Subject: Re: A terminal emulator widget for U++
Posted by Oblivion on Sun, 08 Dec 2024 09:23:45 GMT
View Forum Message <> Reply to Message

[code]
Hi,

The day has arrived, and TerminalCtrl 2025.1 is released.

Release Notes:

Requirements
- U++ >= 2024.1
- C++17

Customizability
- Custom Word/Cell Selection: Supports client-defined word selection via a filter. Selections can be based on text, SGR attributes, data types, or colors.
- Selector Mode: Introduced "Selector Mode" for keyboard navigation and text selection, supporting text, rectangle, line, and word selections.
- Configurable keyboard Shortcuts: All shortcuts are made configurable and can be redefined in client code.
- Text Search: TerminalCtrl now have support for text search, including range-based and parallel (`CoFind`) search
- Annotations: Users and hosted apps can now add annotations to terminal text (plain text or rich text via a simple protocol or menu).
- Custom Highlighting: Custom *cell-level* highlighting support.
- Application Mouse Tracking Override: Added the ability to override application mouse tracking with customizable modifier keys (Ctrl, Shift, Alt, and their combinations).

Protocols
- Directory Change Protocol: Implements the `OSC 7` and `OSC 9 ; 9`protocols to notify the terminal of the shell's working directory.
- Message Notification Protocol: Implements the `OSC 9 ; 2` message notification support.
- Progress Notification Protocol: Implements the `OSC 9 ; 4` protocol, used by modern terminal emulators, this protocol allows tracking of progress from hosted applications.
- iTerm2 Background Image Protocol: Implements iTerm2's background image change protocol.

Appearance
- Hyperlink Underline: Hyperlinks now switch to an under-dot pattern when using the underline attribute.
- Size Hint: The size hint now has rounded edges for better aesthetics.

Bug Fixes
- Mouse Reporting: Fixed mouse reporting bug.
- PC-Style Keys: Fixed the handling of HOME/END keys in PC-style mode.
- Stray Unicode Character: Fixed the Alt+Win key combination issue that produced stray Unicode characters.
- Background Transparency: Addressed transparency and background rendering issues.
- Selection on Resize: Selections are now cleared properly when resizing the terminal.

Performance Improvements
- Renderer Optimization: Refactored the renderer to combine non-contiguous characters, resulting in better performance

---

Behavior Changes
- Modifier Keys: `AltGr` and `Super` modifier keys no longer scroll the page.
- Page Scroll on Paste: The page now automatically scrolls to the bottom when pasting.
- Terminal Resize Behavior: Terminal lines are no longer truncated when resizing to a smaller window.

Code Maintenance
- Code Cleanup: Removed unused variables, redundant code.
- APtyProcess Base Class:
  - All PtyProcess classes are now derived from APtyProcess.
  - On Linux, the active class is `PosixPtyProcess`.
  - On Windows, it can use either `WinPtyProcess` or `ConPtyProcess` based on system configuration.
  - This makes it possible to dynamically manage different PtyProcess types in a single container.

If you wonder what can be achieved with TerminalCtrl, just check Bobcat (which is to be released officially before the end of the year.).

Best regards,
Oblivion