Introduction to the WFS system in H0104

1 Room and hardware setup

The lecture hall H0104 is fairly large, providing seats for ca. 700 people. The loudspeakers are mounted on all walls surrounding the hall.

The loudspeakers are built into loudspeaker panels, each providing 8 audio channels and which can be fed with an ADAT signal. Each panel additionally has 2 larger speakers which emit the low-pass filtered sum of the 4 channels above it.

To drive these speakers a cluster of Linux computers is used, so that each computer can compute the loudspeaker signals for a number of loudspeaker channels (varying between 48 and 64 channels). For this, each computer is equipped with an RME HDSP MADI sound card. Each MADI output is connected to an MADI to ADAT bridge (RME ADI648), which is built into the wall, so that the ADAT cables can be kept short. The input to the system comes from a mixing desk and is multiplexed to each MADI sound card with the use of MADI bridges (RME MADI Bridge). This audio chain is synchronised with the use of Word-clock.

The cluster communicates with each other via Ethernet, using the OSC-protocol. An overview of the hardware setup is given in figure 1.

2 Software architecture

The software is divided in several parts: a user interface, a score player, a control unit, a real-time render unit, an offline render unit and a common library for general functions.

The user interface provides dialogs for array configuration, grid point configuration (possible source positions and their characteristics), composition and a real time control interface. In the real time control interface, it is possible to move sources around with the mouse, as well as to store different scenes, between which can be switched.

The score player is used to synchronise with an audio player and record and playback source movements.

The control unit parses information from the user interface into useable data for the render engines and provides feedback to the user interface and score player. The render engines take care of the actual audio signal processing.

There is an offline render unit, which takes care of all calculations that cannot be done in real-time, but can benefit from parallel execution on a cluster.

Communication between the different parts of the program is based on the OSC protocol. Figure 2 gives an overview of the program parts and their communication.

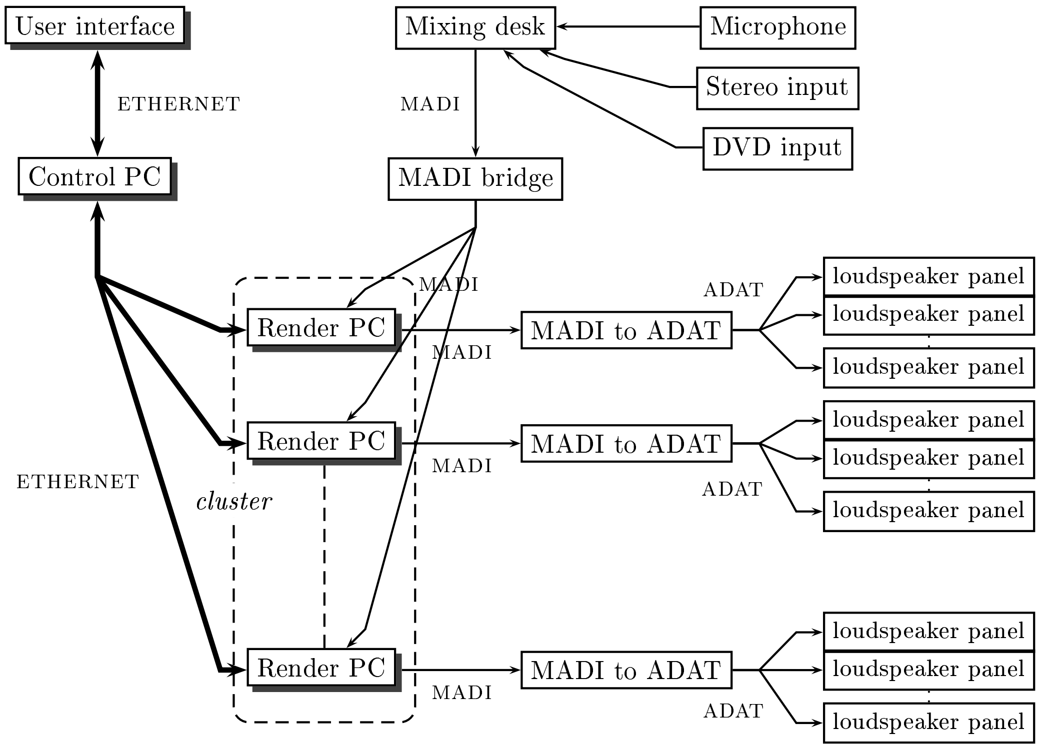

Figure 1: Schematic overview of the hardware setup for the WFS system in the lecture hall of the TU Berlin.

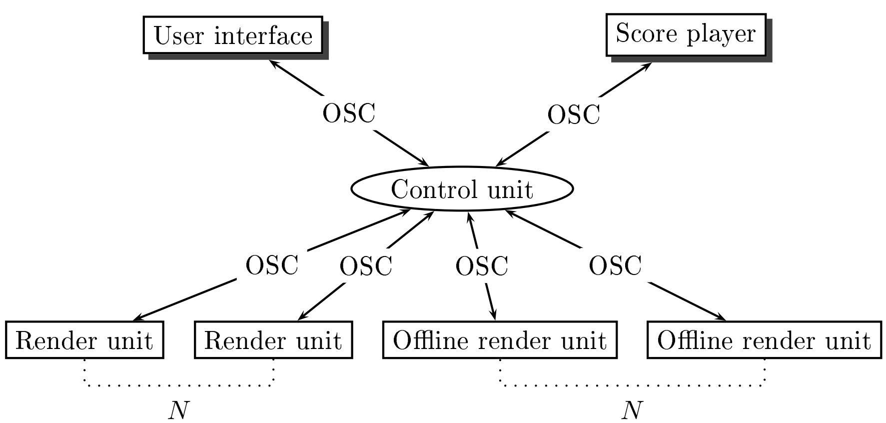

Figure 2: Schematic overview of the different parts of the software *sWONDER*

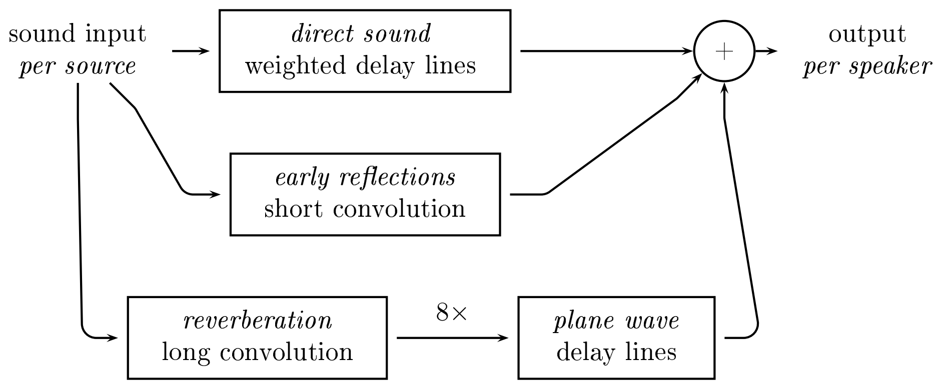

Figure 3: An overview of the audio signal processing by the real-time render unit

In figure 3 an overview is given of the audio rendering setup.

3 Sonic possibilities for the LAC

For the LAC2007, the following possibilities will be available:

point source A sound source with a specific location in space

plane wave A sound source with a specific direction

Transitions between these two source types are also possible.
The system can render 48 sources simultaneously.

3.1 Movement

Point sources can be moved freely through the space. Here there are two options: (1) a realistic movement, which will automatically cause a Doppler effect if the source moves fast. (2) a movement without Doppler effect; this is achieved by fading the source between static points. The update frequency of this can be configured.

3.2 Limits of the position

When the rendering is started, it must be determined within which space the sources will move: the source can move in front of the speakers. The extent to which it does this, needs to be determined, as the most extreme position here determines the necessary predelay. The other limit is how far the sources can go away, that determines the maximum delay time.

3.3 Reflections

The offline renderer is not yet completed.

But the following options will be possible to add room effects:

mirror image sources Additional source points, playing back the same signal as another one, can function as mirror image sources to add early reflections.

reverb through plane waves A signal can be convolved with an impulse response, and the result of this can be played back as a plane wave. In this way, reflections from several directions can be added.

3.4 Scenes

It is possible to store scenes of sound source constellations, which can be later recalled. It is also possible in this way to make transitions between scenes, which means that each source with a certain position in scene 1 will move to its position in scene 2.

4 How to...

To use the system, there are the following options:

- Use Ardour, or another Linux multitrack-program that supports LADSPA, and our LADSPA plugin, to control the movement of the sources (Ardour can run on the control computer, so we can provide this).
- Use your own music software, and send OSC-messages directly to the control unit. The OSC-commands our software understands will be provided in another document.

Our graphical interface is not yet completely finished, so I cannot say yet whether it will be available or not. The score recorder/player is also still in development, but may be in a phase where we can test it.

5 Time and other practical considerations...

Before the conference, we will be able to access the lecture hall in the hours that it is free; this will be mostly in the evening hours. Please let us know when you would like to work, and then we will try to reserve the hall for these times.

Let us know what kind of things you will want to try out, so that we can prepare for this. If necessary we may also implement new options on demand...

Also, please inform us on how you would like to interface the system (directly from your own software, or with Ardour/LADSPA).

6 Working OSC commands

command	types	arguments
/WONDER/project/create	s	projectname
/WONDER/project/load	s	projectname
/WONDER/project/save	s	projectname
/WONDER/scene/add	i	scene no.
/WONDER/scene/select	iff	scene no., time, duration
/WONDER/scene/remove	i	scene no.
/WONDER/scene/set	i	scene no.
/WONDER/source/position	ifffff	srcid, pos x, pos y, pos z, time, duration
/WONDER/source/angle	ifff	srcid, angle, time, duration
/WONDER/source/type	iiff	srcid, type, angle, time

6.1 Project

To be able to store scenes, you need to create a new project with the command: `/WONDER/project/create`, with one *string* as argument: the project name.

You can save the project with the command: `/WONDER/project/save`, and later load it again with the command `/WONDER/project/load`.

6.2 Scenes

You can create a snapshot of the current source positions (called a "scene") and store them in the project, using the command `/WONDER/scene/add` with an integer as argument for the slot number under which you want to store the scene.

Later you can recall the scene with the command `/WONDER/scene/select`, with as arguments the scene number, the time at which the change to the scene should start, and the duration in which it should fade to the new scene.

With `/WONDER/scene/remove` a scene is deleted (and thus the slot is freed again). With `/WONDER/scene/set` you can overwrite an existing scene. Note the subtle difference between adding a scene and setting a scene: adding creates a new scene and stores the current source positions to it. It gives an error back when the scene number already exists. "Set" stores the current source positions to an existing scene and gives an error back if the scene slot does not exist.

6.3 Source control

There are two types of sources: point source (see fig. 4) and plane wave (see fig. 5).

With the command: `/WONDER/source/type` you can set the type for one source. Plane wave is "0", point source is "1". The angle argument is the start angle for the plane wave. Whenever the type is changed you should also send a `/WONDER/source/position` command, to set the position of the source. In the case of a point source, this will be the actual position of the source. In the case of a plane wave, this is a reference point for the calculation; it should be chosen to be a position somewhere behind the array in the direction where the plane wave is coming from. This point determines the basic latency of the plane wave.

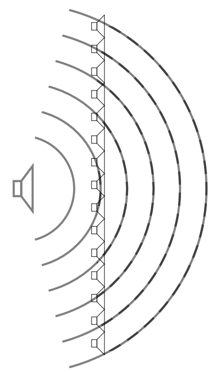

Figure 4: A point source.

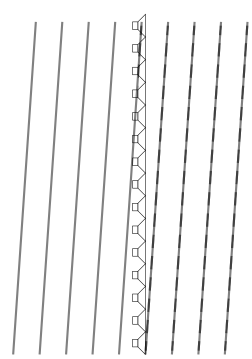

Figure 5: A plane wave.

`/WONDER/source/position` takes as arguments the source id, the x and y position (in meters), the z position (which should be 1.0 for now), the time at which the change should start (in seconds from "now"), and the duration for the change to take place (also in seconds).

`/WONDER/source/angle` takes as arguments the source id, the angle, the time at which the change should start, and the duration for the change to take place.

The coordinate system is illustrated in figure 6.

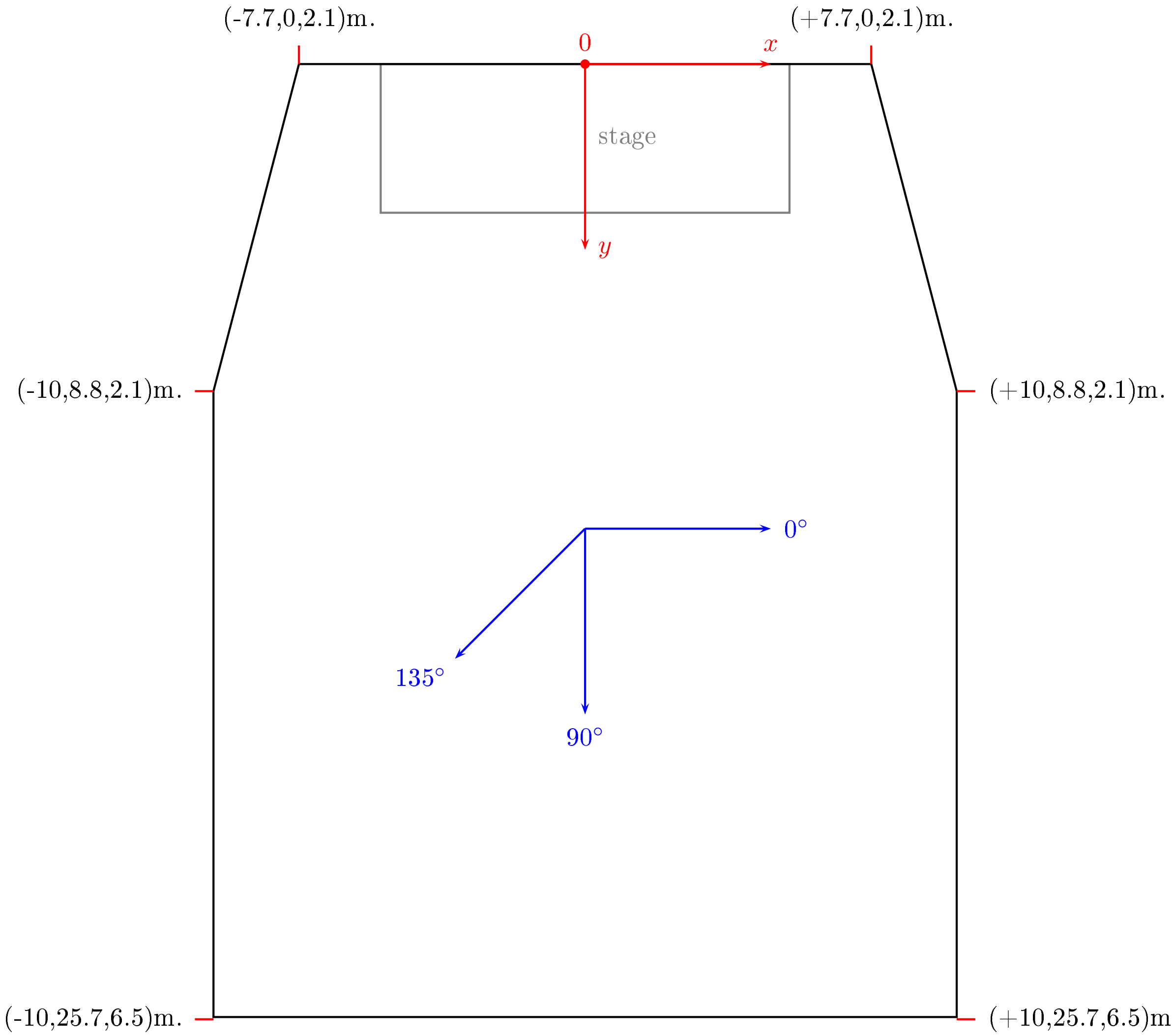

Figure 6: The coordinate system of H0104. The origin is in the middle of the front wall. The hall has a slight elevation; the coordinates of the corners of the hall are indicates. The z-coordinate is approximately the height of the speakers. Angles: the arrows show in which direction the wave fronts of the plane wave are traveling.