Due to testing an event union for SDL_BUTTON_LEFT without checking whether it was a mouse or keyboard event in the first place.

SDL client has been making the same mistake forever, so the shortcuts only ever worked by luck.

**History**

**#1 - 2020-01-01 06:14 PM - Jacob Nevins**
- Related to Bug #854477: SDL2: non-left mouse buttons (and maybe finger events?) don't work in various places added

**#2 - 2020-01-01 06:42 PM - Jacob Nevins**
- File m-30-26-sdl2-key-shortcuts.patch added
- Status changed from In Progress to Resolved

This builds on zeko's patch in #831742 by changing PRESSED_EVENT() and using it in the remaining places where that patch used SDL_FINGERDOWN directly.

**#3 - 2020-01-01 06:57 PM - Jacob Nevins**

SDL client has been making the same mistake forever, so the shortcuts only ever worked by luck.

In detail:

In `/usr/include/SDL` on my system, we have:

```c
typedef union SDL_Event {
    Uint8 type;
    SDL_KeyboardEvent key;
    SDL_MouseButtonEvent button;
    /* etc */
} SDL_Event;
```

where the two union members are

```c
typedef struct SDL_KeyboardEvent {
    Uint8 type;   /* < SDL_KEYDOWN (2) or SDL_KEYUP (3) */
    Uint8 which;  /* The keyboard device index */
    Uint8 state;  /* SDL_PRESSED (1) or SDL_RELEASED (0) */
    SDL_keysym keysym;
} SDL_KeyboardEvent;
```

```c
typedef struct SDL_MouseButtonEvent {
    Uint8 type;   /* < SDL_MOUSEBUTTONDOWN or SDL_MOUSEBUTTONUP */
    Uint8 which;  /* The mouse device index */
    Uint8 button; /* The mouse button index */
    Uint8 state;  /* < SDL_PRESSED or SDL_RELEASED */
    Uint16 x, y;  /* < The X/Y coordinates of the mouse at press time */
} SDL_MouseButtonEvent;
```
It so happens that any SDL_KEYDOWN event (state = SDL_PRESSED) will look as though button = SDL_BUTTON_LEFT if the wrong union member is used, probably on all platforms (assuming these structures have been the same shape for a while).

So, any test of "Main.event.button.button == SDL_BUTTON_LEFT" will happen to pass for any SDL_KEYDOWN event. The SDL client has relied on this.

In contrast, in /usr/include(SDL2 on my system:

```c
typedef union SDL_Event {
    Uint32 type; /**< Event type, shared with all events */
    SDL_KeyboardEvent key; /**< Keyboard event data */
    SDL_MouseButtonEvent button; /**< Mouse button event data */
    /* etc */
} SDL_Event;

typedef struct SDL_MouseButtonEvent {
    Uint32 type; /**< ::SDL_MOUSEBUTTONDOWN or ::SDL_MOUSEBUTTONUP */
    Uint32 timestamp; /**< In milliseconds, populated using SDL_GetTicks() */
    Uint32 windowID; /**< The window with mouse focus, if any */
    Uint32 which; /**< The mouse instance id, or SDL_TOUCH_MOUSEID */
    Uint8 button; /**< The mouse button index */
    /* etc */
} SDL_MouseButtonEvent;

typedef struct SDL_KeyboardEvent {
    Uint32 type; /**< ::SDL_KEYDOWN or ::SDL_KEYUP */
    Uint32 timestamp; /**< In milliseconds, populated using SDL_GetTicks() */
    Uint32 windowID; /**< The window with keyboard focus, if any */
    Uint8 state; /**< ::SDL_PRESSED or ::SDL_RELEASED */
    Uint8 repeat; /**< Non-zero if this is a key repeat */
    Uint8 padding2;
    Uint8 padding3;
    SDL_Keysym keysym; /**< The key that was pressed or released */
} SDL_KeyboardEvent;
```

Here, what SDL_MouseButtonEvent::button maps to in a SDL_KeyboardEvent is more complex and probably platform-dependent. This is probably why we didn't get away with it in SDL2.

To fix this latent bug for S2_6 SDL client probably requires backporting zeko's #831742 plus my subsequent fixes to SDL-client. I'm not going to do that here and now; it probably wants its own ticket, if anyone's going to do it.

#4 - 2020-01-03 11:01 AM - Jacob Nevins
- Status changed from Resolved to Closed

Files

m-30-26-sdl2-key-shortcuts.patch 8.3 KB 2020-01-01 Jacob Nevins