

# Put a Bela On Your Head

Becky Stewart  
rebecca.stewart@qmul.ac.uk

# Disclaimer

We won't be creating head-tracking binaural audio with Ambisonics using Bela.

But we will be creating head-tracked stereo audio using Bela.

And there's a demo of a commercial product with head-tracked binaural audio using Ambisonics.

And we will go through what it takes to make your own head-tracked binaural audio system (and why it isn't on Bela (yet)).

# Disclaimer 2

I am not the person in this room that knows the most about Bela.

I am probably not the person in this room that knows the most about programming in Pure Data.

If either or both of these things are new to you, totally fine. Please ask questions and interrupt me whenever something doesn't make sense.

# Today We Will Cover

1. Some spatial audio with Pure Data
2. Introduce Bela and programming with Pure Data
3. Drink tea!
4. Working with Inertial Measurement Units (IMUs)
5. IMUs and Bela

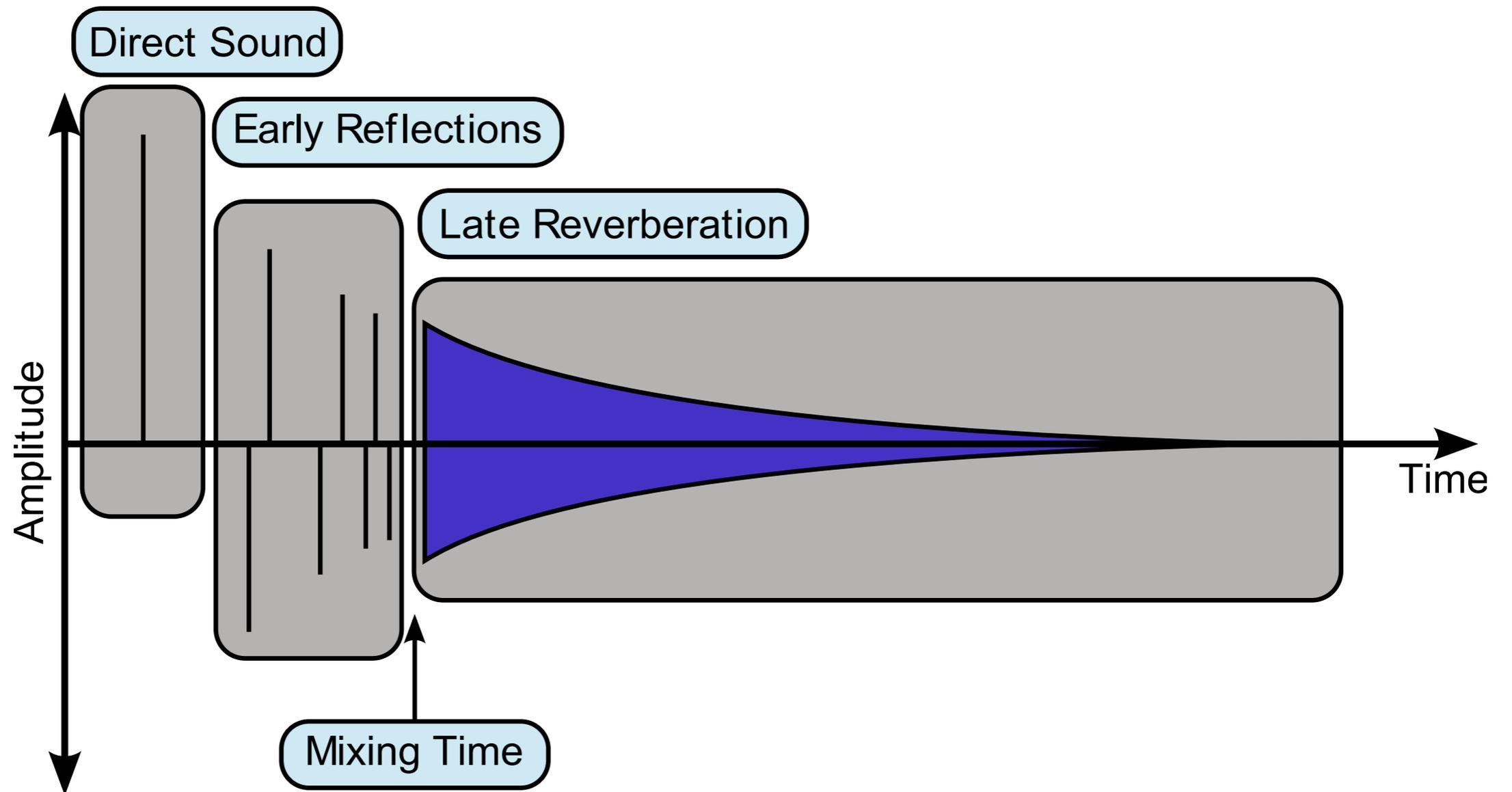
# Spatial Audio and Listening

# Spatial Audio is...

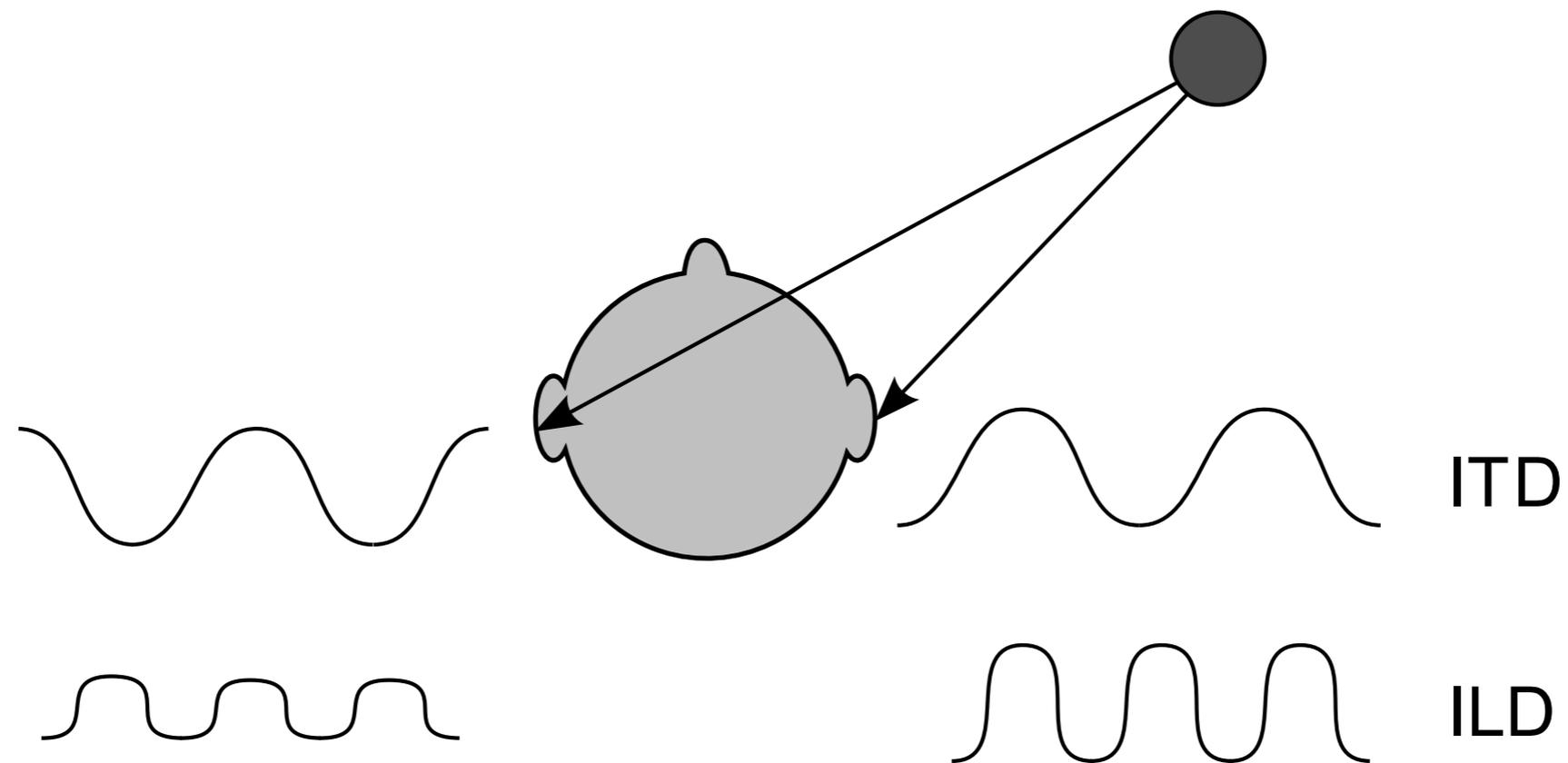
- Physical Science
- Perceptual Psychology
- Cultural Anthropology

(from 'Spaces Speak, Are You Listening' by Blesser and Salter)

# How do we locate a sound when we hear it?

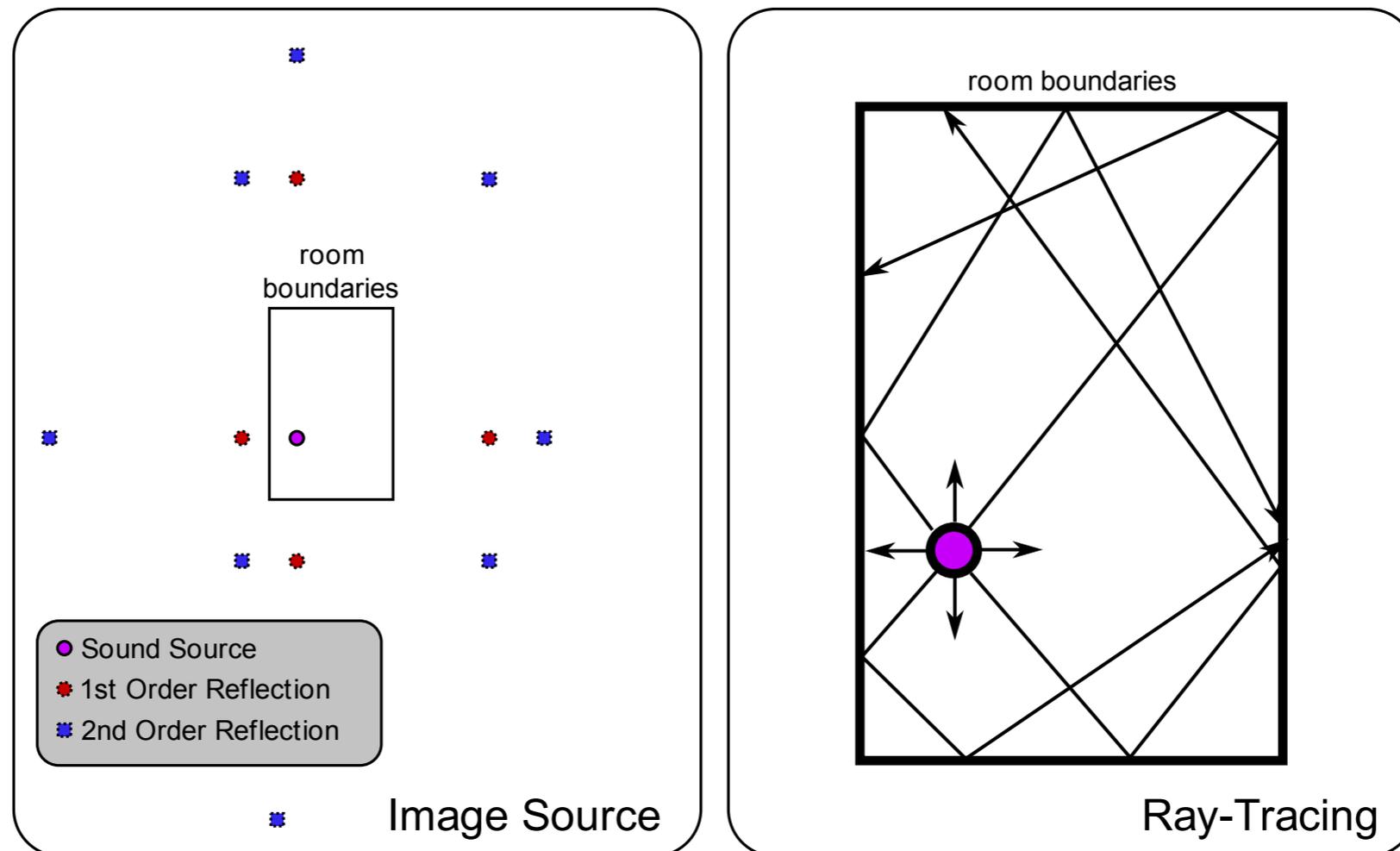


# How do we locate a sound when we hear it?



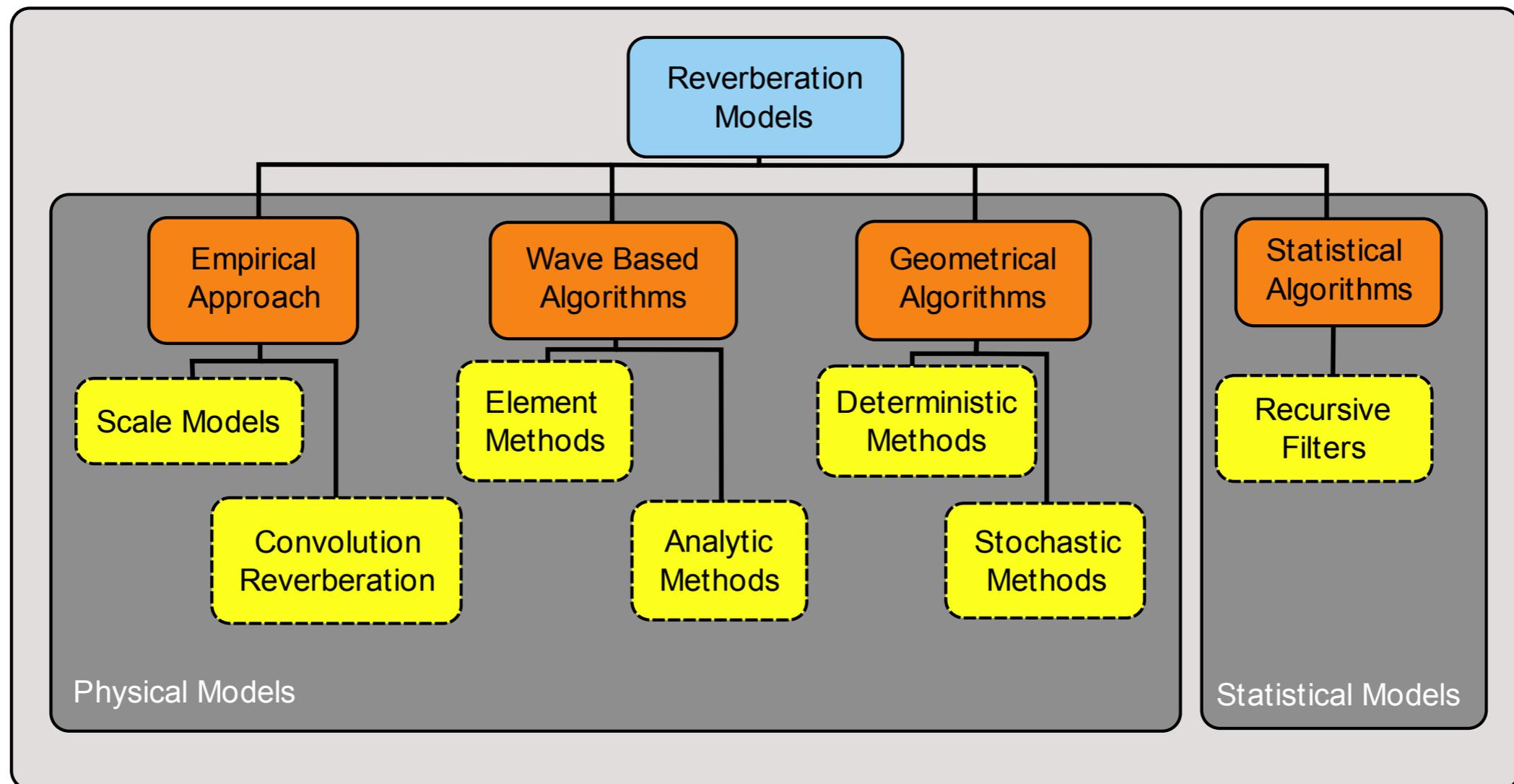
**Direct Sound**

# How do we locate a sound when we hear it?



## Early Reflections

# How do we locate a sound when we hear it?



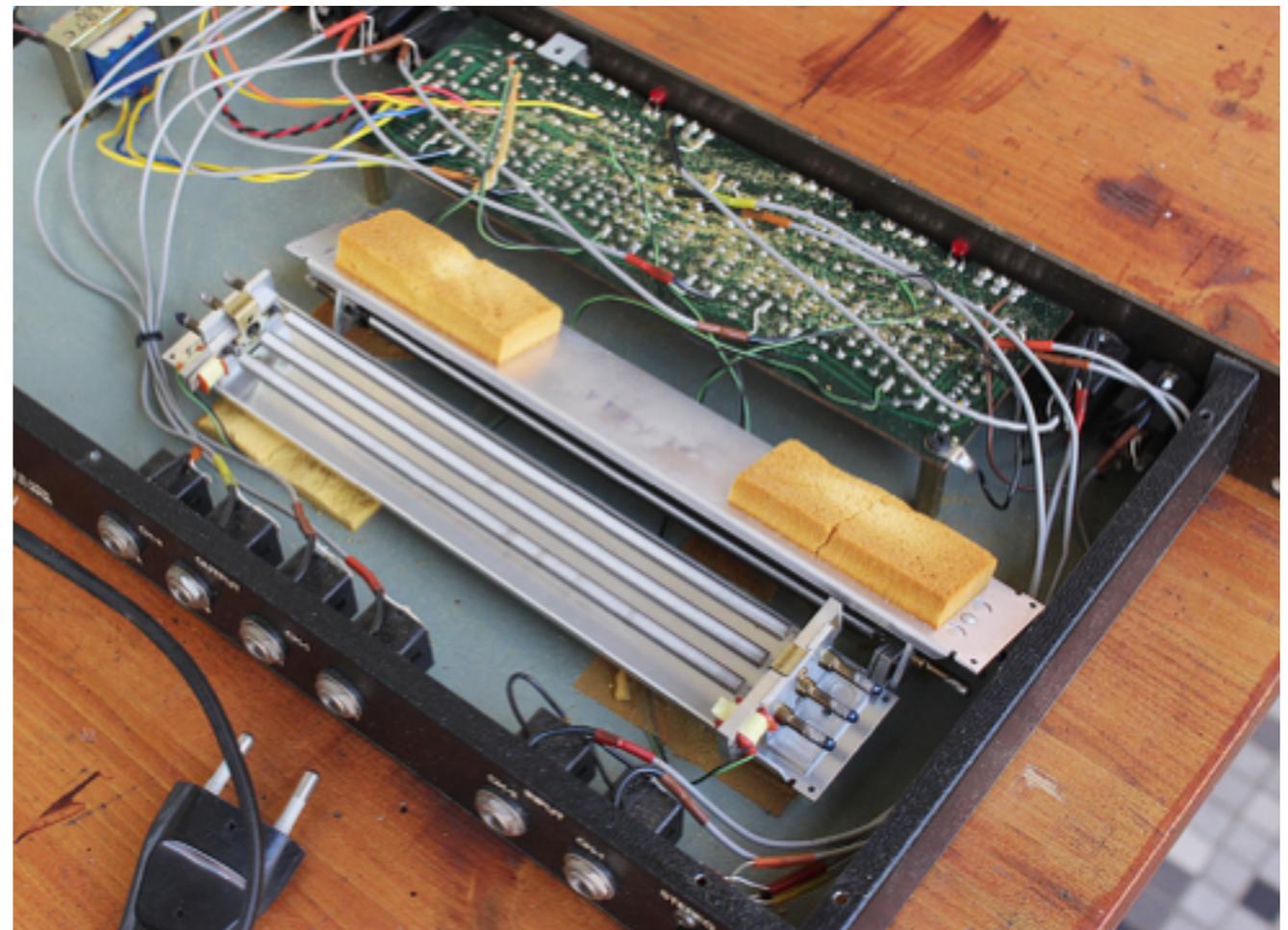
## Late Reverberation

# Spatial Audio in Broadcast/ Recorded Media

- We don't like the way ambient mics sound, especially with speech.
- Close miking provides the solution for removing or controlling those early reflections and late reverberation.
- We don't like listening to (acoustic) music without room information.
- A century of engineers have worked and continue to work on how reverberation can be added to a dry sound.

# Spatial Audio in Broadcast/ Recorded Media

- A century of engineers have worked and continue to work on how reverberation can be added to a dry sound.

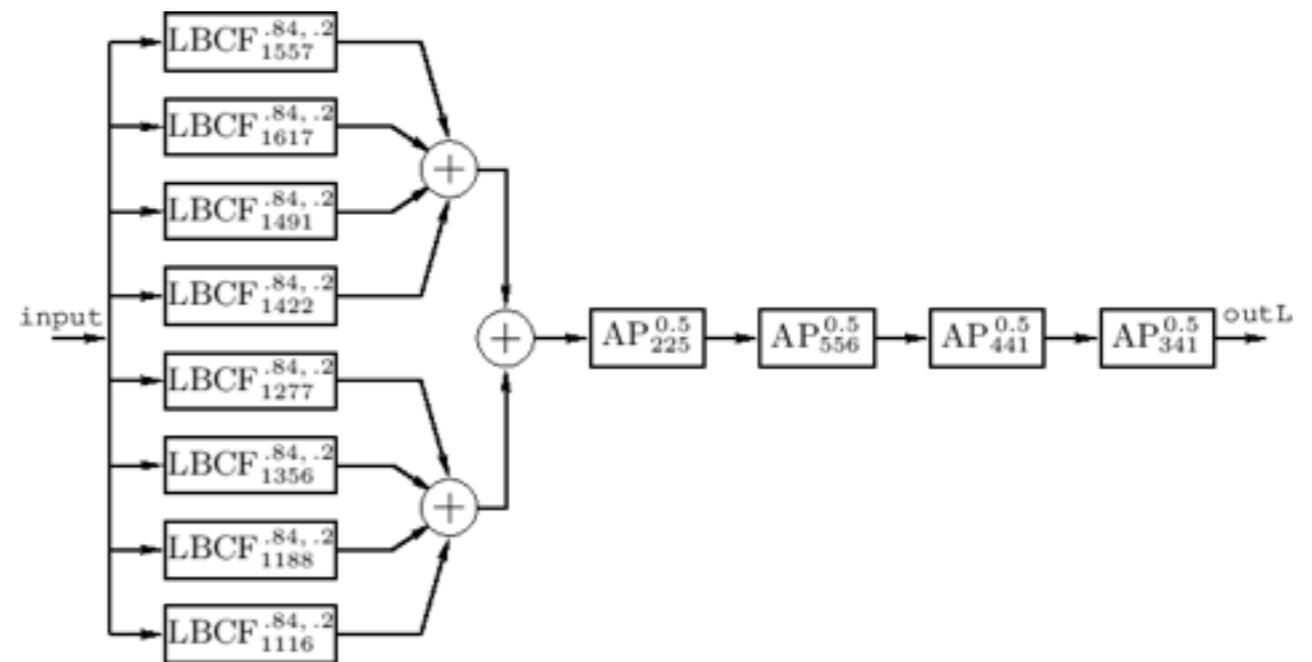


# Simple Reverb with Pure Data

# Freeverb

- Can't just use delay lines for many reasons.
- Freeverb is public domain C++ code
- vfreeverb~ is PD patch by forum user katjav and is available at

<http://www.pdpatchrepo.info/hurlleur/vfreeverb~alt.pd>



# Stereo Panning with Pure Data

# Panning

- Making a sound appear to be located in between multiple speakers.
- Lots of approaches, even for only two speakers.

Working with Bela

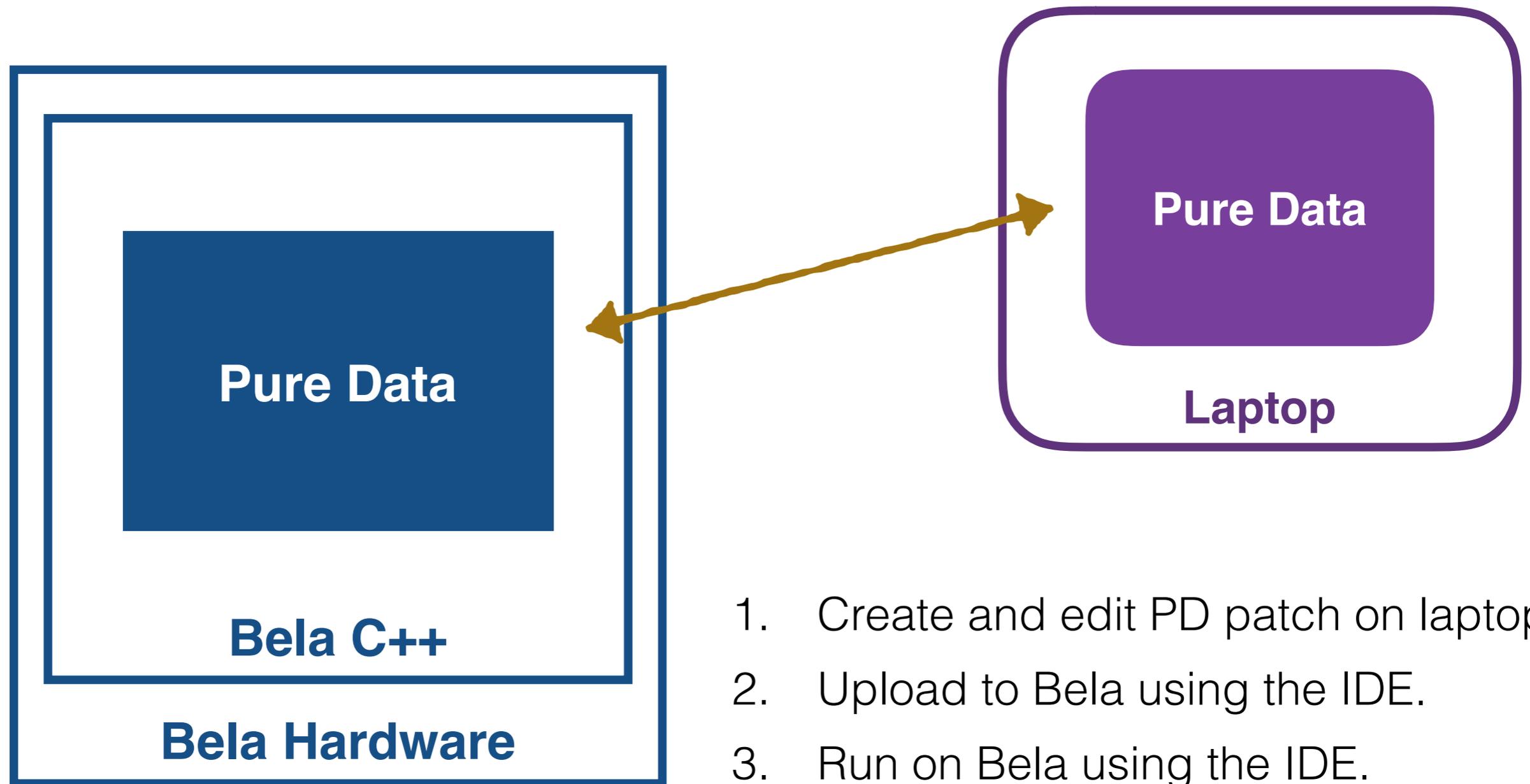
# Introducing Bela

- A computer without a screen, mouse, or keyboard with some extra features that make it super friendly with audio.
- Program it by connecting it to another computer with a USB cable.
- Can run from a battery, connect to external sensors, and very importantly prioritises audio over other tasks.
- See [www.bela.io](http://www.bela.io) for documentation and tutorials.

# Turning on Bela

1. Give the Bela a couple minutes to start up after you've given it power with the USB lead.
2. Open up a web browser and go to 192.168.7.2 instead of a website.
3. This is the Bela IDE, which lets you program and control the Bela. Anything you see here is happening on the Bela, not on your own computer.

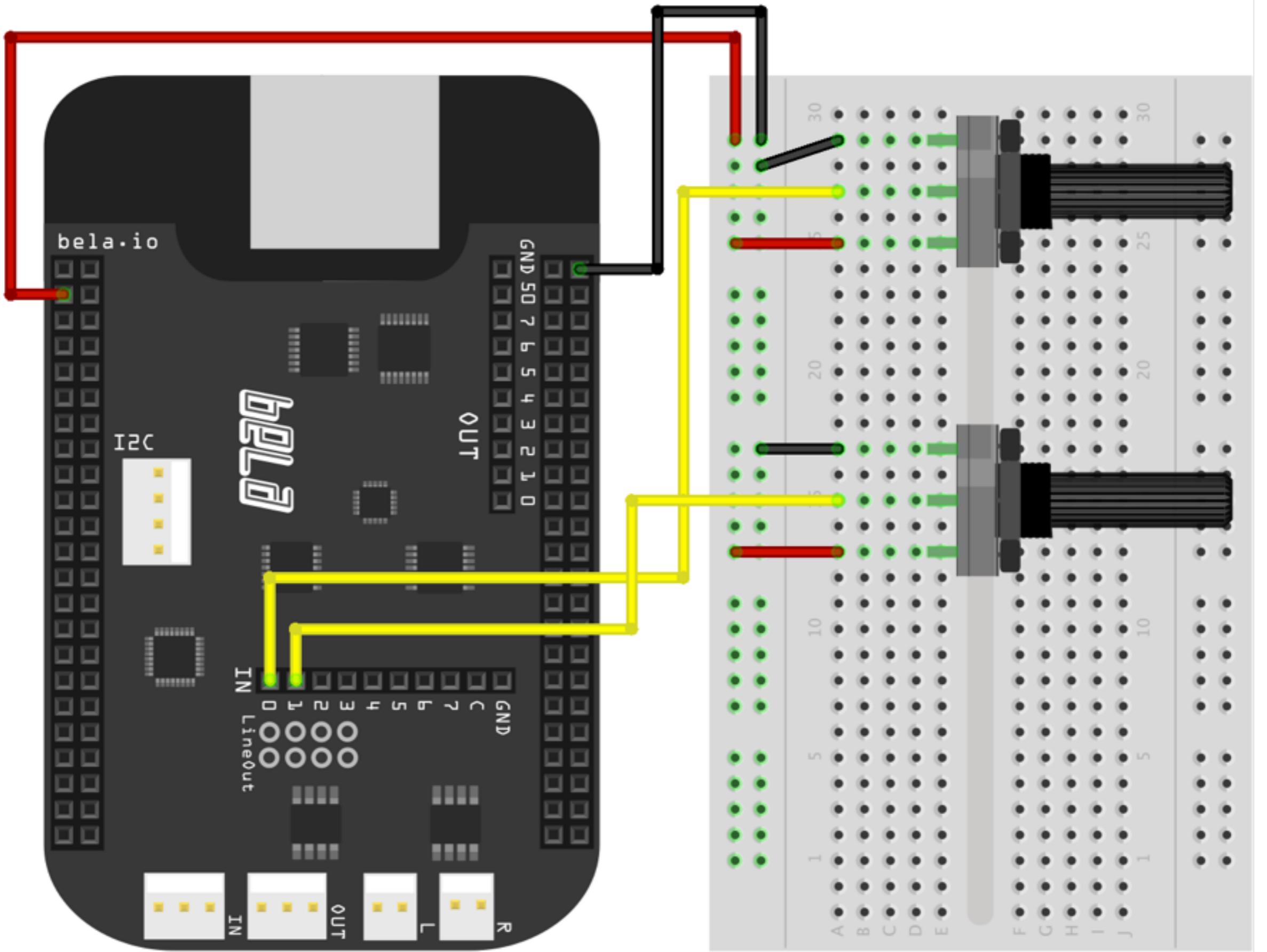
# Programming Bela Using Pure Data



1. Create and edit PD patch on laptop.
2. Upload to Bela using the IDE.
3. Run on Bela using the IDE.
4. If need to change the patch, edit on laptop and repeat process.

# Reverb on Bela

- Need to create a new project in the IDE and add our patch.
- ADC and DAC objects are now tied to Audio and Analog Inputs and Outputs.
- Can't interact with mouse, so add potentiometer to change reverb level.



# Panning on Bela

- Use potentiometer to pan a sound.
- Bonus: Make a friend to pan two sounds using two potentiometers on one Bela.

# Working with an Inertial Measurement Unit

# IMUs

- Contains multiple sensors: magnetometer; accelerometer; gyroscope.
- The data from the different sensors are combined to give some higher level information - this process is called sensor fusion.
- The sensor fusion algorithm needs to deal with drift and calibration issues. This is not easy!
- We are using the BNO055 by Bosch on a breakout board by Adafruit. The chip does all the hard work for us (though we also don't know exactly what it's doing).

# IMUs

- We are using the BNO055 by Bosch on a breakout board by Adafruit. The chip does all the hard work for us (though we also don't know exactly what it's doing).
- This sensor was selected as it had good performance in a study was recently published with the AES.
- The study's source code available online as MrHeadTracker - <https://git.iem.at/DIY/MrHeadTracker/wikis/home>

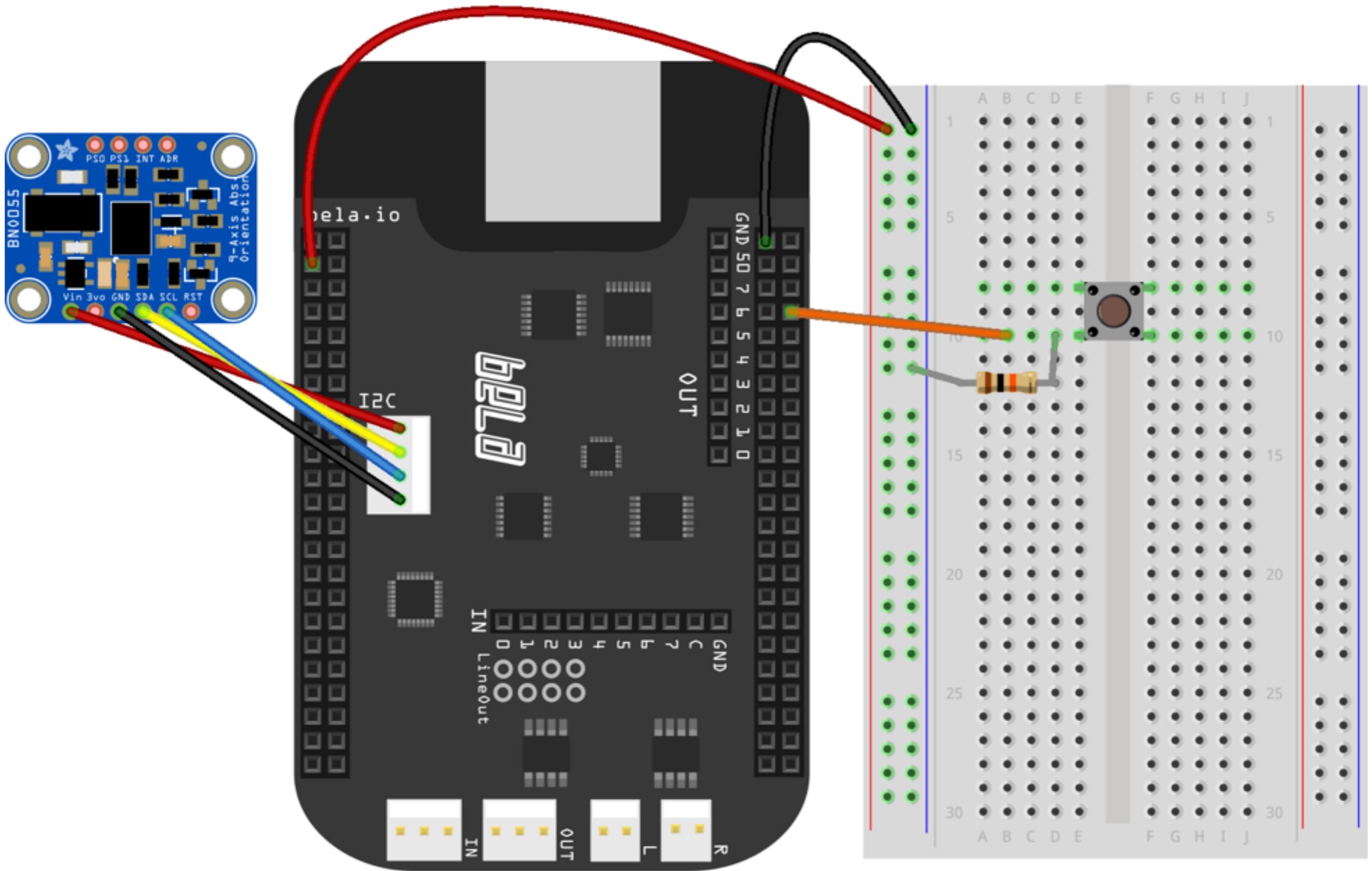
# IMU Synth

- Create project from imu-sine-synth-pd in the repository.

# Head-Tracked Stereo Audio with Bela

# Head-Tracked Stereo

- We need to use the custom `render.cpp` file in the repository in order to receive data from the IMU.
- Add a button with a pull-down resistor to P8\_08. This walks through the calibration process:
  - Hold the IMU in the neutral position (looking forward); press and release the button.
  - Hold the IMU in the down position (as in looking down); press and release the button.
  - This can be repeated at any time to change the orientation of the sensor.



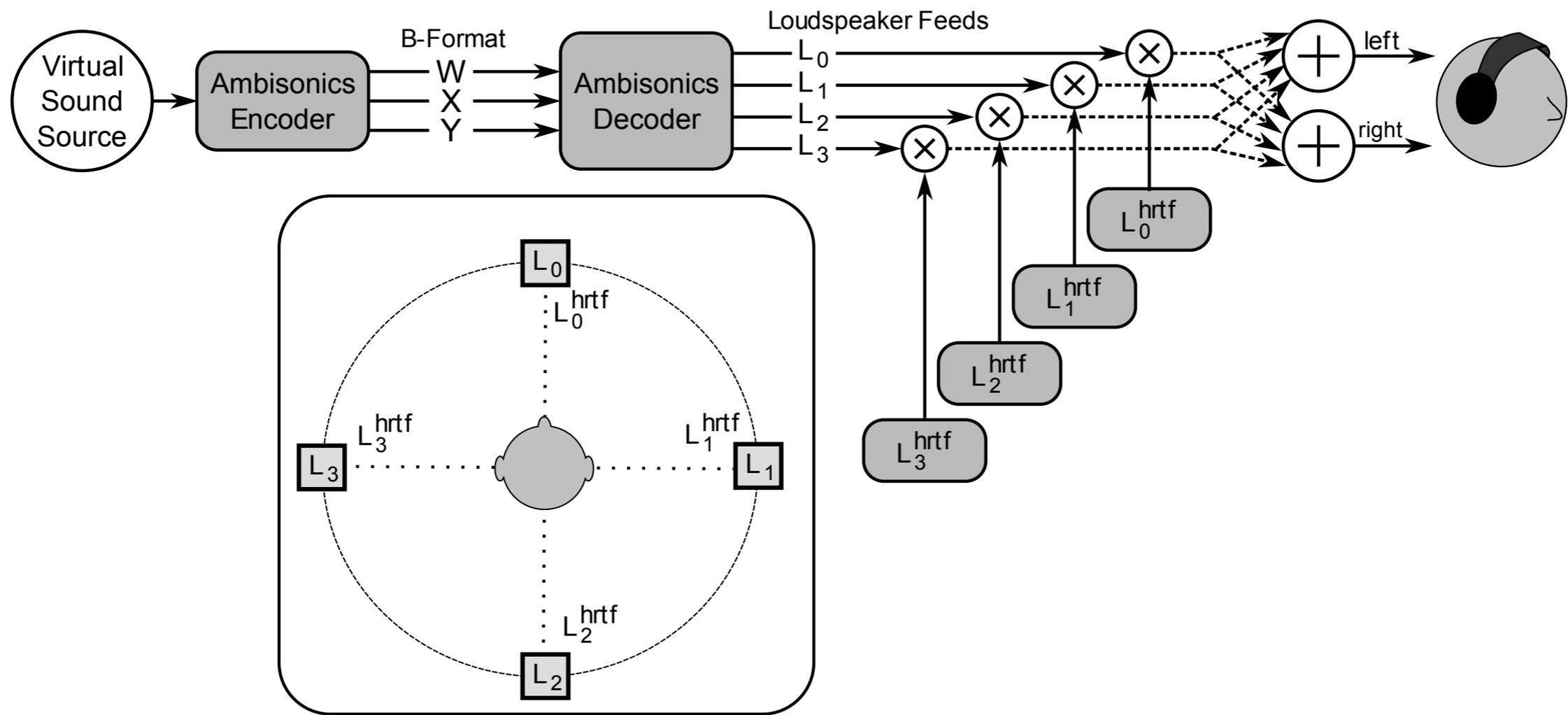
Add Reverb to the  
Head-Tracked Audio

# Binaural Audio

# Beyond Stereo to Binaural

- Binaural means that the filtering effects of the head and maybe torso are included in the audio.
- Ideally the audio sounds like it is from outside the head, not between the ears.
- Stereo panning simulates ILD and/or ITD, it doesn't directly change

# Virtual Loudspeakers



# Audio Design for VR