

Donna Poreda

From: Brad Silverberg
To: davidcol
Subject: RE: non-synergy areas
Date: Thursday, June 03, 1993 8:51 AM

yes, apps synergy is our focus but we know cairo synergy will come up so we should be prepared.

From: davidcol
To: bradsi
Subject: RE: non-synergy areas
Date: Thursday, June 03, 1993 8:05 AM

I hate to look like a complete butt head here, but I think our focus should be on synergy with the apps group. They are the ones who will help make Chicago successful. They share goals with Chicago, we are in effect on the same mission.

I'm not saying we shouldn't work with the Cairo guys and fix some things. If we and the apps group align, then cairo will too by default. The last time we tried to line up with Cairo is cost us 5 months.

From: bradsi
To: davidcol; dennisad; johnlu
Subject: non-synergy areas
Date: Thursday, June 03, 1993 7:39 AM

we talked yesterday just about chicago:apps non-synergy. we should also address chicago:cairo non synergy. stevem has given us a good list of ui differences...

From: stevem Thu Jun 3 06:07:44 1993
To: bobmu bradsi chrisgr davidcol janineh jimall joeb markmal tandyt timsat
Subject: chicago/cairo diffs
Date: Thu Jun 3 06:07:16 1993

this is a list compiled off the top of several peoples' heads on where there are programming model differences between cairo and chicago. in many cases we will be visually compatible partially or totally, in other cases functions don't exist in chicago by current plan. most of this arises from cairo's decision to go whole-hog with ole and use it to define new function in the shell and UI in general, and chicago's decision not to.

please don't take this as bible, there is not a lot of detail behind the list at this point, and quite likely some errors due to my misunderstanding of chicago plans. it's meant as a starting point for discussion on how we spin this situation to isvs and ms apps - and where we might sync stuff up.

thanks
?
steve

(also, some items overlap/are redundant)

Drag/drop and transfer in shell, inter-shell-app
scrap support
scraps as embedded objects in shell objects
scrap drag and drop
links implementation, interop
app-specific shell handlers
common dialogs
rich text control
controls architecture (tho cairo picks up chic work)
projects
next/prev support
wastebasket interface/api
mechanisms for adding prop sheets
twins ? this one is in the air right now
twins reconcilers
?
explorer extensibility
link and view mgmt api for custom/app implemented folders
OLE2"c" - steve's paranoia placeholder for chicago work on ole
help - info pointers, authoring, title structure
icon registration
typeinfo/typelib usage
longfilename nit differences (e.g. maxlen)
EZ shell work - tho likely cairo will pick up
ctxt menu registration, extensibility mechanisms
shell ini files for config state, desktop state, folder view state
16bit components/apis and 32b app usage
which is faster - 16b or 32b app (i.e. what will perf-sensitive
isvs want to write)
ini file based storage model for links
providing storage
mail integration, mail add-ons
address book integration
"Find" (file) interfaces, UI integration
"Search" (text) integration (commdlg issue)
templates (?)

again, there is redundancy here, and we ain't done with the
?
app prototype with chrisgr, so I expect this list to be trimmed
and to then grow.