posted by fudadmin on Fri, 27 Jul 2007 15:59:47 GMT

View Forum Message <> Reply to Message

Well, first of all, I think, it's worse than a joke when Upp makefile does some kind of "installation" and copies files onto Linux systems leaving users with a manual tracing of installed files in case they want to uninstall...

Also, if a user tries compiling all U++ examples plus some of his own, it becomes easier to find a needle in a rubbish truck than to clean his home directory without damaging some of his useful configuration.

So, some of my proposals (regarding clean order on Linux) are:
1. First and the most important - keep all (to one Upp version related) files under ONE Upp configs directory with subdirectories for each app and NOT scattered all over users HOMEDIR!
2. According to some freedesktop recommendations, e.g:
==> /home/username/.config/upp/appname/*
and/or, because some of us have several versions of upp:
==>/home/username/.config/upp_version_name/appname/*
(open /home/username/.config/ to check which open source projects have already switched)

3. This is achieved easily with a change I've for my version (Core/App.cpp line ~123):

```
String  ConfigFile(const char *file) {
#if defined(PLATFORM_WIN32)
	return GetExeDirFile(file);
#elif defined(PLATFORM_POSIX)
	String p = GetHomeDirFile(".config/upp/" + GetExeTitle());
```

Ok, that is not ideal. The subdirs should be programmable, too. At least, upp_version_name. In fact, I really want that Upp users had an easy possibility to have their own versions to experiment and easy switching but that can wait a few days...

4. Similar functions to Zardos:
 http://www.ultimatepp.org/forum/index.php?t=msg&&th= 2511&goto=10729#msg_10729
should be discussed in the context of above.
I also have:
(they work for me on my Linux but must be corrected...)

```
String LoadAppConfigDirFile(const char *filename);
String LoadUPPConfigDirFile(const char *filename);

String GetAppConfigDirFile(const char *filename);
String GetUPPConfigDirFile(const char *filename);

bool SaveAppConfigDirFile(const char *filename, const String& data);
bool SaveUPPConfigDirFile(const char *filename, const String& data);
//====
String LoadAppConfigDirFile(const char *filename)
{
 return LoadFile(GetAppConfigDirFile(filename));
}

String LoadUPPConfigDirFile(const char *filename)
{
 return LoadFile(GetUPPConfigDirFile(filename));
}

String GetAppConfigDirFile(const char *filename)
{
 return Environment().Get("UPP_ARIS_CONFIG")+"/"+ GetExeTitle()+"/"+filename;
}

String GetUPPConfigDirFile(const char *filename)
{
 return Environment().Get("UPP_ARIS_CONFIG")+"/"+filename;
}

bool SaveAppConfigDirFile(const char *filename, const String& data)
{
 return SaveFile(GetAppConfigDirFile(filename), data);
}

bool SaveUPPConfigDirFile(const char *filename, const String& data)
{
 return SaveFile(GetUPPConfigDirFile(filename), data);
}
```

5. Maybe a separate logs subdir in each Upp app subdir?
6. Maybe more programmable configs flexibility in general?
7. A separate Utils-Configs-Settings package to keep with ever changing freedesktop and distros spirit?

I'm not saying that everything must be accepted. (Personally, I can keep them in my own libs and be happy.
But if many Upp users waste time on the same mundane things then maybe it's better to change something globally?

Just some thoughts. Perhaps, others opinions would be more help.