ALSA Jack Controls

Why we need Jack kcontrols

ALSA uses kcontrols to export audio controls(switch, volume, Mux, ...) to user space. This means userspace applications like pulseaudio can switch off headphones and switch on speakers when no headphones are pluged in.

The old ALSA jack code only created input devices for each registered jack. These jack input devices are not readable by userspace devices that run as non root.

The new jack code creates embedded jack kcontrols for each jack that can be read by any process.

This can be combined with UCM to allow userspace to route audio more intelligently based on jack insertion or removal events.

Jack Kcontrol Internals

Each jack will have a kcontrol list, so that we can create a kcontrol and attach it to the jack, at jack creation stage. We can also add a kcontrol to an existing jack, at anytime when required.

Those kcontrols will be freed automatically when the Jack is freed.

How to use jack kcontrols

In order to keep compatibility, snd_jack_new() has been modified by adding two params:

initial_kctl
: if true, create a kcontrol and add it to the jack list.

phantom_jack
: Don't create a input device for phantom jacks.

HDA jacks can set phantom_jack to true in order to create a phantom jack and set initial_kctl to true to create an initial kcontrol with the correct id.

ASoC jacks should set initial_kctl as false. The pin name will be assigned as the jack kcontrol name.