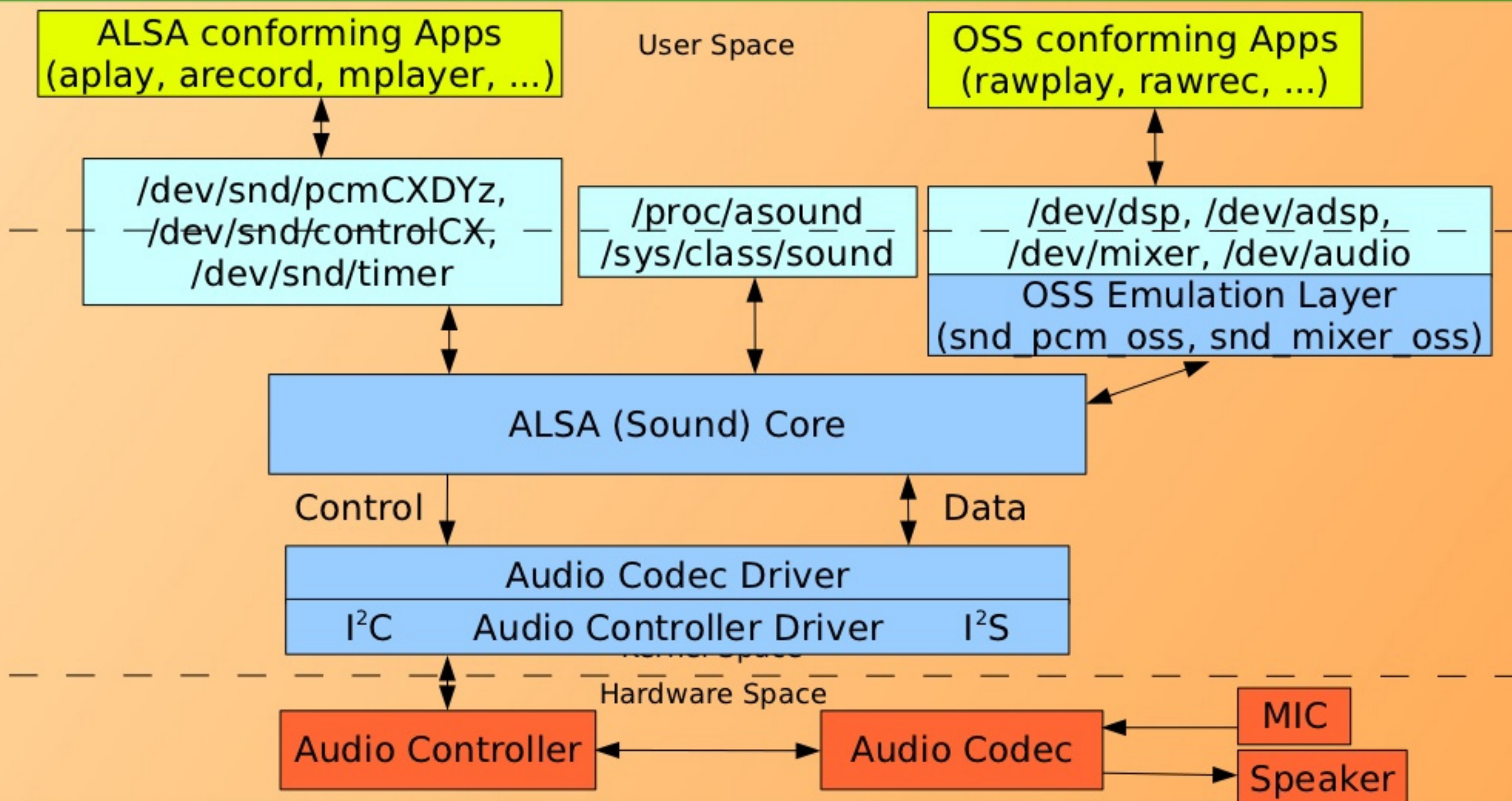


# Audio Drivers

# What to Expect?

- ★ Introduction to Linux' Audio Subsystem
- ★ Audio Vertical: Sound Core
- ★ Audio Horizontal
- ★ Putting them all together: Porting

# Audio Subsystem





# ALSA Sound Card Interface

- ★ Header: `<linux/sound/core.h>`
- ★ Data Structure: `struct snd_card`
- ★ APIs
  - ▶ `int snd_card_create(int idx, const char *id, struct module *module, int extra_size, struct snd_card **card_ret);`
  - ▶ `int snd_card_register(struct snd_card *card);`
  - ▶ `int snd_card_free(struct snd_card *card);`
    - Inverse of both the above

# ALSA PCM Interface

★ Header: <linux/sound/pcm.h>

★ Data Structure

- ◆ struct snd\_pcm
- ◆ struct snd\_pcm\_ops

★ APIs

- ◆ int snd\_pcm\_new(struct snd\_card \*card, const char \*id, int device, int playback\_count, int capture\_count, struct snd\_pcm \*\*rpcm);
- ◆ void snd\_pcm\_set\_ops(struct snd\_pcm \* pcm, int direction, struct snd\_pcm\_ops \*ops);
- ◆ int snd\_pcm\_lib\_malloc\_pages(struct snd\_pcm\_substream \*substream, size\_t size); // Typically used in hw\_params
- ◆ int snd\_pcm\_lib\_free\_pages(struct snd\_pcm\_substream \*substream); // Typically used in hw\_free



# struct pcm\_ops

- ★ `int (*open)(struct snd_pcm_substream *substream);`
- ★ `int (*close)(struct snd_pcm_substream *substream);`
- ★ `int (*ioctl) /* Miscellaneous controls */`
  - ◆ `(struct snd_pcm_substream * substream, unsigned int cmd, void *arg);`
- ★ `int (*hw_params) /* Set h/w params & allocate the buffer */`
  - ◆ `(struct snd_pcm_substream *substream, struct snd_pcm_hw_params *params);`
- ★ `int (*hw_free) /* Free the buffer */`
  - ◆ `(struct snd_pcm_substream *substream);`
- ★ `int (*prepare) /* Set audio parameters for transfer */`
  - ◆ `(struct snd_pcm_substream *substream);`
- ★ `int (*trigger) /* Trigger the transfer */`
  - ◆ `(struct snd_pcm_substream *substream, int cmd);`

# ALSA Sound Card Interface

- ★ Header: `<linux/sound/control.h>`
- ★ Data Structure: `struct snd_kcontrol_new`
- ★ APIs
  - ▶ `int snd_ctl_add(struct snd_card * card, struct snd_kcontrol * kcontrol);`
  - ▶ `int snd_ctl_remove(struct snd_card * card, struct snd_kcontrol * kcontrol);`



# Porting a Audio Driver

## ★ Standard Audio Codec

- Mostly involves changing pin assignments as per the Board Design

## ★ New Audio Codec

- Complete Driver as per the preceeding discussions, need to be implemented



# Browse some Audio Drivers

- ★ ALSA driver examples: sound/
  - arm/aaci.c
  - soc/soc-\*.c {core, pcm, utils}
- ★ Codec driver examples: sound/
  - arm/aaci.c
  - soc/codecs/twl4030.c
- ★ Browse & Discuss any

# What all have we learnt?

- ★ Introduction to Linux' Audio Subsystem
- ★ Audio Vertical
  - ▶ ALSA Sound Core & its Programming I/f
- ★ Audio Horizontal
  - ▶ Audio Codec Driver
  - ▶ Audio Controller Driver
- ★ Putting them all together: Porting



Any Queries?