TransSkel 3.09 — Release Notes

This document is an adjunct to the TransSkel Programmer's Manual. It lists changes and additions specific to release 3.09. If you have only programmed with pre-3.0 releases of TransSkel, be sure to read the earlier 3.XX release notes as well.

- **Added new core routines `SkelGetWindContentRect()` and `SkelGetStructureRect()`, for returning a window's content and structure rectangles. Rewrote `SkelGetWindTitleHeight()` in terms of them.**

- **`SkelGetWindTitleHeight()` could goof up if the window wasn't current port. It doesn't now.**

- **Fixed conditional-testing botch in `SkelGetRectDevice()`.**

- Incompatible change. **`SkelGetWindowDevice()` and `SkelGetRectDevice()` were highly redundant, but separate, due to different requirements of what they were being called for: `SkelGetWindowDevice()` also returned the rectangle describing the largest usable area on the device, `SkelGetRectDevice()` didn't. `SkelGetWindowDevice()` cared about subtracting the menu bar area from the device area, `SkelGetRectDevice()` didn't.**

  In order to minimize the redundancy, I rewrote `SkelGetRectDevice()` to make it more general. It now takes two more parameters in which it returns the device rectangle and a flag indicating whether the device is the main device. `SkelGetWindowDevice()` becomes simply a special case of searching for a device rectangle, so I rewrote it in terms of `SkelGetRectDevice()`.

  These modifications change the status of `SkelGetRectDevice()`, which was formerly a convenience routine. Since it's now called by the core routine `SkelGetWindowDevice()`, `SkelGetRectDevice()` must also be a core routine.

  TransSkel's internal calls to `SkelGetRectDevice()` have been updated. If you have applications that call this function, those calls must be changed, too. This can be done relatively easily. If you call the function like this:

      result = SkelGetRectDevice (&r, &gd);

  Then change the call to look like this:

      result = SkelGetRectDevice (&r, &gd, (Rect *) nil, (Boolean *) nil);

- **Modified `SkelRouteEvent()` so that when the user drags a window, the window's `userState` rectangle is offset by the amount that the window is dragged. Otherwise you can't save window positions by reading the `userState` because they'll be inaccurate after the drag.**

- **New convenience routine `SkelPause()` for introducing momentary pauses into process execution.**

- **Removed the tutorial section from reference manual and put it in its own folder in the Documents folder. This is in anticipation of revising the tutorial so it's actually useful, which isn't now.**

- **Added new `SkelQuery()` selector `skelQGrayRgn`, which returns a copy of the desktop**

region. It differs from `GetGrayRgn()` in that it works whether or not `GetGrayRgn()` is implemented.