Subject: Re: Final release
Posted by mdelfede on Sun, 27 Jul 2008 12:56:35 GMT
View Forum Message <> Reply to Message

luzr wrote on Sun, 27 July 2008 09:01mdelfede wrote on Sat, 26 July 2008 15:07
Just think at 'path' problem and you'll see that's not manageable, OR you loose the ability to run apps from command line.

Why? If systems knows to search for binaries in Apps, it is only a little bit more complicated than PATH (which would be maintained only to express eventual priority).

It depends on how you do manage it.
If you have
/Apps/myapp1
/Apps/myapp2
.....
/Apps/myapp2345

you have OR a kilometric path OR your OS must search recursively through the full Apps folder.
Both of them are slow.

Quote:

Quote:
Second problem... where do you put config files ? in app folder... impossible, unix doesn't have a *reliable* way to locate executable location.

Actually, current solution in Core/App.cpp seems to work quite well...

I must admit that I didn't look at it I'll do sometimes.
But believe me, that's a known problem with no 100% reliable solution.

Anyways, I don't like the windows way of putting config files (sometimes) on app executable folder. That one should be write protected and not accessible by normal app usage. Like it is now it's an opened door for malware and/or coding mistakes.
In app folder should go only fixed config files, I mean files with data not modifiable by program itself. The rest should go on registry or, as in linux, on user owned folders.

Quote:
Anyway, generally, I could imagine a better way how to handle these issues, but I have already got used to current model and .debs and I think that they work acceptably well. Arguing whether "home" should be called "users" and "usr" -> "apps" is wasting of time

I agree
Btw, those names are there for historical reasons (don't ask me which, I knew some of them in past but I forgot !!!)

Just another small OT : the same belongs to LISP language CAR/CADR statements (getting first elements of a list/list with first element removed). IIRC these names came from a PDP11 machine instruction code...

Max
