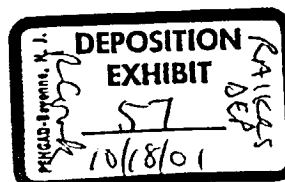From jeffr Tue Jan 29 08:03:33 1991
To: debem
Subject: Please print
Date: Tue Jan 29 08:00:43 1991
<EndOfHeader>
For Go file. Thx.

_____

>From mikemap Mon Jan 28 12:36:12 1991
To: gregs jeffr lloydfr
Subject: FYI - please fwd to Pen extensions folk
Date: Mon Jan 28 12:34:01 1991

> From paulma Mon Jan 28 08:16:57 1991
To: mikemap
Subject: FYI - please fwd to Pen extensions folk
Cc: billg bobmu darrylr nathanm steveb
Date: Mon Jan 28 08:14:35 1991

Last Friday, I was doing duty "voting" to incorporate Unicode as a non-profit
entity. At the meeting were folks from GO (Robert Carr and one other) who
tooks notes all meeting on their GO notebook computers. Afterwards Carr gave
me a demo:

-   Go will be selling a notebook to software and corporate developers
    only as a seeding exercise, thereafter they are purely a s/w company.
    Carr said that IBM, NCR, Grid will be releasing machines with GO
    software.

-   The machine he showed me looked slick and was light to carry. It had
    a 16MHz 286, 1MB Flash, 3Mb RAM.

-   He characterized the s/w as follows:
    -   fully 32bit enabled (all API's are 32bit), pre-emptive
        m/ting OS
    -   OO in the sense of defining how apps interface with each
        other and the system - though they do not pre-req C++ or
        such.
    -   System is organized as a set of "pages" where each page is
        instantiated as a separate process as it is referred to.
    -   When you start the system up, you see a "Notebook" index
        showing you which pages belong to which documents. Pages can
        be embedded in other pages (he showed me their "embedding"
        equivalent working). You navigate thru the document space by
        referring to pages.
    -   The user never sees "programs" per se - just documents of
        different types. It was not clear who one installs a new
        document type - but the systems comes with a couple of
        standard types - memo, drawing, spreadsheet, mail, etc.
    -   The file system is hidden behind these abstractions, but it
        will be accessible for those apps that want to - they will
        have a FAT compatible file system option.
    -   He made a big deal of the fact they provide "constraint based
        layout" for dialogues etc - which he claims will allow them

to deal with lot of different aspect ratio's, resolutions, etc. They (righly) believe that it the future, shirt pocket devices will be big market and they want apps to "scale" across lots of different sizes of displays.

- The system seemed a tad bit slow, but certainly acceptable. It did crash once during the demo, but otherwise seemed functional. The drawing package is still rudimentary, but works well with the pen. The handwriting recognition seemed to work OK (although he was doing all the writing).

My impression was that if h/w technology were such today that one could run their system on sub $500 shirt-pocket devices, then they could clean up. They have a nice, complete "works type" solution for pocket-reminders. However given that they cannot do that now, their venture is more questionable. It does however again demonstrate that if we do not provide solutions, we potentially create footholds for others.