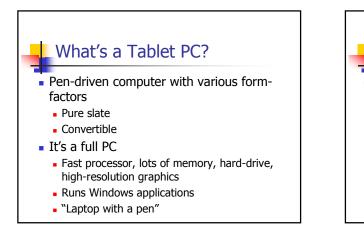
The Tablet PC: Designing Penbased Applications

Richard Anderson

Slides derived from Rob Jarrett Microsoft Corporation robjarr@microsoft.com

Topics

- Tablet PC introduction
 - What and why
- Usability issues
 - Hardware requirements
 - Display
 - Pen as mouse
 - Digital ink entry, editing, gestures, and recognition



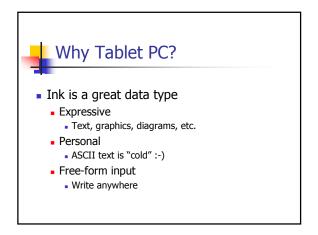
Why Tablet PC?

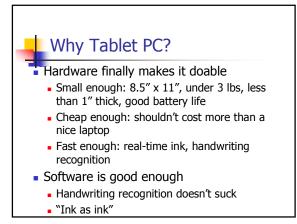
- Desktop and laptop computing doesn't fit well with certain common situations
 - Meetings: laptops can be annoying and/or remove user from discussion
 - Relaxing (e.g. leaning back in chair, lying on couch): laptops must be on a flat surface
 - On-the-go (walking, standing waiting): laptops must remain statically positioned for use

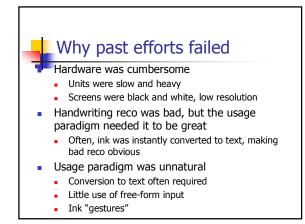
Why Tablet PC? Desktop and laptop computing can feel unnatural Entering information can be awkward (e.g. ASCII text into a word-processor) Using the mouse – hard to master, and disconnected from the task at hand Must be sitting upright when operating a computer

Why Tablet PC? Using a pen is natural Task adapted to human physiology rather than the other way around Evolved over thousands of years

- Easy
 - Most people can do it
 - Direct interaction/input
 - "Always on"







Why past efforts failed

- No standard software development platform
 - Low market penetration, tough business justification for 3rd parties
 - Therefore no "killer apps"
- Portable computers were niche
 - Networking wasn't common for PCs
 - Tradeoffs were significant in comparison to today (e.g. screens, upgradability, speed)

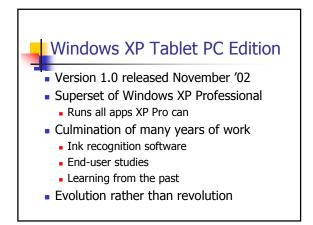
What about palm-sized devices?

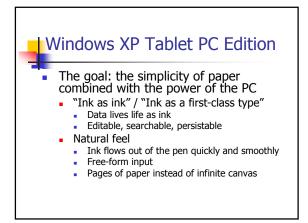
Great at what they do

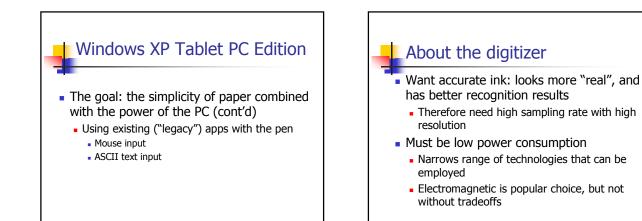
- Contacts, scheduling, jotting down short notes, etc.
- Saving grace (IMO): synchronization with desktop PCs
- Form factor is awesome
 - Fits in your pocket or hand!

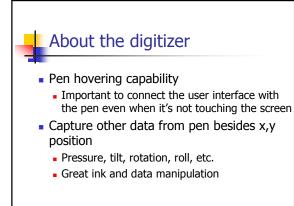
What about palm-sized devices?

- We still have to adapt to them
 - Small screens
 - "Graffiti" input
- Not good at desktop tasks
 - Unsatisfying for web surfing, email composition, document viewing, etc.
 - Syncing is a pain for some
 - Peripheral device for many users









About the digitizer

- Tradeoffs: sensitive to interference
 - Hard drive, CPU, battery, and other components can alter where the digitizer thinks the pen actually is
 - Calibration system (i.e. software correction) helps tremendously, but still not perfect

Display hardware

- User can't write directly on the LCD surface
 - Psychedelic color blooming occurs because of squishing liquid crystals; very distracting
 Bad for the display
- Solution: glass overlay

High DPI displays

Ink and eBooks look great

Tradeoff: Legacy apps suffer

Targeting areas get smaller

icons hard to see

increased DPI

easier

Displays are small but high resolution –

Many hard-coded pixel sizes, meaning text,

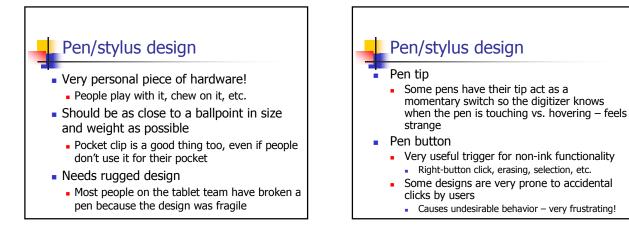
 Can only expect evolution here; platform makes this a pain today, tomorrow will be

- Doesn't allow any "give" across the display
- Protects the LCD

Display hardware

Tradeoffs:

- No "give" means it feels unnatural
- Glass is slippery to a hard plastic pen
 Pen skids a little, making writing and targeting a bit more difficult
- Parallax
 - Thickness of glass causes visual disconnect from ink/cursor/etc. when pen tip touches the display
- These will get better



Portrait-mode display

Support portrait mode; just like paper

- Great for web surfing, reading eBooks and most other document types
- Hot-switch to landscape and back is great for convertibles
- Tradeoff: Legacy apps suffer
 - All written assuming horizontal > vertical resolution
 - Toolbars, menus, etc. can be cut off

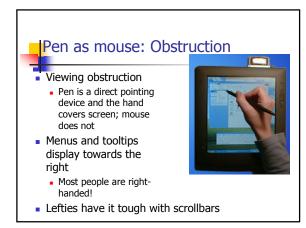
4

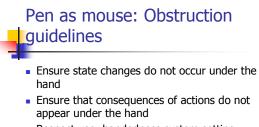
Pen as mouse: Hovering

- Hovering still with a pen is tough
 - While in-air, our control of a pen is considerably worse than when the pen is touching a surface
 - Mouse is intrinsically static, pen is not
 - Legacy applications typically assume cursor must be perfectly still for e.g. tooltips to appear
 - Software help needed to "smooth" hover location of cursor

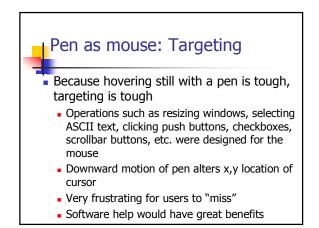
Pen as mouse: Hovering guidelines

- Features requiring hover should have generous tolerances
 - For instance, tooltips
 - Use COMCTL32 provided ToolBars
- Test hover-triggered features for ease of use





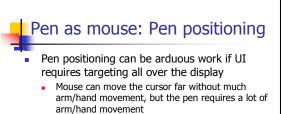
- Respect user handedness system setting
 Check SPI_GETMENUDROPALIGNMENT via SystemParametersInfo()
 - Apply it to Tooltips, menus, popup menus



Pen as mouse: Clicking

Clicking with a pen is tough

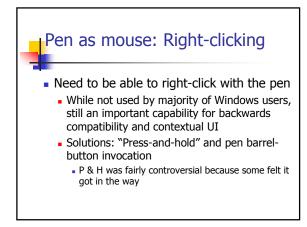
- Legacy applications typically assume during a click the mouse doesn't move
- Pen taps are more like little strokes or stabs because of pen skidding and high-precision digitizers
- Detecting the difference between tap and a drag is an interesting problem!
- Double-clicking is even tougher
 - Quick motion means sloppier result



- Menus and toolbars are typically at the top of a window; editing often occurs mid-way or toward the bottom
 - Lots of physical arm/hand movement results a real pain for users
 - More local UI is desirable (e.g. context menus)

Pen as mouse: Targeting guidelines

- Cursor feedback
- Bigger, easily-targeted controls
- Generous tap, double-click, and hover tolerances
- Keep related objects in proximity





Digital ink performance

Writing requires uninterrupted inking

- Users have difficulty with delays in ink appearance
- Users are frustrated with delays in inking
- Guideline
 - Ensure fast efficacy
 - Is it as fast as writing on paper?

Pen modes & cursors

- Explicit vs. implicit input modes
 A.k.a. *Modal* vs. *modeless*
- How to allow things to be efficient but not confuse users
 - Select mode uses "right-mouse button" for implicit mode as well as utilizing an explicit mode
 - Erase mode uses pen's eraser tip (if available) for implicit mode as well as utilizing an explicit mode

Pen modes & cursors

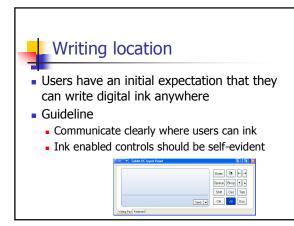
- Need feedback as to the mode of the pen
 - Indicates actions available to the user

Guideline

- Develop a set of cursor feedback to indicate the different modes of the pen
- Careful attention to cursor design
 - Either symmetric or use handedness setting

Pen gestures Gestures need precise tuning

- Trade-off between accidental activation vs. not getting when wanted
- False activations are annoying and distracting to the task!
- Guideline
 - Use gestures guardedly
 - Error on the side of having "zero" incidence of false activation
 - Non-destructive consequences are better



Ink selection

- Traditional rectangular selection tools are inadequate
- "Lasso" selection is much more natural
 - Percentage-based stroke tolerances
 - Employ word-based selection
 - Visual feedback is essential, real-time is much better than static

Recognition expectations

- Handwriting recognition is highly variable by person
 - Errors are expected
 - Perceived good or bad handwriting effects expectation of accuracy
- Guideline
 - Be realistic about recognition accuracy rate, don't rely heavily on it for authoring