

TransSkel

Programmer's Notes

3: TransSkel Event Masks

Who to blame: Paul DuBois, dubois@primate.wisc.edu

Note creation date: 02/16/91

Note revision: 1.03

Last revision date: 01/03/94

TransSkel release: 3.00

This Note discusses the meaning and use of the TransSkel event masks, and how that contrasts with the system event mask.

The system event mask determines which events can be returned from `GetNextEvent()` or `WaitNextEvent()`. This is different than the event mask passed to either of those functions, which determines which events the application wishes to be informed of, and is a subset of the system event mask.

The TransSkel event mask corresponds to the mask passed to the event-getting functions: it determines which, from among the set of events that the system will retrieve, the application is interested in receiving. The dialog event mask is similar, but applies to modeless dialogs only.

Normally an application only sets the TransSkel event mask and doesn't do anything with the system event mask. One case which is an exception is when the application is interested in key-up events, which are normally discarded by the system. An application can request that they be returned by setting the system event mask. Releases of TransSkel prior to 3.00 provided no way for an application to receive key up events even if the system event mask had been modified. This is no longer true, but the application must explicitly request them both from the system and from TransSkel.

- The system event mask must be modified, e.g., by calling:

```
SetEventMask(everyEvent);
```

- The TransSkel event mask must be modified:

```
SkelSetEventMask (SkelGetEventMask () | keyUpMask);
```

- If modeless dialogs are used, and key up events should be passed to them as well, modify the dialog event mask:

```
SkelSetDlogEventMask (SkelGetDlogEventMask () | keyUpMask);
```

Key up events are signified by the high bit (bit 7) being set in the character code passed to a window handler's key callback function.

Remember that if your application modifies the system event mask, it must restore the mask to its normal state (`everyEvent - keyUpMask`) before exiting in order to avoid breaking other programs. See TN 202. (and PGMF?)