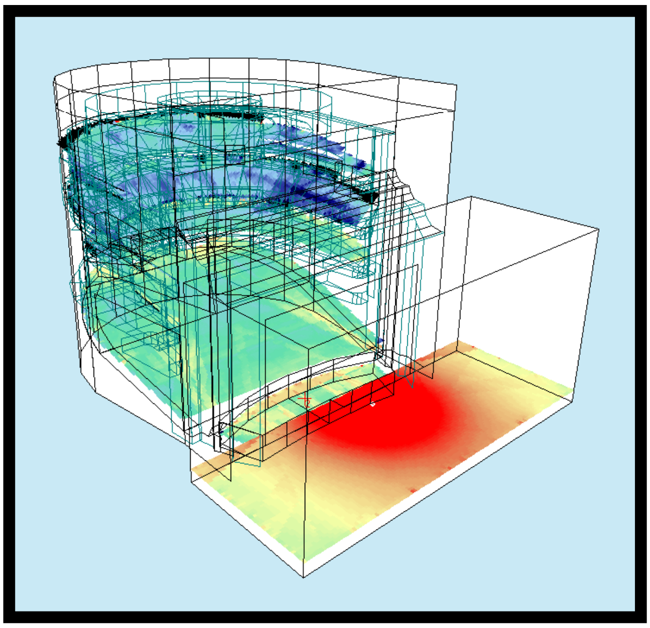
ODEON Room Acoustics Program

Version 6.5

User manual

Industrial,
Auditorium
and
Combined
Editions



Front figure: The Opera House project for Ankara Congress and Cltural Centre (Architect: Özgür Ecevit, Acoustics: Jordan Akustik, Denmark)

ODEON Room Acoustics Program
Version 6.5, Industrial, Auditorium and Combined Editions

by
Claus Lynge Christensen

ODEON A/S
c/o Acoustic Technology, Ørsted•DTU,
Ørsteds Plads
Technical University of Denmark
Building 352, DK-2800 Lyngby
Denmark

http://www.odeon.dk
E-Mail:odeon@oersted.dtu.dk
Fax: +45 45 88 05 77
Phone: +45 45 25 39 30

2003

Introduction

This manual is intended to serve as an introduction on modelling room geometries and to the facilities in ODEON. It will not cover in depth all facilities; explanations of displays, calculation parameters, results, etc. are available as context sensitive help from within the program (shortcut F1). It is recommended to use the online help to learn about the specific features available from the different displays, the interpretation of results, calculation parameters, etc.

The contents of this manual are as follows:

Chapter 1 covers installation of the program, changes from previous versions etc.

Chapter 2 is a Short guided tour, introducing the ODEON program and its facilities, offering a short guided tour to the operation of the program, including assignation of calculation parameters, definition of receivers, receiver grids, different kind of sources and the presentation of results. Section 2.1 is intended for the Auditorium and Combined editions. Section 2.2 is intended for the Industrial edition. Most of the description on how to operate the program is found here.

Chapter 3 covers the modelling process, it is recommended that users of early versions of ODEON do also read this chapter as a new and far more flexible modelling format is available, supporting the modelling of symmetric and semi-symmetric rooms, use of constants, variables, counters etc.. ODEON also allows import of CAD models using the DXF format available from AutoCAD and other CAD programs. Also take the time to read about how to verify room models.

Chapter 4 deals with the materials to assign to the surfaces of the rooms, scattering and transparency coefficients. Special materials that may speed up the modelling process and how to extend the material library is also covered in this chapter.

Chapter 5 deals with the auralization options in ODEON Auditorium and Combined; the hardware requirements, how to publish calculated sound examples on the Internet or on audio CD's etc.

Chapter 6 introduces the calculation principles used in the ODEON program. It should not be thought of as a thorough description of all the calculation principles used, merely a short introduction that may give an idea on the capabilities and limitations of the program.

Chapter 7 describes the calculated point response parameters available in ODEON, how they are calculated and how to interpret the results.

Chapter 8 describes the various calculation parameters available in the program. Most of the parameters are automatically set to reasonable values by ODEON, however for special cases you may find adjustment of some of the calculation parameters useful.

Chapter 9 is the discussion on quality of results and how to achieve good results. This chapter may be relevant one when familiar with the program.

Chapter 10 describes how to extend the library of directivity patterns available for point sources, e.g. if you wish to define your own loudspeakers etc.

Happy modelling, Claus Lynge
Lyngby, April 2003.

Index

1 Installation and running the program

1.1 Installing and running the program

ODEON comes on a CD-ROM containing the three different editions of Odeon and a demo version of the program. To install the program:

a) Double click on the Edition you wish to install.
b) Run the Setup.exe program and follow the instructions during the installation process (this will install the ODEON program onto your computer).
c) To run the program, the supplied hardware HASP key must be inserted into the parallel port on the PC (or USB port if its a USB key). If you start the program without the hardware HASP key it can only be used as a viewer program.

On the disk you will also find the directories DemoVersion and Install Noise Explorer.

ODEON Demo Version

Free demo version, which allows you to carry out all the operations available in Odeon 6.5 Combined. The only limitation to the program is that you cannot make any calculations to new or modified rooms. You will only be able carry out calculations on the supplied rooms and to view results, which has been calculated in a registered version of Odeon.

Feel free to pass on a copy of the demo version or if the demo has been downloaded from the Internet to pass on the odeon.zip file. The most resent demo version of Odeon can be obtained at http://www.odeon.dk.

Installing the Demo Version

Run the Setup.exe program from the DemoVersion\ directory and follow the instructions during the installation process (this will install the Odeon Demo Version onto your computer).

Upgrading your software license

The program Noise Explorer is used here for upgrading Odeon software licenses.

The hardware HASP key contains coded license information that is used to "unlock" different the software.

To upgrade your license install Noise Explorer from the Install Noise Explorer folder onto your computer and make sure the hardware HASP key is attached to the parallel port.

a) Run the Noise Explorer program
b) Choose the Tools menu and select License Utility...
c) A window appears showing what licenses are installed on the hardware HASP key.
d) Note down the Key Identification code (top right corner of the window).
e) This Key Identification code must be given to *Brüel & Kjær* when an upgrade is ordered, so a new license code can be generated.

When you have received your new License Code:

a) Start the Noise Explorer program
b) Choose the Tools menu and select License Utility...
c) Type in your newly received code in the License code field.
d) Press Install.
e) Close the dialog box.

You can now use all the facilities of the new version of the software.

a) Run the Setup.exe program from \Install\Disk1 and follow the instructions during the installation process (this will install the ODEON program onto your computer).
b) To run the program, the supplied hardware lock must be inserted into one of the parallel port on the PC.

1.2 Troubleshooting

Odeon makes heavy use of facilities, which are built-in to the Windows operating systems. If parts of the user interface in Odeon is malfunctioning or looking odd and your computer is running an edition of Windows installation, which has not been updated recently then it is likely that it will help updating the operating system. The Windows update is available at www.microsoft.com.

1.3 Upgrading from previous versions

If you are upgrading from previous versions of ODEON, it is recommended that you read this chapter, in order to be up to date with the changes in ODEON.

Preparing geometry, Trace Rays etc.
If you upgrade from a version earlier than 3.0 you should notice that many operations are carried out automatically without you having to care about it e.g. the Prepare Geometry and the Trace rays processes. The 'Short Guided Tour' in chapter 2 will take you around this version and allow you to get a feel of how to operate the program.

Project files
The only project file from versions earlier than 3.0, being fully compatible is the surface file (.SUR). The rest of the project files are no longer valid.

Directivity files (refereed to as source types in ODEON 2.xx)
ODEON 3.0 and later version uses eight bands of frequency information. Thus, previous directivity files (e.g. OMNI.SOU) are no longer valid. You can translate your old directivity files into new eight-band directivity files (e.g. OMNI.SO8) using the Tools|Directivity patterns|Translate 6 band into 8 band menu entry in the Odeon programe

New features
If upgrading from ODEON 3.1 or earlier versions of ODEON, the guided tour in chapter 2 and chapter 3 on modelling is indeed recommended reading. Stepping through these chapters will save much time later on - in particular its is important to be familiar with the new geometry modelling language and /or CAD import options before starting large modelling projects.

If upgrading from ODEON 4.0 or later, a list describing the most resent features, is available from within the program. To view the list, activate the help from within ODEON using the F1 shortcut, then select the Contents tab|Whats new in Odeon… entry.

Upgrading from Odeon 4, 5 and 6 to Odeon 6.5
If having problems loading a room, which was created in one of above the listed versions of Odeon and which worked fine in these versions, this is probably be due to a change that has been made to the surface numbering mechanism which is applied in Odeon. The numbering mechanism has been changed slightly in order to avoid a conflict, which appeared when using 'symmetric surfaces' along with modelling entities such as CountSurf, Box, Cylinder etc. and in particular to make the automatic surface numbering work without problems (when the NumbOffSet is set to Auto).
If having problems loading a room due to the reasons just mentioned, Odeon will either give an error message that surfaces are repeated in the geometry file or that materials are not applied to all surfaces. In these cases you may wish Odeon to use the Old numbering mechanism – this can be done using the Version4 flag in the .par file; As the first line in the geometry file, just after the ### sign, type: Version4 TRUE

2 Short guided tours

This chapter will give an introduction to the use of the ODEON program. Depending on the edition purchased; the guided tour differs. The Combined and Auditorium editions are covered in section 2.1 and the Industrial edition is covered in section 2.2.

Buttons, hints and menus
The most common operations can be carried out using buttons. Pointing the mouse on a button will display a small 'bubble' telling the function of that button (a hint). You can also operate the program using menus or shortcut keys. Less common operations are available from the drop downmenu in the top of the Odeon program window – menus will change in order to facilitate the currently selected window or indeed the selected tab in the currently selected window. If needing some facility in a window, it is likely that its hidden in the dropdown menu.

Context sensitive help
Context sensitive help is available using the *F1* shortcut key throughout the program. The help includes description of the facilities available in a particular window, suggestions on the choice of calculation parameters, hints on the evaluation of calculation results, etc. Answers to to questions which goes on a specific window is found in the context sensitive help rather than in this manual.

Saving data and maintaining consistent results
The ODEON program automatically saves the user-entered data, such as sources and materials with the room. Whenever data need to be defined in order to carry out calculations, ODEON will prompt the user whether to accept or discard of changes. If the changes are accepted ODEON will automatically erase results that are no longer valid, ensuring that results are always consistent with the user data entered.

2.1 Short guided tour - Combined and Auditorium editions

Run the ODEON application

You will find the ODEON program at the Windows Menu Start|Programs|ODW6.5|Odw. Execute the program and begin the tour.

Open a room model to work on

Click the Open a room model button to select a room. The room files containing the geometries for ODEON carry the extension .par (or .sur for compatibility with previous version of ODEON) and are plain text files following the specifications outlined in chapter 3. For this guided tour select the room model named Example.par.

3D View

Have a look at the room. Whenever ODEON loads a room, it is displayed in a 3DView. This allows you to investigate the geometry and check it for errors, etc. Several facilities are available in the 3DView, e.g. rotation, zooming, highlighting of selected surfaces and corner numbers etc. Hit the F1 shortcut to get an overview of the facilities and their use.

Having assigned a room, this is a good time to get familiar with the MDI concept (Multiple Document Interface). At this point the title bar of the 3DView will be blue (or some other colour) indicating this is the active window. Being the active window, the 3Dview menu item is added to the menu bar next to the toolbar dropdown menu. You can operate the functions of the window using this menu or the shortcut keys displayed in the menu.

Define sources and receivers

Before any calculation can be carried out by ODEON, at least one source will have to be defined. Also a receiver will have to be defined in order to calculate a point response.
In this guided tour we shall define point, line and surface sources although only the point source is relevant to this auditorium type of room. Finally we define a receiver.
Click the Source-receiver list button at the toolbar to open the Source-receiver list from which sources and discrete receivers are defined. If the Source-receiver list is already open, but hidden behind other windows, etc., clicking this button will rearrange the windows as needed.

Define a point source

Click the New point source button in the local toolbar at the right side to open the Point source editor. Enter the values x = 3 (metres), y = 2 (metres) and z = 1.2 (metres)[1,2]. If you are not sure of the position of the source, you can select the 3D Edit source display. If you do so, you should notice how the menu item 3D Edit Source appears on the dropdown menu, when this window becomes active. The 3D Edit Source-Receiver menu will allow you to operate the 3D display, e.g. use the SPACE key to switch between different predefined views.
Finally set the overall gain to 65 dB. To save the new source just close the Point source Editor and confirm.
If you are running the Auditorium edition of ODEON, define two extra point sources of your own choice and go to the paragraph 'Activate sources'. Otherwise if you are running the Combined edition continue below, defining a line and a surface source.

Define a line source (Combined edition)

Click the New line source button to open the line source editor. Enter the values x = 4 (metres), y = 2 (metres), z = 2 (metres), Length = 2 (metres) and Azimuth = 135°.
Finally set the Overall Gain to 65 dB. To save the new source just close the Line source Editor and confirm.

Define a surface source (Combined edition)

Click the New surface source button to open the Surface source editor. Select surface 2001 End wall behind podium for this source and click the Invert normal checkbox to make the source radiate into the room.
Finally set the Overall gain to 65 dB. To save the new source just close the Surface source Editor and confirm.

Define a receiver

Click the New receiver button to open the Receiver editor. Enter the values x = 18 (metres), y = -5 (metres) and z = 3 (metres). To save the new source just close the Receiver Editor and confirm.

---

[1] Hint; Use the Tab or Shift+Tab keys to move between fields.
[2] Depending on the language selected on your computer '.' or ',' is used as decimal point.

Define other receivers at:

(x, y, z) = (12; 3; 2.2)
(x, y, z) = (8; 7; 1.5)
(x, y, z) = (21; 1; 3.6)

We will get back to the receivers and sources under the point: Calculating Point Responses.

Assign material properties

Open the Materials List and see how to operate it in the Materials menu.

Assign the following material data to the surfaces in the model:

Surface number	**1001**	**1002**	**2001**	**-2002 2002**	**-2003 2003**	**2004**	**3001**	**3002**
Material	901	905	702	702	702	702	702	702
Scatter	***0.7***	***0.7***	0.1	0.1	0.1	***0.7***	0.1	0.1
Transparency	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Hit the F1 shortcut to learn more about scattering and transparency coefficients. The transparency coefficient allows you to easily model a reflector panel build from many small surfaces, simply by modelling one surface and applying an appropriate transparency coefficient.

Quick Estimate, fast estimation of Reverberation Time

From within the Materials List run the Quick Estimate to get an idea of the order of the size of the reverberation time. Note the longest reverberation time. This calculation is very useful while assigning materials for the evaluation of different materials and their impact on the overall reverberation time. Before leaving Quick Estimate you may want to try this out by selecting different materials. It is also possible to select among the defined sources. However, the source position will only have minimal effect on the estimated reverberation time, unless strong decoupling effects are present in the room.

Room setup, calculation parameters

At this point you should have an idea of the order of size of the reverberation time. To continue the series of calculations you should enter the Room setup and specify the Impulse response length. The Impulse response length should cover at least 2/3 of the reverberation curve; in this case 1500 ms should be sufficient. To learn more about the other parameters available from this page, please press F1.

Global Estimate, a reliable method for estimation of reverberation time

Run Global Estimate and let it run until you are satisfied that the decay curve has become stable, then press the Derive results button. Note the longest reverberation time. The reverberation time differs from the values calculated by Quick Estimate, because the room shape and the position of absorbing material are taken into account. It is important that the Impulse response length in the Room Setup is at least 2/3 of the reverberation time.

Calculating point responses

At this point we are ready to calculate point responses. Three different point response calculations are available:

- Single Point response offering detailed calculation results and auralization options for a selected receiver.
- Multi Point response offering room acoustical parameters for all the receivers defined in the Receiver list at the Source-receiver list.
- Grid response offering a calculated map of room acoustical parameters, if a grid has been specified from the Define grid menu.

Setup a single point response and run it:

- Select source number 1 as the Receiver towards source for each of the jobs 1 – 4. Notice how the blue cross changes into red in the Source Receiver view, indicating that it has been selected for the selected job.
- Select receiver number 1 as the Single Point receiver for job 1, 2, 3 and 4.
- Activate source 1 in job one, source 2 in job two, source 3 in job three and all three sources in job four.

- Click the Run all button in the local toolbar at the right side to run the jobs and the four Single Point response responses will be calculated.

View Single point response

Select job number 1 in the Job List and click the View Single Point response button when the calculations have ended to see the results. You will find six pages available in the Single Point Response window displaying room acoustical parameters, energy curves, reflectograms[3], 3D reflection paths and Binaural Room Impulse Response filters (BRIR). You can view results for each of the four jobs by first selecting the job in the Job List, then clicking the View Single Point response button. To learn more about the results and options available from the Single Point response window please press F1 to consult the online help. You may also select the page of interest and investigate the menu, which then appears at the top menu bar. As a last option play the Binaural Room Impulse Response through headphones using the ALT+I keystroke.
If the Multi option had been checked, you would also be able to view the Multi point response results and if the Grid option had been checked and a receiver grid had been defined, you would be able to view the Grid response results. These topics will be covered below.

Calculate Multi point

Activate the Multi option from the Job list, by checking the Multi option for job 4; then click the Run all or Run Selected Job button.
When the calculation has finished, select job number 4 in the Job list and click the View Multi button to view the Multi point response results. To learn more about the results and options available from this display; press F1. You may also select the page of interest and investigate the dropdown menu, which then appears in the top of the program window.
Note that point responses calculated using the Multi point response option are calculated much faster than Single point responses because no filters are created for auralization use.

Define a receiver grid and calculate grid response

Enter the Define Grid menu and select the two floor surfaces (surface 1001 and surface 1002). Specify the Distance between receivers to 2 (metres) then click the Show grid button.

Note! If the Define Grid button is disabled this is because some process is open, which requires data to be saved. In this case, it is probably the Estimate Reverberation display that needs to be closed. To find this open window, use the Windows menu item on the menu bar. Other displays containing calculation processes may cause the same kind of disabling of miscellaneous options.
Hint! The grid may also be used for easy positioning the point sources and discrete receivers, which are usually defined in the Source-receiver list. To learn how to operate the 3DGrid display, select the display (and the 3DGrid tab in that window) - then select the 3DGrid Parameters dropdown menu.

Calculate grids

Click the Job list button again. Activate the Grid option from the Job list, by checking the Grid option for job 1; then click the Run all or Run Selected Job button.
ODEON will now start calculating the Grid response for this job; this may take a while.
When the calculation is finished, select job number 1 in the Job list and click the View grid button to view the grid results. To learn more about the results and options available from this display; press F1.

Calculate Reflector Coverage

Enter the Define reflector surfaces menu and select the podium-ceiling surface (surface 3001). Then click the Calculate reflector coverage button on the main toolbar to calculate the reflector coverage for the selected surface(s).
Reflector coverage calculates the coverage provided by chosen reflecting surfaces, at the first reflection order only. This is an efficient tool for investigating whether the receiver area is covered by the reflectors or not and if the reflectors are positioned correctly.

Investigate Ray Tracing

The Investigate ray tracing display visualises the ray tracing as it is carried out during any point response calculation. By default its calculation parameters are also set up as the parameters used for the point response calculations (Single Point, Multi and Grid). This display is a very valuable tool for testing new room models, e.g. to detect missing or misplaced surfaces. It may also give an impression of what is happening in the calculations, e.g. the effect of the scattering assigned to the surfaces. Click the Ok button, then click the Single forward button a few times and note the behaviour of the ray tracing.

---

[3] Reflectograms are only used with point sources and will not contain any relevant information for line and surface sources.

3D Billiard

The 3D billiard display is a tool that can be used for investigating or demonstrating effects such as scattering, flutter echoes or coupling effects. A number of billiard balls are emitted from the source and reflected by the surfaces in the room. To speed up the process, set the Dist. per update to a higher value. To visualize a flutter echo, a large Number of billiard balls should be used, e.g. 10000 balls. It's easier to visualize a flutter echo, if rays are only emitted in the relevant plane (XZ, YZ or XY). If the geometry is complicated, it may be hard to see the billiard balls, in that case toggle parts of the geometry off using the T shortcut.

Auralization - Listening to the rooms

At this point you have tried calculation of room acoustical parameters, operating visual display like decay curves, 3D reflection paths, reflectograms etc. Its time to move on, trying the auralization options in ODEON. Two ways of auralization are available in Odeon, a real-time /Streaming convolution which produces one or two channel auralization on the fly (with some latency) and off-line convolution allowing auralization with up to 10 simultaneous channels which may be assigned individual signal, delay and level. The result of the off-line convolution is stored in files for later playback. No matter the way of auralization chosen, auralization in Odeon is based on full filtered Binaural Room Impulse Responses and convolution.

Real-time /streaming convolution

Select job number 1 in the Job List and click the Streaming convolution button. This will open the Streaming convolution dialog. Select the Voice Sabine Short file in the Source signal field (this is an anechoic recording of voice stored in a Windows Wave file residing in the directory set in the Options|Program setup|Auralization|Wave signal file Directory). ODEON will start convolving the selected signal file with the selected Binaural Room Impulse Response (BRIR), in this case the BRIR for job 1. Listen to the output over headphones, to benefit from the binaural quality of the auralization.
The real-time auralization facility allows auralization with two simultaneous channels, e.g. simulating the left as well as the right part of an orchestra using a stereo recording as input signal, also, the input from the soundcard may be use directly for auralization (that is, on most computers this is possible) if the windows play and record controls have been correctly setup. Please press F1 from within the Streaming Convolution display to learn more about operating the options available. Before leaving this example you may want to try the Listen to input signal option.

Listen to a stereo setup

The next example will demonstrate how to set up a classic stereo setup with a receiver position and the two loudspeaker positions. To run this example, you need to have a stereo recording stored on your harddisk as a Windows wave file, in 16 bit resolution and at a sampling rate of 44100 Hz (or to be able to use the soundcard as the input). This file, which do not need to be an anechoic recording for this demonstration, should be residing in the directory set in the Options|Program setup|Auralization|Wave signal file Directory.

Enter the Source-receiver list

Make a copy of the point source, source 1. To do this, select source 1 in the Source list then press the C shortcut to copy - this will open the Point Source Editor with the new source; change the Y-coordinate into -4 and type Left source in the Description field.
Following the scheme above, create a copy of source 4; change the Y-coordinate into 4 metres and type right source in the Description field.

Enter the Job list to carry out calculations

First activate source number 4 in job 5 and source number 5 in job 6, and then select the receiver and point towards which the receiver is oriented. For both jobs you will be sitting in receiver position 1, looking towards source 1; therefore select source 1 as the Receiver towards source point and receiver 1 as the Single point receiver for both jobs. You have now set up job 5, left speaker and job 6, right speaker for calculation of the two binaural impulse responses.

Two channel real time auralization

Select job number 5 in the Job List and click the Streaming convolution button. This will open the Streaming convolution dialog. Select the your stereo input file in the Source signal field. The convolver will begin to convolve an mono version of the input signal with the BRIR which was calculated for job number 5. To obtain 'stereo auralization', select BRIR number 6 as the Secondary BRIR for 2-channel auralization. ODEON will begin convolving the left channel of the input signal through BRIR number 5 and the right channel of the input signal through BRIR number 6 the result being a stereo playback in our simulated room. The process involves four convolutions in parallel, mixing binaural signals, level adjustment and much more, luckily this is

all taken care of by ODEON. To learn more about the Streaming convolution please press F1 from within that display.
Please notice that ODEON allow the combination of BRIR 5 and 6 because the same Point response receiver and Receiver towards source are used in both simulations; after all the same person (receiver) can not sit at more than one place and have more than one head orientation simultaneously.

Offline convolution
The offline convolution more or less repeats what you have tried with the real time convolver, the difference being greater flexibility, more channels allowed, individual adjustment of each channel is allowed, plus that auralization results are stored in wave files which may be used on home pages, CD-ROM's, Power Point presentations etc.
Click the Toggle button to get to the auralization display. This display is divided into a left and a right part. In the left display, mono signals are convolved with Binaural Room Impulse Responses (BRIR's, which have been calculated as part of the Single Point responses), this process may be compared to a binaural recording of a mono signal played through simulated source(s) in the room. The right part of the auralization display (two tables) is a mixer allowing convolved results to be combined into one (wave file) allowing multi channel simulations, e.g. stereo setups, singer versus orchestra etc. The Offline auralization offers greater flexibility than the real-time auralization, allowing full control over which signal to pass through which of the 10 channels available and assigning individual level and delay to each channel. If for some reason you need the auralization output as a wave file it is also the offline auralization, which should be used.

Single channel simulation
First try to create a one-channel simulation of a person speaking from source position 1. In the auralization display, select the Conv. no.1 row and select the Voice Sabine Short file in the signal file field (this is an anechoic recording of voice stored in a Windows Wave file residing in the directory set in the Options|Program setup|Auralization|Wave signal file Directory). To play the selected signal file, make sure this cell is selected; then press the Alt+S shortcut (or the Play wave button).
Adjust the Rec. Lev (recording level) to -30 dB. Then arrow right to the Job no. column and select Job no. 1 from the dropdown list. Once you have exited the Job no. cell the corresponding 3D Source Receiver view is updated to show active sources etc. Click the Run All button to convolve the signal with the BRIR. If other calculations, e.g. point response calculations have to be carried out before the convolution is allowed, ODEON will manage this automatically.

Play auralization file through headphones
Once the calculations have been carried out, click the Play wave result button and listen to the result through headphones. If you have selected the Signal file column in the Convolve BRIR and Signal file table, the (anechoic) input file is played. If any other column in this table is selected, the convolved result file is played.

Convolving BRIR's with signals and mixing signals
In the following we will assume that you have a stereo recording called MyStereoRecording.wav stored on your computers harddisk in a 16-bit resolution, sampled at 44100 Hz. Toggle to the Auralization display (ALT+T).
First step is to set up two mono playbacks, one playing the left channel of the stereo signal, another playing the right channel.

Left speaker playing left signal
In Convolve BRIR and Signal file table, setup Conv. no. 2 (Row number 2):
- Select a stereo signal file e.g. MyStereoSignal in the Signal file Column.
- Select channel 1 to select the left channel of the signal[4] in the channel column.
- Select job number 5 in the Job no. column to simulate the left channel being played through the left loudspeaker.
- Adjust the Rec. Lev. (Recording level) to -40 dB (or use the Overall recording level available in the Auralization setup, this setting is effective on all convolutions)

Right speaker playing right signal
In Convolve BRIR and Signal file table, setup Conv. no. 3 (Row number 3):
- Select the same signal file as above e.g. MyStereoSignal in the Signal file Column
- Select channel 2 to select the right channel of the signal in the channel column
- Select job number 6 in the Job no. column to simulate the right channel being played through the right loudspeaker.
- Adjust the Rec. Lev. (Recording level) to -40 dB.

[4] In a stereo wave file, the first channel is always the left channel and the second channel is always the right channel.

Mixing signals

So far we have setup two mono simulations, one playing the left channel and another one playing the right channel of a stereo signal. To finish the stereo setup we need to mix the two binaural signals together. The binaural mixer is in the right part of the auralization display. If you are using a low-resolution display not all of the display may be visible, however you can drag the borders between (and inside) the tables using the mouse.

Select the Mix. No. 1 in the Mix Convolved wave results into one wave file table. The rightmost table displays the binaural results that are combined in this Mix. Select row 1 in the table and select Conv. No. 2 (the simulation of the playback of the left signal), then select Conv. No. 3 in row 2 (the simulation of the playback of the right signal).

Notice that you may also apply attenuation and a delay to each of the signals in the rightmost table of the mixer. The attenuation corresponds to the attenuation knob on a mixer. The delay is used for delaying the appropriate Convolved signal (and should not be confused with a source delay). An example where the delay feature could be useful is a simulation of an underground station where one signal is the train noise and another is the loudspeaker announcement, the signals are not necessary of the same length and you may want to delay one of the signals. You may also use the delay if you wish to make noise sources which are playing the same noise signals less correlated, e.g. if ten sources are playing the same 'cocktail party' noise (ten Single Point response Jobs and ten convolutions), apply delays like 0, 2, 3, 5, 7, 11, 13, 17,19 seconds (or another time unit). You may mix up to ten convolutions together.

Calculating BRIR's, left signal, right signal and the stereo signal

The stereo setup has now been completed and you may start calculations (Run All). Depending on the source gains you have chosen, you may experience overload or underload, in this case you should adjust the Recording Level and /or Mixer level and recalculate. These levels correspond to the levels on a tape-recorder and on a mixer and the problems concerning overload and underload are the same. If levels are too low you will get a poor dynamic range and if too high you will experience clipping. The Out Lev. in the rightmost columns of the tables should not exceed 0 dB, on the other hand if the output level is say -30 to -50 dB a very poor dynamic range is obtained.

 **Playing results**

When calculations have finished you may play the calculated binaural simulations. Select the relevant table, row and column, then click the play button (or the Alt+S short-cut):

- To play the input signal, select the Signal file column in the Convolve BRIR and Signal file table.
- To play the mono signals convolved with a BRIR select any other column in the Convolve BRIR and Signal file table.
- To play the mixed results, select the relevant row in the mix table.
- Finally to play the individual components in a selected mix, select the relevant row in the rightmost table (mixer level adjustments are not taken into account). These signals are just a repetition of the convolved results from the rightmost table.

Getting further

To familiarise further with ODEON you should try to change some of the materials, sources etc. in the room and make new calculations. A suggestion is to try changing the scattering coefficient on surface 2004 Rear wall behind audience from 0.7 to 0.1 and listen to the change in sound quality (echo problems); Create a copy of the room using the File|Copy files option - then make the changes to this room model. In this way you will have results from both of the rooms present for comparisons.

Precalculated Rooms

At this point you have tried the basic functions in ODEON and may want the view results for more realistic rooms. In the room directory you will find pre-calculated results in the room Elmia.Sur, which is a model of a multi purpose hall in Sweden. For the combined edition of ODEON, the room Studstrup.Sur is also available; this is a model of a turbine hall at a power plant. If you wish to try the auralization options in these rooms you should turn on the Auralization Setup|Create Binaural impulse response file option and make the necessary calculations in the Job list.

2.1.1 Summary of the calculation methods

Global Estimation of reverberation time

There are two calculation methods for the calculation of global reverberation time built into ODEON. The global estimated reverberation times are estimations for the complete room with one selected source position.

 **Quick estimate** is the fast method, which is found in the Material List. This method is based on the Sabine and Eyring formulas and as such assumes diffuse field conditions. Diffuse field cannot be assumed if:

- the room absorption is unevenly distributed.
- the room contains de-coupling effects, e.g connected corridors or niches.

Thus the results given by Quick Estimate should not be considered the final result. Even so the method is useful in the initial work on assigning reasonable materials to the surfaces in the room.

Global estimate is a more precise method, which doesn't make any assumptions about diffuse field conditions and as such, it is a more reliable method for estimation of global reverberation time.

- For workrooms where all absorption is often situated in the ceiling region and sources are situated in the floor region the RT predicted by Global Estimate will typically be longer than the values predicted by Quick Estimate, a factor two is not unlikely if walls are basically smooth.
- In auditoriums the opposite is the case, because the dominant absorption area (the audience) is close to the source.

In any case the RT's predicted by Global Estimate is the most reliable.

Point Response calculations

The Point response calculations estimate not only RT, but also room acoustic parameters like Clarity, Deutlichkeit, SPL, SPL_A, STI and LF_{80} (see chapter 7). The calculated results can be thought of as a simulated measurement. Calculated results relates to:

- a number of active sources
- one receiver position
- orientation of the receiver (for LF_{80}, $LLSPL_A$ and auralization)

The orientation of the receiver(s) in a particular job is set in the Job list by selecting a point source through which the receivers are looking. There are three kinds of point response calculations; the Single Point Response, the Multi Point Response and the Grid Response.

Single Point Response is calculated for a selected receiver position, which must be defined in the Source-Receiver list. The Single Point Response is the most detailed calculation method allowing:

- Prediction of room acoustical parameters (including stage parameters)
- Display of predicted Decay curves
- Tracking of individual reflections in a reflectogram and display and tracing the reflection(s) in 3D displays of the room e.g. for tracking down echo problems.
- Auralization (see sections **Error! Reference source not found.**)

Multi point response calculates room acoustical parameters for all the discrete receiver positions defined in the Source-receiver list.

Grid Response calculates room acoustical parameters for a mapped receiver area. The surfaces over which grids should be calculated are selected in the Define Grid display.

The Auralization features are available from within the Job list. Auralization is based upon binaural impulse responses (BRIR's), which may be calculated as a part of the Single Point Response. Fast and easy to use auralization is provided by the Steaming convolution facility – to learn more about this facility use the online help from within the JobList. If greater flexibility is needed a separate auralization display appears when using the toggle button (Alt+T) from within the JobList.

Offline convolution for flexible Auralization

In the left part of the auralization display mono signals are convolved with BRIR's - in the right part of the auralization display, such convolved signals may be mixed together in order to create multi channel simulations. The Auralization results are always two channel signals (binaural), which should be listened to through headphones.

The mono input signal is selected in terms of a signal file and a channel in that file. In a stereo signal file, channel 1 is the left channel signal, channel 2 is the right channel signal and average is the average signal of the channels included in the file. A signal file is typically 1- channel (mono) or 2-channel (stereo), but may in principle contain many channels.
The BRIR is selected in the Job no column and refers to the Single Point Response with that job number. Once the BRIR has been selected, the point through which the receiver is oriented is displayed in

the corresponding 3D display and the used receiver position is displayed in the table in the auralization display.
The recording level may have to be adjusted in order to get a good dynamic range or on the other hand to avoid overload. The output level achieved when convolution has been carried out is displayed in the rightmost column and should never exceed 0 dB. The recording level corresponds to the recording level on a tape-recorder. If you wish to compare different simulations you should use the same recording level.

Creating multi channel auralization is by nature a little complicated and you should get familiar with one-channel simulations before using this feature (the mixer).
Multi channel simulations can be created using the mixer in the auralization display (the two rightmost tables). The mixer allows you to mix together up to ten (one-channel) simulations from the Convolve BRIR and Signal file table. The simulations can only be mixed together if they:

- use the same receiver position (Point Response Receiver)
- use the same orientation (receiver towards source)

To check this, scroll through the Convolve BRIR and Signal file table and view the receiver column (Rec.) and the receiver towards source point, which is displayed as a red cross in the corresponding 3D display.

Other facilities in ODEON
Apart from the features demonstrated in the above tour, ODEON also contains facilities for:

- Deleting calculation or result files (available from the Files menu item).
- Archiving project files in one single compressed /zipped file, for efficient and safe storage or for easy posting by e-mail (available from the Files menu item).
- Tools for detecting errors in a new model, e.g. warped or overlapping surfaces (available from the Toolbar dropdown menu).
- Set-up for printouts (available from the Options|Program Setup menu item).
- Export of calculated data in ASCII /text format for post processing.

2.2 Short guided tour - Industrial edition

Run the ODEON application

You will find the ODEON program at the Windows Menu Start|Program files|Odw4_0|Odw. Execute the program and begin the tour.

Open a room model to work on

Select the Open a room model button to select a room. The room files containing the geometries for ODEON carry the extension .Par or (.Sur for compatibility with previous versions of ODEON) and is plain ASCII /text files following the format outlined in chapter 3. For this guided tour select the room model named WorkPlace.Sur.

3DView

Have a look at the room. Whenever ODEON loads a room, it is displayed in a 3DView. This allows you to investigate the geometry and check it for errors, etc. Several facilities are available in the 3DView; e.g. rotation, zooming, highlighting of selected surfaces and corner numbers etc. Press F1 to get overview of the facilities and their use.

Having assigned a room, this is a good time to get familiar with the MDI concept (Multiple Documents Interface). At this point the title bar of the 3DView will be blue (or some other colour) indicating that this is the active window. Being the active window, the 3DView menu item is added to the menu bar next to the Toolbar dropdown menu. You can operate the functions of the window using this menu or the shortcut keys displayed in the menu.

Define sources and receivers

Before any calculation can be carried out in ODEON, one or more sources will have to be defined. Of course a receiver will also have to be defined in order to calculate a point response.
In this guided tour we shall define a point, a line and a surface source. Finally we define a receiver.

Click the Source-receiver list button at the toolbar to open the Source-receiver list from which sources and discrete receivers are defined. If the Source-receiver list is already open, but hidden behind other windows, etc., clicking this button will rearrange the windows as needed.

Define a point source

Click the New point source button to open the point source editor. Enter the values x = 0.5 (metres), y = 1.6 (metres), z = 1.65 (metres) and Azimuth = 90° [5][6]. Finally select Directivity file to CARDIOID. If you are not sure of the position of the source, you can select the 3D Edit Source-Receiver display. If you do so, you should notice how the menu item 3D Edit Source-Receiver appears on the toolbar dropdown menu, when this window becomes active. The 3D Edit Source-receiver menu will allow you to operate the 3D display.
Finally set the Overall gain to 80 dB. To save the new source just close the Point source Editor and confirm.

Hint; Use the Tab or Shift+Tab shortcuts to move between data fields.

Define a line source

Click the New line source button to open the Line source editor. Enter the values x = 0.5 (metres), y = -1 (metres), z = 1.65 (metres), Length = 2 (metres) and Azimuth = 90°.
Finally set the Overall gain to 82 dB. To save the new source just close the Line source Editor and confirm.

Define a surface source

Click the New surface source button to open the surface source editor. Select surface 8 for this source. Finally set the Overall gain to 75 dB. To save the new source just close the Surface source Editor and confirm.

Define receivers

Click the New receiver button to open the Receiver editor. Enter the values x = 1.5 (metres), y = -0.5 (metres) and z = 1.65 (metres). To save the new source just close the Receiver Editor and confirm.

---

[5] Hint; Use the Tab or Shift+Tab keys to move between fields.
[6] Depending on the language selected on your computer '.' Or ',' may be used as decimal point.

Define other receivers at:

(x, y, z) = (12; 3; 2.2)
(x, y, z) = (8; 7; 1.5)
(x, y, z) = (21; 1; 3.6)

We will get back to the receivers and the activated sources under the point: Calculating Point Responses.

Assign material properties

Open the Materials List and see how to operate in the Materials menu.

Assign the following material data to the surfaces in the model:

Surface number	**1**	**2**	**3**	**-4 4**	**6**	**7**	**8**	**9**	**-10 10**
Material	702	1008	100	100	702	100	30	30	30
Scatter	0.1	0.5	0.7	0.3	0.3	0.1	0.5	0.5	0.5
Transparency	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Hit the F1 shortcut to learn more about scattering and transparency coefficients. The transparency coefficient allows you to easily model a reflector panel or installations build from many small surfaces, simply by modelling one surface and applying an appropriate transparency coefficient. Notice high scattering coefficients are used on the floor and sidewalls in order to model machinery and beams.

Quick Estimate, fast estimation of Reverberation Time

From within the Materials list run the Quick Estimate to get an idea of the reverberation time. Note the longest reverberation time. This calculation is very useful while assigning materials for evaluating different materials and their impact on the reverberation time. Before leaving the Quick Estimate you may want to try this out by choosing different materials. It is possible to select among the defined sources. However, the source position will only have minimal effect on the global estimated reverberation time, unless strong coupling is present in the room.

Room setup, calculation parameters

At this point you should have an idea of the order of size of the reverberation time. To continue the series of calculations you should enter the Room setup and specify the Impulse response length. The Impulse response length should cover at least 2/3 of the reverberation curve, in this case 800 ms should be sufficient. To learn more about the other parameters available from this page, use F1 shortcut.

Global Estimate, reliable method for estimation of reverberation time

Run Global Estimate and let it run until you are satisfied that the decay curve has become stable, then press the Derive results button. Note the longest reverberation time. The reverberation time differs from the values calculated by Quick Estimate, because the room shape and the position of absorbing material are taken into account. It is important that the Impulse response length in the Room Setup is at lease 2/3 of the reverberation time.

Calculating point responses

At this point we are ready to calculate point responses. Two different point response calculations are available in the industrial edition:

- Multi Point offering room acoustical parameters for all the receivers defined in the Receiver List at the Source Receiver List.
- Grid offering a calculated map of room acoustical parameters, if a grid has been specified from the Define grid menu.

Setup a Multi point response and run it:

- Activate source 1 in job one, source 2 in job two, source 3 in job three and all three sources in job four.
- Turn on the Multi option for the jobs 1 to 4 in order to calculate the point responses for the five receivers you have defined. Notice how the active sources are displayed in the 3D Source-receiver display as you scroll through the Job list.

Viewing results

When the calculation has finished, select job number 4 in the Job list and click the View Multi button to view the Multi point response results. To learn more about the results and options available in this window; press F1. You may also select the page of interest and investigate the dropdown menu, which then appears in the top of the program window. You can view the Multi point response results for each of the four jobs by first selecting the job in the Job List, then clicking the View Multi Point response button.

If the Grid option had been checked and a receiver grid had been defined, you would be able to View Grid Response results as well. This topic will be covered below.

Define a receiver grid and calculate grid response

Enter the Define Grid menu and select the floor surface (surface 3). Specify the Distance between receivers to 3 (metres) then click the Show Grid button.

Note! If the Define Grid button is disabled this is because some process is open, which requires data to be saved. In this case, it is probably the Estimate Reverberation display that needs to be closed. To find this open window, use the Windows menu item on the menu bar. Other displays containing calculation processes may cause the same kind of disabling of miscellaneous options.

Calculate grids

Click the Job list button again. Activate Grid option from the Job List, check the Grid option for job 1 - 4 and click the Run All button.

ODEON will now start calculating the grid response for the four jobs; this may take a while.

When the calculation is finished, select job number 1 in the Job List and click the View Grid Response button to view the grid results. To learn more about the results and options available from this display; use the F1 shortcut.

Investigate Ray Tracing

Investigate ray tracing visualises the ray tracing as it is carried out during any point response calculation. By default its calculation parameters are also set up as the parameters used for the point response calculations (Single Point, Multi and Grid). This display is a very valuable tool for debugging of new models, e.g. to detect' missing or misplaced surfaces. It may also give an impression of what is happening in the calculations, e.g. the effect of the scattering assigned to the surfaces. Click the Ok button, then click the Single forward button a few times and note the behaviour of the ray tracing.

3D Billiard

The 3D billiard display is a tool that can be used for investigating or demonstrating effects such as scattering, flutter echoes or coupling effects. A number of billiard balls are emitted from the source and reflected by the surfaces in the room. To speed up the process, set the Dist. per update to a higher value. To visualize a flutter echo, a large Number of billiard balls should be used, e.g. 10000 rays. It's easier to visualize a flutter echo, if rays are only emitted in the relevant plane (XZ, YZ or XY). If the geometry is complicated, it may be hard to see the billiard balls, in that case toggle parts of the geometry off using the T shortcut.

Precalculated Rooms

At this point you have tried the basic functions in program and may want the view results for more realistic rooms. In the room directory you will find the precalculated results in the room Elmia.Sur, which is a model of a multi purpose hall in Sweden. The room Studstrup.Sur is a model of a turbine hall at a power plant.

Other facilities in ODEON

Apart from the features, which have been demonstrated in the above tour, ODEON also contains facilities for:

- Copying the project files generated by ODEON (available from the Files menu item).
- Deleting calculation or result files (available from the Files menu item).
- Archiving project files in one single compressed /zipped file, for efficient and safe storage or for easy posting by e-mail (available from the Files menu item).
- Tools for detecting errors in a new model, e.g. warped or overlapping surfaces (available from the Toolbar dropdown menu).
- Setup for print-outs and graphics (available from the Options|Program Setup menu item).

3 Modelling rooms

Creating new room models is probably the most time consuming task in room acoustical modelling. However good modelling practice will greatly reduce the time used for modelling and remodelling rooms.

In order to study a room in ODEON, a file containing the description of the room's geometry will have to be created. All subsequent derivative files and result files are created and managed by ODEON.
The file containing the room model must be written as an ASCII text file, having the file extension .Par (the 'old' ODEON .Sur file format is also allowed though not described in this manual). You can choose to create the geometry file either by typing the model data directly into a text file in the supplied text editor OdwEdit, using the format described in section 3.2 or with the aid of a CAD program (e.g. AutoCAD 2000) which is capable of creating 3D surface models and exporting these as DXF files as described in section 3.3.

No matter which approach you choose for modelling, always check the validity of the models. The room model must form a (almost) closed enclosure. It should also be (almost) free from warped (twisted), duplicate or overlapping surfaces. ODEON has several tools for checking models for such problems. The tools are presented in section 3.4. It is suggested that you always use these tools when working on models of some complexity.

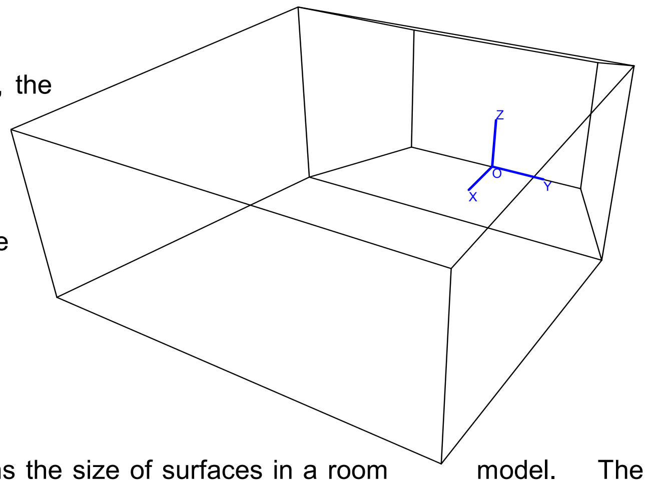
3.1 Guidelines on room modelling

Whether you choose to model your rooms by typing your rooms directly into a text file or using a CAD program, there are considerations that are common to either case. Some guidelines of general nature are given below.

3.1.1 Default coordinate system

To make it as easy as possible to operate ODEON, the following orientation of room geometries should be applied (using a concert hall as the example):

- X-axis pointing towards the audience
- Y-axis pointing to the right as seen from the audience
- Z-axis pointing upwards



3.1.2 Recommended size of a surface

The most important theoretical consideration concerns the size of surfaces in a room model. The laws of geometrical acoustics are such that for the purposes of calculating how much energy is reflected; all surfaces are considered to be infinitely large in comparison to the wavelength. In ODEON approximate diffraction damping is introduced in order to take into account the limited size of the surfaces, still it is an approximation so to keep the calculated results reasonably accurate, surfaces should be kept "reasonably large". In ODEON, this restriction only applies to early reflections from point sources which are calculated using the image source method, late reverberate reflections are not constrained in this way. The implication for model-building is that important surfaces should be as large as possible. Modelling a lot of small surfaces to achieve high geometrical fidelity is likely to produce worse rather than better results. It is difficult to put concrete limits on the size of surfaces, which should be used, and there will almost always be a need for small surfaces to fill in awkward corners of the geometry. Such small surfaces need not invalidate the results of calculations. However, significant numbers of small surfaces in positions likely to be important for early reflection sequences should be avoided. In this context, "small" may also refer to a dimension rather than an area; thus a very narrow surface is acoustically "small". A typical model of a concert hall can typically be modelled with a surface count of say 100 to 1000 surfaces. A model of a concert hall containing more than 1000 surfaces is likely to be contain too many small surfaces so at its best, producing models with such a degree of detail is likely to be a waist of time.

3.1.3 Level of detail in room model

This subject is closely related to that of surface size, but is of a more general nature. Computer models like ODEON are by their nature only rough tools. It is therefore senseless (and may be counterproductive) to expend enormous effort modelling rooms to high precision. What is important is to mimic the character of the room's geometry. It is only important to place planes accurately and model geometry closely in those parts of the room, which participate strongly in early reflections (and here, the considerations above come into play); everywhere, quite large inaccuracies and simplifications may be tolerated.

3.1.4 Curved surfaces

All surfaces in ODEON must be (almost) plane; so curved surfaces have to be approximated by dividing them into plane sections. The question of how finely to subdivide depends on the type of curved surface and how important the surface is.

Convex curves naturally disperse sound energy, so if the surface is in an exposed position (e.g. the end of a balcony near the stage), one should avoid for example simply replacing a quarter circle with a single plane at 45, which might then act like a reflector.

Concave curves naturally focus sound energy, and since focussing is a fault we wish to model, we must try to arrange that it be preserved. However, this does not mean that a large number of subdivisions are the solution. Using many surfaces in the model will:

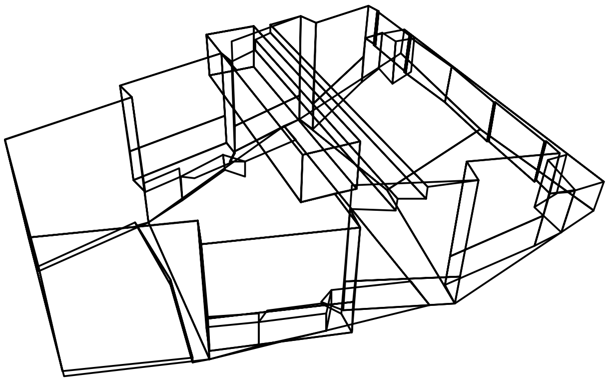
- Make the model visually complex, and increase the probability of errors in the model, typically small leaks becomes a problem.
- Not combine with the image source theory used for the early reflections (point sources).

Subdivisions about every 10° to 30° will probably be adequate to reproduce focussing trends, without excessive numbers of surfaces.

3.1.5 What to model?

How to model an audience area?

Modelling each step between the rows in an audience area is not recommended, the audience area can be simplified a lot without compromising the quality of the results – in fact using one of the suggestions methods below is likely to produce better results.

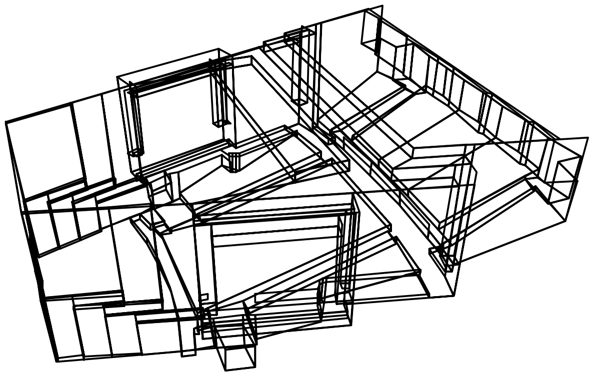


Audience A) Modelling the floor surfaces

a) Define the floor area below the audience.
b) Assign appropriate 'absorption material' e.g. Odeon material 901,902,903 or 904.
c) Assign a high scattering coefficient of 0.7 to this area.
d) Place the receivers some 1.2 metres above the floor.

Audience B) Modelling the audience as boxes

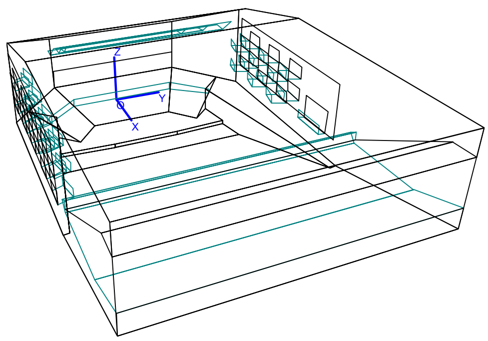
Model the audience area as 'audience boxes' with a height of approximately 0.8 metres above the audience floor.
b) Assign appropriate 'absorption material' e.g. Odeon material 901,902,903 or 904.
c) Assign a high scattering coefficient of 0.7 to the surfaces of the 'audience box'.
d) Position the receivers some 0.4 metres above the modelled 'audience box'.



Experiences from tests with first approach are positive and it is far the easiest to model. The 'Elmia' hall is an example of this. A potential drawback of modelling the audience area as a box is that it removes volume from the room, which is may be a problem in rooms with low ceiling height /volume.

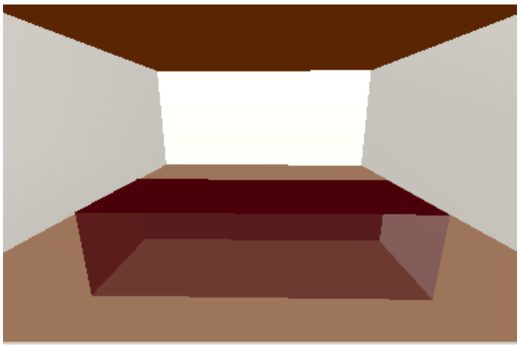
How to model the podium on stage?

Same guideline as for the audience area goes here. Rather than modelling each step of the podium on stage, the podium can be simplified into a few sloped surfaces.



Should furniture such as tables, chairs and shelves be included in a model of an office?
If a table plate is close to a source or receiver point, then it I likely to produce a strong reflection at the receiver, so if this is the case then it should be included. Furniture such as shelves and screens in large office environments, which subdivides the room – breaking up long reflection paths and introducing extra absorption and scattering should neither be omitted.
Furniture at more distant locations in the room, which does not produce any strong early reflections to the receiver can be greatly simplified or even omitted from the model as long as the a extra absorption and scattering produced by that furniture is somehow included on other surfaces in the same regions of the room.

How to model a table with chairs?
An easy way to model a table with chairs around it is to model a box, making its side surfaces semi transparent by setting the transparency coefficients to values greater than zero (e.g. 0.5) in the Materials list inside Odeon.
Recommended scattering coefficients for furniture are between 0.6 and 0.8.
It may be acceptable to model the geometry in more detail, but the above method works well and makes the modelling process much faster.



Orientation of surfaces – does tilt of a surface have any significance on room acoustics?
Small changes to the orientation of surfaces can indeed cause dramatic changes. Making dominant surfaces slightly off-angle can cause extra scattering in the room almost as if extra scattering had been assigned to the surfaces in the room. A classical example on this is the box shaped room where a flutter echo can be removed by changing the angle of a surface by a degree or two.

3.2 Modelling rooms by typing it into a text file

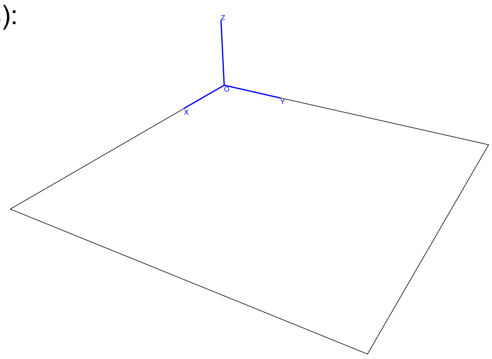
3.2.1 The ODEON .Par modelling format /language

Geometry models can be made using the parametric modelling language, which is built-in to ODEON. The model data are typed into an text file given the file extension .Par using the modelling language described below. You may use the supplied editor OdwEdit to create and edit your text files. The ODEON modelling format is not case sensitive, so upper and lower case letters can be used as desired.

A simple modelling example

At its simplest (but not fastest), a floor with the dimensions 4 x 4 metres can be defined as follows (using the reserved keywords ***Pt*** and Surf in order to define points and surfaces):

```
FloorSurface.Par
###
Pt      1     0     0     0
Pt      2     4     0     0
Pt      3     4     4     0
Pt      4     0     4     0
Surf    1     floor
  1     2     3     4
###
```



One may chose to model the room point-by-point and surface-by-surface as in the example above, however for many geometries it will be an advantage to use parameters to describe basic dimensions in the rooms and to use high level statements to describe multiple points and surfaces in a fast and flexible way. Before starting your first large modelling project it is a very good idea to read through chapter 3 or at least skim it - it will pay off in the end.
Another way to learn about the modelling language is to study the examples, which are installed in the \Odw..\rooms\manual examples directory along with the Odeon program – open the room(s) in Odeon, then click the Open the Odeon Editor icon on the toolbar in order to study the .

Components in the modelling format

The basic function of the modelling format is to allow modelling of surfaces in room geometries. The surfaces can be modelled point-by-point, surface-by-surface, however it is also possible to make use of symmetry and to create repeated features in a room such as columns, using programmatically loops, finally it is possible to use hybrid functions, which creates points as well as surfaces in order to create shapes like boxes etc.

Constants, variables and counters

Constants and variables can be defined and used in the file format. It is a good habit to use constants whenever a value is used more than a few times in a file, this reduces typing errors and it also makes it easier to make general changes to a geometry such as changing the height of a room.

Mathematical expressions

Mathematical expressions can be used to express any real or integer number in the file format e.g. coordinates, constants, variables, counters, point numbers, surface numbers etc. If you use a value that is not an integer value to describe a point- or surface number, which is an integer, then that value will be rounded to the nearest integer value.

You may describe coordinates using mathematical expression like:

Length*Sin(PI/4)

where ***Length*** is a user-defined constant or variable. Mathematical expressions may not contain any SPACE or TAB (tabulation) characters. To get a complete overview of the mathematical functions available, please refer to appendix A.

Points

A point is made up from an unique point number and its X,Y and Z coordinates, Use the ***Pt***, ***MPt*** and ***CountPt*** statements to define points. Points can also be defined implicitly, using one of the hybrid statements.

Surfaces

A surface is made up from a unique number, an optional descriptive text and a number of points connected to one another. To define surfaces use the ***Surf***, ***MSurf, CountSurf, ElevSurf*** and ***RevSurf*** statements. Surfaces may also be defined implicitly using the hybrid statements.

Hybrid statements
Hybrid statements are; the ***Box***, ***Cylinder***, ***Cylinder2***, ***Cone***, ***Dome, Dome2*** and ***ElevSurf*** statements. The hybrid statements create the points and surfaces needed to model the specified shape. The points and surfaces created must always have unique numbers.

Coordinate manipulation functions
A set of functions for coordinate manipulation (and surfaces made up from coordinates) is included. This includes rotation around the various axes, scaling and translation. These functions are needed in order to insert shapes defined by the hybrid statements in the geometry with the correct position and orientation.

Comments and empty lines
Lines containing comments, and empty lines, may be inserted anywhere in the file, as long as they do not come between data items, which should occur on one line. Comment lines must begin with a colon (:), a semicolon (;), a slash (/) or an asterisk (*). The semicolon can also terminate a non-comment line, allowing a non-comment line to be terminated with a comment.

A series of comment lines are started with a ‘{‘ and ended with a ‘}’, both as the first sign on a line.

Reserved keywords, predefined counters and constants
The following keywords are reserved by ODEON and has a special meaning in the parametric modelling language:

Constant and variable statements
Const, ***Var***

Point statements
Pt, ***MPt***, ***CountPt***

Point lists statements
Plist0 – Plist9
ResetPList0 – ResetPList9
PlistA, PListB

Surface statements
Surf, ***MSurf***, ***RevSurf***, ***CountSurf, ElevSurf***

Hybrid statements
Box, Cylinder, Cylinder2, Cone, Dome, Dome2

Loop statements
For..End,

Transformation statements
Mreset, ***MPop***, ***MScale***, ***MTranslate***, ***MRotateX***, ***MrotateY***, ***MrotateZ***
(and for compatibility with earlier releases of ODEON: ***Scale***, ***UCS)***

Predefined constants
PI = 3,14159265358979312

Predefined variables
NumbOffSet, ***ONVert***

Predefined Counters
PtCounter

Coordinate system definition statements
Unit, ***CoordSys***

Defining constants

Constants must follow the syntax:

```
Const<Name><Value>
```

where value is a mathematical expression, which may be based on numbers or constants and variables that has already been defined.

Example 1:

```
Const     CeilingHeight     3.4
```

Example 2:

```
Const FloorLevel             1
Const CeilingHeight FloorLevel+3
```

Example 3:

```
Const     FloorHeight     1
Const     Length          6
Const     CeilingHeight   FloorLevel+Length*TanD(30)
```

Defining and reassigning variables

The definition of variables must follow the syntax:

```
Var <Name><OptionalValue>
```

Example 1; defining the variable ***FloorLevel***:

```
Var     FloorLevel
```

Example 2; defining the variable ***FloorLevel*** and assigning the initial value ***0***:

```
Var     FloorLevel     0
```

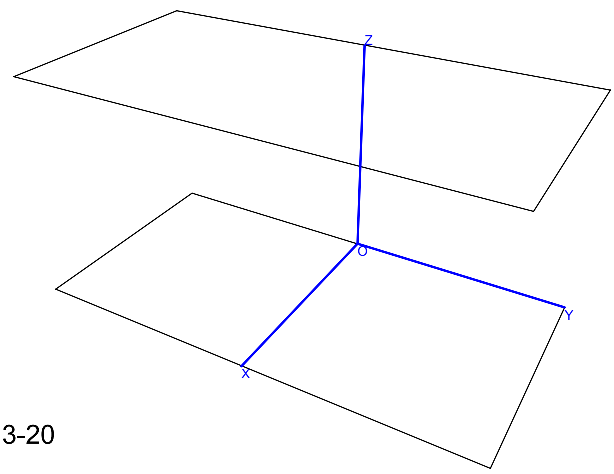
Example 3; reassigning a variable /adding 1 metre to the ***FloorLevel***:

```
FloorLevel FloorLevel+1
```

Remark: The predefined variable ***NumbOffSet*** may be used like any other variable, but has a special meaning because is offsets point and surface numbering. This variable is useful if copying a part of a geometry from another geometry file, it is also useful in connection with the ***for..end*** statements.
'***Auto***' can also be assigned to ***NumbOffSet***, in doing so Odeon will automatically increment the value of ***NumbOffSet*** to be greater than any point and surface number previously defined. This has the advantage that repeated point and surface numbers can easily be avoided without having to keep track on the numbers used - the drawback is that slight changes in the geometry file may change numbers on many subsequent surfaces, ruining the relationship between surface numbers and the material assigned to that surface inside the Odeon program.

Example on the use of ***NumbOffSet***; creating surface 101 containing the points 101 to 104 and surface 201 containing the points 201 to 204:

```
NumbOffSet.Par
###
NumbOffSet 100
Pt        1        0        -1        0
Pt        2        0         1        0
Pt        3        1         1        0
Pt        4        1        -1        0

Surf      1        A surface
   1      2        3         4

NumbOffSet         NumbOffSet+100
Pt        1        0        -1        1
Pt        2        0         1        1
Pt        3        1         1        1
```



```
Pt      4     1     -1    1
Surf    1     Another surface
  1     2     3     4
###
```

Example creating point 1–4 and surface 1, setting ***NumbOffSet*** to ***Auto***, then creating Point 5-8 and surface 5:

```
###
Pt      1     0     -1    0
Pt      2     0     1     0
Pt      3     1     1     0
Pt      4     1     -1    0

Surf    1     A surface
  1     2     3     4

NumbOffSet    Auto
Pt      1     0     -1    1
Pt      2     0     1     1
Pt      3     1     1     1
Pt      4     1     -1    1
Surf    1     Another surface
  1     2     3     4
###
```

Defining a point using the Pt statement

Use the ***Pt*** statement to define a single point. The syntax must be as follows:

Pt <Point Number><XMathExpression><YMathExpression><ZMathExpression>

Example defining point number 100 in (x,y,z) = (1,1,1):

```
Pt      100   1     1     1
```

Hint! Point number and coordinates can be written using mathematical expressions, allowing greater flexibility and reusability.

Parametric modelling - defining multiple points

Use the ***MPt*** statement to define a series of points, which is typically used in connection with the ***ElevSurf*** statement. The syntax must be as follows:

MPt <Number> <NumberOfPoints>
<XMathExpression1><YMathExpression1><ZMathExpression1>
<XMathExpression2><YMathExpression2><ZMathExpression2>
....NumberOfPoints lines each defining a point in the multi point sequence, should follow the MPt statement.

<Number>

A unique number from 1 to 2.147.483647 for identification of the first point in that multi point sequence.

<NumberOfPoints>

The number of points defined by this multipoint statement - if the number is 3, then 3 lines should follow, each describing the coordinates of a point.

Example 1;defining point number 100 in (x,y,z) = (1,1,1) and point number 101 in (x,y,z) = (2,2,2)

```
MPt   100 2
  1.0   1.0   1.0
  2.0   2.0   2.0
```

As a special option for multi points, it is possible to repeat a coordinate used in the previous point of that multipoint sequence or to repeat the coordinate while adding or subtracting a value from that point:

Example 2; defining point number 100 in (x,y,z) = (1,1,1) and point number 101 in (x,y,z) = (1,2,0)

```
MPt   100 2
  1   1   1
  =   =+1  =-1
```

Defining a series of points using the CountPt statement

The ***CountPt*** statement must follow the syntax:

CountPt<FirstPointNo><MaxCount><XMathExpression><YMathExpression><ZMathExpression>

Use the ***CountPt*** statement to define a series of points using a counter. This statement makes use of the predefined counter ***PtCounter***, which will run from ***0*** to ***MaxCount-1***, producing the points with the numbers ***FirstPointNo*** to ***FirstPointNo+MaxCount-1***. Use the ***PtCounter*** in the expression of the x,y and z coordinates to create the desired differences between the 'count points'.

Example; defining 7 points on a circle with a radius of 10 at Z=0 metres:

```
CountPt   100   6+1   10*CosD((PtCounter)*360/6)   10*SinD((PtCounter)*360/6)   0
```

Note! First and last point in this series of 'count points' are equal (redundant). This will typically be desirable when using the ***CountPt*** statement along with ***RevSurf*** statement.

Defining a single surface using the Surf statement

A ***Surf*** surface is divided into two lines and must follow the syntax:

Surf <SurfaceNumber><Optional Description>
<ListOfPointNumbers>

The ***Surf*** statement is used to define a single surface (in some situations with symmetry, two surfaces). The ***Surf*** statement is constructed from two lines, one identifying the surface by a number and an optional name and another with a list of corner numbers.

<SurfaceNumber>

A unique number from 1 to 2.147.483647 for identification of the surface. Using the same number, but with negative sign defines the surface and its mirrored counter part in the XZ-plane (Y = 0). The surface number may be defined using mathematical expressions.

<Optional Description>

A string displayed and printed for easy identification of the surface. Could be something like 'Main floor'.

<ListOfPointNumbers>

Each surface may be bounded by between 3 and 500 corners, which all lie in a plane. Corner numbers refer to the corners, which must have been defined (e.g. using the ***Pt*** or ***CountPt*** statements) before using the surface statement. The order of listing must be as obtained by travelling around the surface's edge (in either direction). The list of corners must be on the same line. A room may contain up to 10000 surfaces.

Example 1, surface made from point 1, 2, 3, 4:

```
Surf   100   floor
  1     2     3     4
```

Example 2, surface made from point 1,2,10,11,12,13,14,4,5:

```
Surf   200   Ceiling
  1     2     10>14   4   5
```

If there is a need to programmatically build a list of points this can be done using the ***PList*** and ***ResetPList*** statements.

Building lists of points using PList and ResetPList

The ***PList*** and ***ResetPList*** statements are used in special cases together with the ***Surf*** statement. Twelve lists are predefined namely ***PList0*** to ***PList9*** (and ***PlistA*** and ***PListB*** which are handled automatically by ODEON). The ***PList*** statements allows to programmatically construct a list of points e.g. a list like:

100 110 120 130 140 150 160 170 180 190 200

this can be done using a ***for..end*** construct in the following way (adding a point number at a time):

```
for MyCounter 0 10
        PList0    100+MyCounter*10
end
```

It is also possible to add a number of points to a point list, e.g. another ***PList*** to a ***PList.*** In the following example ***PList1*** is assigned the points 100 110 120 130 140 150 160 170 180 190 200 10 11 12 13 15):

```
Plist1  Plist0 10>13   15
```

A point list can be referenced in the following way (adding point 1 before and 2 after the list in this example):

Surf Test_surface
1 PList0 2

To reset the list use the statement (list 0 used in this example):
ResetPList0

Multi Surface - MSurf
The multi surface ***MSurf*** is essentially just a variant of the ***Surf*** statement. Instead of typing one header line (e.g. ***Surf 1 A surface name***) for each surface, the header can be shared by multiple surfaces.

MSurf <SurfaceNumber><NumberOfSurfaces><Optional Description>
<ListOfPointNumbers1>
<ListOfPointNumbers2>
<ListOfPointNumbers3>
.......<NumberOfSurfaces> lines with lists of points describing each surface.

<SurfaceNumber>
A unique number from 1 to 2.147.483647 for identification of the surface. Using the same number, but with a negative sign defines the surface and its mirrored counter part in the XZ-pane (Y = 0). The surface number may be defined using mathematical expressions.

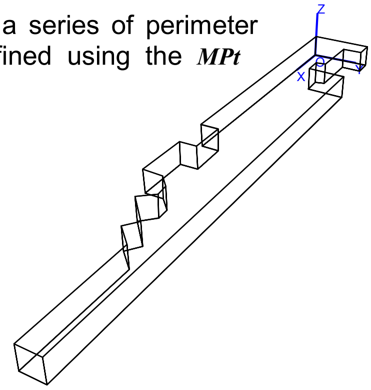
<NumberOfSurfaces>
The number of surfaces in the surfaces in the ***MSurf***.

<Optional Description>
A string displayed and printed for easy identification of the surface. Could be something like ***Main floor***.

Example on multi surface, containing 5 sub-surfaces:
MSurf 1 5 Steps on a stair
5544>5534 5112>5122
5111>5101 5212>5222
5211>5201 5312>5322
5311>5301 5412>5422
5411>5401 5512>5522

Elevation surface

Use The ***ElevSurf*** statement to define a series of vertical surfaces from a series of perimeter points plus an elevation height. The perimeter points are typically defined using the ***MPt*** statement. The syntax of ***ElevSurf*** is:

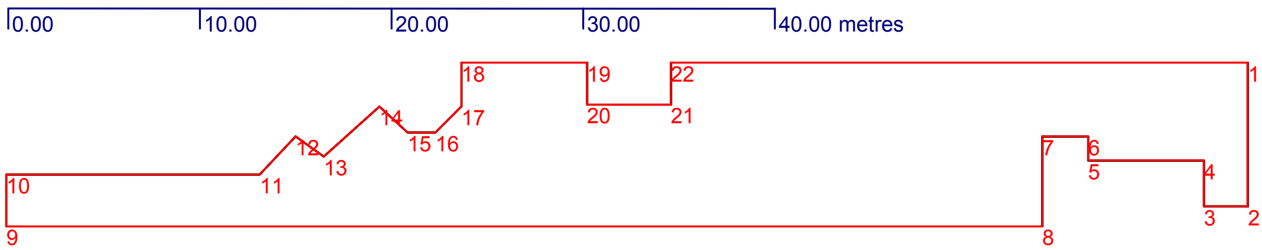


ElevSurf

FirstSurfaceNumber><FirstPointNumber><SectionsInElevSurf><Height><Optional name>

Example on use of the ***MPt*** and ***ElevSurf*** statements:

First the perimeter points (point 1 to 23) at the floor level of an office environment are described using the ***MPt*** statement. Then the elevation surface is created from these points, creating the perimeter walls of the office with a constant height of 2.7 metres. Finally the floor and ceiling is created using the ***Surf*** statement.



Perimeter points at the floor level

Example file: ***MPt and ElevSurf.Par***

Demonstrates the use of ***MPt*** (multi point), ***Surf*** (Surface) and ***ElevSurf*** (Elevation surface) statements. In this example the X-coordinates are made in absolute values whereas the Y-coordinates in most cases are in- or de-creased using the =+ or =- options

To create a closed ***ElevSurf*** (that is, the first wall joins the last wall) first and last point in the series of points handled to the ***ElevSurf*** must be identical - in this example, point 1 and point 23 are identical. If an elevation surface has 22 surfaces, then 23 points must be made available to the ***ElevSurf*** as in this example.

```
###
MPt 1 23
0 0 0
= 7.48 =
2.3 = =
= =-2.38 =
8.35 = =
= =-1.26 =
10.76 = =
= =+4.64 =
64.78 = =
= =-2.68 =
51.52 = =
49.62 =-2.02 =
48.22 =+1.1 =
45.3 =-2.68 =
43.85 =+1.45 =
42.40 = =
40.98 =-1.45 =
= 0 =
34.5 = =
= =2.1 =
30.13 = =
= =-2.13 =
0 0 0
```

ElevSurf 1 1 22 2.7 walls

Surf 200 Floor
1>22

Surf 201 Ceiling
24>24+22-1
###

Defining a number of surfaces using the CountSurf statement

The ***CountSurf*** is mostly here for backwards compatibility. In most cases it will be easier to use the ***Surf*** statement along with a ***for….end*** loop.
A Counter surface is divided on two lines and must follow the syntax:

CountSurf <First Surface Number><NumberOfSurfaces><Optional name>
<ListOfPointNumbers>

<FirstSurfaceNumber>

A unique number from 1 to 2.147.483647 for identification of the surface. Using the same number, but with negative sign defines the surface and its mirrored counterpart in the XZ-pane (Y = 0). A ***CountSurf*** will take up several surfaces numbers, which must all be unique.

<NumberOfSurfaces>

The number of surfaces to be created by the ***CountSurf*** call.

<Optional name>

Optional user defined name for easy identification of the surface, e.g. 'Beam'.

<ListOfPointNumbers>

Each surface may be bounded by between 3 and 50 corners, which all lie in a plane. Corner numbers refer to the corners, which must have been defined (e.g. using the ***Pt*** or ***CountPt*** statement) before using the surface statement. The order of listing must be as obtained by travelling around the surface's edge (in either direction). The list of corners must be on the same line. A room may contain up to 10000 surfaces.

Example:

CountSurf 1000 5 Beam in ceiling
1000 1100 1200 1300

will produce fives surface, the first containing the numbers given in the ***ListOfPointNumbers*** the next surface with 1 added to all the corners in the list etc.. Of course all the points refereed to need to be defined, typically this is done using a ***CountPt*** definition for each of the corners refereed to in the corner list of the ***CountSurf*** statement. In the above example the points 1000-1004, 1100-1104, 1200-1204 and 1300-1304 need to be defined.

Sample room files:
Beams.Par
BeamBox.Par
BeamBoxWithWimdows.Par

Revolution surface RevSurf

RevSurf must follow the syntax:

RevSurf <FirstSurfaceNumber><CurveStart1><CurveStart2><SectionsInRevSurf><Optional name>

The ***RevSurf*** command is typical used together with two ***CountPt*** statements to create a revolution surface using two 'curves' of points. The curves must contain the same number of points. The ***RevSurf*** command will always create a number of surfaces each build from four points.

<FirstSurfaceNumber>
A unique number from 1 to 2.147.483647 for identification of first surface in the revolution surface. Using the same number, but with negative sign, defines the surface and its mirrored counter part in the XZ-pane (Y = 0).

<CurveStart1>
First point number in the first revolution curve. The 'curve of points' is typically created using the ***CountPt*** statement and the curve must contain one more point than number of sections in the ***RevSurf***.

<CurveStart2>
First point number in the second revolution curve. The curve of points is typically created using the ***CountPt*** statement and the curve must contain one more point than the number of sections in the ***RevSurf***. The second curve must always contain the same number of points as the first curve.

<SectionsInRevSurf>
The number of surfaces to be created by the ***RevSurf*** statement. If creating a cylinder a number between 12 and 24 is suggested. Although it is easy to create many surfaces in a revolution surface, too many small surfaces should be avoided. If the ***FirstSurfaceNumber*** is 100 and ***SectionsInRevSurf*** is 3, surface 100, 101 and 102 will be created.

<Optional name>
Optional user defined name for easy identification of the surface, e.g. cylindric wall'.

Example:

```
RevSurf        1000    100    200    6     Cylinder
```

creates a revolution surface divided in 6 surfaces (surface 1000 - 1005). This call requires two curves of each 6+1 points to be defined, namely point 100 to 106 and point 200 to 206. If the two curves of points define corners in the lower and upper edge of a cylinder, a cylinder of 6 sections is created (see example room: ***RevSurfCylinder.Par***)

Loops using the FOR....END construct

The ***For*** statement must follow the syntaks:

For <CounterName><CountFrom><CountTo>

<CounterName>
Name of counter to be used by the ***For*** statement. The counter is automatically defined by the ***For*** statement and becomes undefined when the loop finishes. The counter can be referenced within the ***for..end*** loop as an ordinary constant or variable if desired so.

<CountFrom>
First value the counter takes. The ***CountFrom*** value is considered an integer value. If the number entered here is not an integer, it will be rounded to the nearest integer value.

<CountTo>
Last value the counter takes. The ***CountTo*** value must be greater or equal to the ***CountFrom*** value. The ***For*** statement will take ***CountTo-CountFrom+1*** loops. The ***CountTo*** value is considered an integer value, if the number entered here is not an integer, it will be rounded to the nearest integer value.

The following example will produce the points 1 to 5 with the X-coordinates 5, 10, 15, 20 and 25 metres, while the counter (***MyCounter***) loops through the values 1 to 5:

```
For MyCounter     1        5
  Pt      MyCounter        MyCounter*5    0      0
end
```

When using ***For..End*** constructs it should be remembered that point and surface number must be unique. This is easily obtained by incrementing the special variable ***NumbOffSet*** appropriately in each loop. An example on this kind of numbering can be found in the sample file ***ForColumnRoom.Par*** where ***NumbOffSet*** is incremented by eight in each loop (each time a new column, which contains 4 surfaces and 8 points, is created).

Sample room files:
ForRotunde.Par
BoxColumnRoom.Par

Unit

The ***Unit*** statement is used if you wish to do the modelling in a measure different from metres or indeed if the measures of the drawing from which you are getting the measures are imprecise (e.g. 1:97). The unit used in the parametric file is by default assumed to be metres, however if you prefer to model in another measure, this is possible using the ***Unit*** statement.

Example, modelling in Inches:

Unit ***Inches***

You may chose you unit among the following predefined:

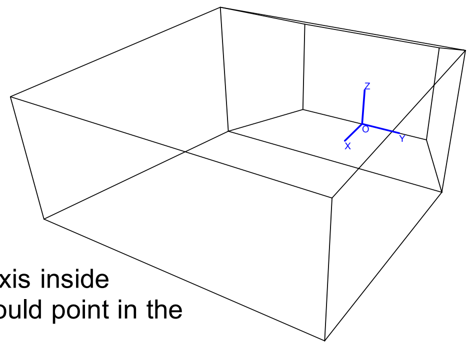
Metres, Centimetres, Millimetres, Inches, Feets and Yards

or if you need a different unit simply type the scaling factor from your unit into metres, e.g. ***Unit Inches*** corresponds to:

Unit 0.0254

CoordSys statement

The ***CoordSys*** statement is used if you wish the redefine the orientation or the coordinate system in which the geometry was modelled. The statement is typically used if the geometry was by accident modelled in an orientation different from the one assumed by ODEON or if it was imported from a CAD drawing where the orientation may also be different. To obtain the easiest operation inside ODEON the following orientation should be used (using a concert hall as the example):



- X-axis pointing towards the audience
- Y-axis pointing to the right as seen from the audience
- Z-axis pointing upwards

The syntax is:

CoordSys <X> <Y> <Z>

where X, Y, Z indicate which axis should be used as the x, y and z axis inside ODEON. X, Y and Z may also have a sign to indicate that the axis should point in the opposite direction.

Example 1, the default orientation (which is assumed by ODEON if the ***CoordSys*** statement is not used in the geometry file):

CoordSys X Y Z

Example 2, changing the direction of the X-axis:

CoordSys -X Y Z

Example 3, Swapping the X and the Z-axis

CoordSys Z Y X

User coordinate system

The ***UCS*** command is mostly there for compatibility with previous versions of ODEON – the coordinate manipulation functions ***MTranslate***, ***MRotateX***, ***MRotateY***, ***MRotateZ***, ***MScale***, ***MPop*** and ***MReset*** included from version 4.21 allow far more flexibility.

The ***UCS*** command must follow the syntaks:

UCS <TranslateX><TranslateY><TranslateZ><RotateZ>

The ***UCS*** command is used to create a User Coordinate System, with its own X, Y and Z translation. It also allows a rotation around the Z-axis (specified in degrees). All point definitions made after a ***UCS*** call will be created in the specified coordinate system. The default coordinate system is defined as:

UCS 1 1 1 0

The ***UCS*** command corresponds to:

MReset
MTranslate <TranslateX><TranslateY><TranslateZ>
MrotateZ <RotateZ>

If the ***UCS*** command does'nt fulfils your needs for coordinate manipulation you may use the matrix manipulation family ***MTranslate***, ***MRotateX***, ***MRotateY***, ***MRotateZ***, ***MScale***, ***MPop*** and ***MReset.***

Scale

The ***Scale*** command is mostly there for compatibility with previous versions of ODEON – the coordinate manipulation functions ***MTranslate***, ***MRotateX***, ***MRotateY***, ***MRotateZ***, ***MScale***, ***MPop*** and ***MReset*** included from version 4.21 allow far more flexibility.

The ***Scale*** command must follow the syntaks:

Scale <ScaleX><ScaleY><ScaleZ>

The scale command will multiply /scale all the points generated after the scale call, using the specified x,y and z-scale. The default setting is:

Scale 1 1 1

The ***Scale*** command evokes scaling of coordinates after all other coordinate manipulation is carried out. If you should need more advanced scaling options please use the ***MScale*** option.

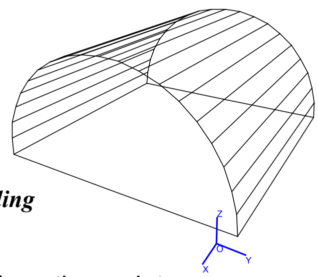
Coordinate manipulations – the M-family

Advanced coordinate manipulation can be carried out using matrix manipulation. The coordinate manipulation functions, which is essential to the use of hybrid statements (***Box***, ***Cylinder***, etc.) is implemented as the following functions:

MTranslate <TranslateX><TranslateY><TranslateZ>
MRotateX <Rotation angle>
MRotateY <Rotation angle>
MRotateZ <Rotation angle>
MScale <ScaleX><ScaleY><ScaleZ>
MPop
MReset

The manipulations carried out by the M-family are accumulative. This means that you can specify more than one operation to be carried out, e.g. first rotate 90° around the Z-axis, then rotate 90° around the Y-axis and finally translate 10 metres upwards. The following example shows these operations carried out on a cylinder shell (the ***Cylinder*** statement is described later):

```
Manipulating a cylinder.Par
###
MRotateZ   90
MRotateY   90
MTranslate        0      0      10
Cylinder    1     20     5      180     10          TB Cylindrical ceiling
###
```

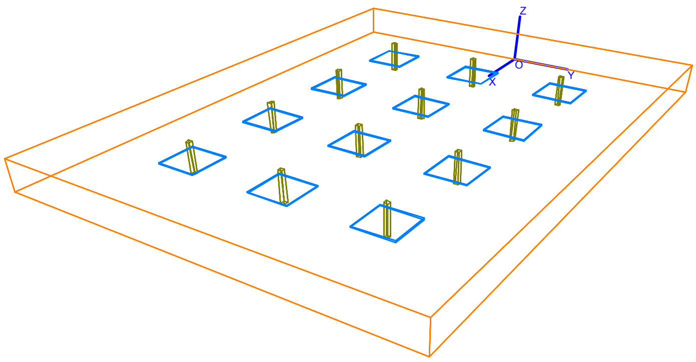


The transformation commands to be carried out must always be stated before the points /geometry on which they should work is created. To reset all previous coordinate manipulations, use the ***MReset*** command. To reset /cancel the most resent manipulation (***MTranslate*** in the example above), use the ***MPop*** command (which will pop the operation of the matrix stack).

Hints!

The order in which the coordinate manipulations are carried out is important, usually (but not necessarily always), the ***MScale*** commands should come first, then the ***MRotate*** commands and finally the ***MTranslate*** commands.
If you are not familiar with coordinate manipulations, it may be a good exercise to try different manipulations on the sample geometry above and load the geometry into ODEON upon each change.

Using layers in ODEON



The ***Layer*** statement allows dividing a geometry into separate parts, which can be displayed separately and in its own layer-colour in the 3DView, 3DOpenGL and Materials list. This makes it easier to model and investigate selected groups of surfaces. When importing geometry from a DXF file (e.g. from AutoCAD where layers are an integrated part) the layers included in that file will be preserved in the imported version of the room.

If layers have been used in a geometry, the layer can be activated or deactivated in the 3DView, 3DOpenGL and Materials list. The layers menu is activated from these windows using the Ctrl+L shortcut.

Vocabulary - what's a layer?

Layers are commonly used in CAD modelling programs such as AutoCAD in order to make complicated geometries more manageable. Layers in CAD programs (and some drawing and picture editing programs) can be compared to overhead sheets (without any thickness). You define a number of layers with different names (and possibly different line colour/thickness etc.) and draw the different parts of your geometry on the different layers. The layers can be turned on or off in the CAD program allowing better overview by hiding parts of the geometry that are not relevant in a part of the 'drawing' process.

Syntax for the Layer statement:

Layer <"Layer name in quotes"> <R-intensity><G-intensity><B-intensity>

or as another option:

Layer <"Layer name in quotes"> <LayerColour>

<"Layer name in quotes">
A descriptional name, which must start and end with a quote sign (").

<R-intensity><G-intensity><B-intensity>
Three floating-point values between 0 and 1, which together is describing the colour of the layer as a Red-Green-Blue intensity. If using the Layer command Shift+Ctrl+L from within the Odeon editor, the colour intensities are set by clicking the desired colour in a dialog-box.

<LayerColour>
As an option the colour of the layer can be described, using one of the predefined colours Black, Blue, Cream, Fuchsia, Gray, Green, Lime, Maroon, Navy, Olive, Purple, SkyBlue, Teal, or White. The LayerColours.par example demonstrates the different colours.

The LayerStatement.par example shows how to create a geometry on three different layers.

Selected surfaces can be selected for display in the 3DView, 3DOpenGL and the Materials list.

```
BoxcolumnRoom.Par
###
const L 40
const W 30
const H 3
const NumColX 4
const NumColY 3
const ColumnW 0.3

MTranslate l/2 0 0
Layer "Walls" 1.000 0.502 0.000    ;orange colour
 Box 1 l w h tb walls in the room
```

```
 MPop
 :modelling the columns

for ColYCnt 1 NumColY
 for ColXCnt 1 NumColX
  MReset
  MTranslate ColXCnt*L/(NumColX+1) w/2-ColYCnt*W/(NumColY+1) 0

  NumbOffSet Auto
  Layer "Columns" 0.502 0.502 0.000
  Box 1 ColumnW ColumnW h n columns in the room ;olive colour
  NumbOffSet Auto
  MTranslate 0 0 1.2
  Layer "Table plates" 0.000 0.502 1.000  ;bluish colour
  Box 1 3 3 0.1 tb tables
 end
end
###
```

Symmetric modelling
Symmetric rooms can be modelled, taking advantage of the ODEON convention for symmetric models. This allows generation of symmetric or semi symmetric rooms with symmetry around the XZ-plane, Y = 0 – symmetric modelling is always carried out in the main coordinate system, it does not take into account manipulations carried out using ***UCS***, ***MTranslate*** etc.
Modelling a surface symmetric around the main axis' (e.g. a reflector above the stage) can be done using symmetric points. Modelling left and right walls at the same time can be done using a symmetric double surface.

Symmetric points
Surfaces symmetric around the XZ-plane, Y= 0 can be made using symmetric points. If defining the point:

```
Pt    2          1.0    1.0    1.0
```

in the geometry file, using the point -2 in a surface definition of the geometry file will refer to the auto generated point:

```
Pt    -2         1.0    -1.0   1.0
```

Thus the following surface definition:

```
Surf    1000  Symmetric surface
  1     2       -2      -1
```

will model a surface symmetric around the XZ-plane, Y=0 (e.g. an end wall or a reflector).

If the surface is completely symmetric as above then the symmetric points can also be specified using the Mirror word, which should be the last component in the corner list:

```
Surf    1000  Symmetric surface
  1     2       Mirror
```

Note: You should not try to define the point -2 in the geometry file it is automatically generated.

Symmetric double surface
Symmetric double surfaces are pairs of surfaces symmetric around the XZ-plane, Y= 0 (e.g. a right and a left wall):

```
Surf    -2      Right wall/ Left wall
  12    22      23      13
```

will appear as two surfaces inside the ODEON program. Thus you will have the following two surfaces (inside the ODEON program):

```
-2      Right/ Left wall **Double symmetric surface 1**
```

containing the symmetric points:

```
-12     -22     -23     -13
```

and the surface:

2 Right/ Left wall **Double symmetric surface 2*

containing the points:

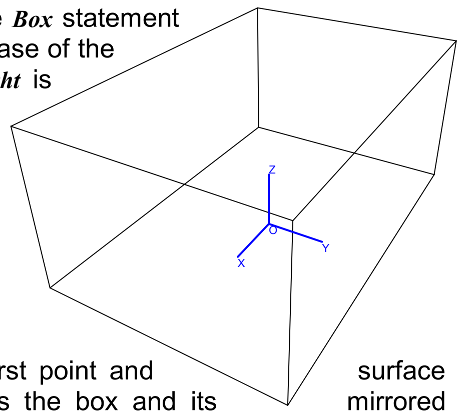
12 22 23 13

as they are defined in the geometry file.

Note: You may not define surface 2, if you are using the symmetric double surface -2, because ODEON automatically generates surface number 2.

The Box statement

The ***Box*** statement defines a Box with or without top and bottom. The ***Box*** statement may typically be used for Box shaped rooms and columns. A special case of the Box statement is when one of the dimensions ***Length***, ***Width*** or ***Height*** is zero; in this case only one surface is created.



The syntax of the ***Box*** statement is:

Box <Number><Length><Width><Height><T/B/N><optional name>

<Number>

A unique number from 1 to 2.147.483647 for identification of the first point and surface in the ***Box***. Using the same number, but with negative sign defines the box and its mirrored counterpart in the XZ-pane (Y = 0). A Box will take up several point- and surface numbers, which must all be unique.

<Length>

Length is oriented in the X-direction on the figure.

<Width>

Width is oriented in the Y-direction on the figure.

<Height>

Height is oriented in the Z-direction on the figure.

<T/B/N>

The ***T/B/N*** parameter specifies whether the ***Box*** should have a top and /or a bottom. The options are ***T***, ***B***, ***TB*** and ***N*** (for none).

Insertion point:

The insertion point of the Box is always the centre of the floor (bottom) surface.

Special cases:

If one of the dimensions ***Length***, ***Width*** or ***Height*** equals zero only one surface is created.

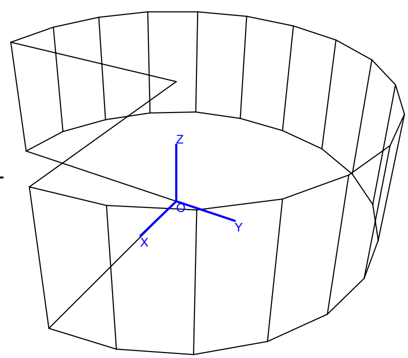
Connection points:

The four foot-points in ***Box*** are stored in ***PlistA***
The four top-points in ***Box*** are stored in ***PListB***

The ***Box*** example shown was generated with the following code:

```
BoxStatement.par
###
const L 6
const W 4
const H 2.7

Box 1  L W H  TB  Walls, floor and ceiling
###
```

The Cylinder statement



The *Cylinder* statement defines a cylinder shell with or without top and bottom. The statement may typically be used for modelling cylindrical room or columns. The *Cylinder2* statement, which creates a cylinder of the calotte type, will usually be preferable for modelling cylindrical ceilings.

The syntax for Cylinder is:

Cylinder<Number><NumberOfSurfaces><Radius><RevAngle><Length><T/B/N><optional name>

<Number>

A unique number from 1 to 2.147.483647 for identification of the first point and surface in the ***Cylinder***. Using the same number, but with negative sign defines the cylinder and its mirrored counterpart in the XZ-pane (Y = 0). A Cylinder will take up several point- and surface numbers, which must all, be unique.

<NumberOfSurfaces>

For a full cylindrical room (with a revolution angle of 360°), around 16 to 24 surfaces are recommended. For columns a number between 6 to 8 is recommended.

<Radisus>

Radius of the cylinder must always be greater than zero.

<Revangle>

Revangle must be within the range +/-360° and different from zero. If ***RevAngle*** is 180°, a half cylinder is generated, if its 360° a full cylinder is generated. Positive revolution angles are defined counter clockwise.

<Height>

If the height is less than zero, the orientation of the cylinder is inverted. If height equals is zero, one circular surface is generated.

Insertion point:

The insertion point of the cylinder is always the centre of the floor (bottom) surface.

Connection points:

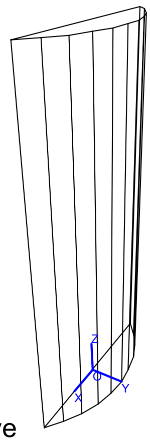
The foot-points in ***Cylinder*** are stored in ***PlistA***
The top-points in ***Cylinder*** are stored in ***PListB***

The example shown was generated with the following code:

```
CylinderStatement.Par
###
const N 16
const R  15
const  H  10

Cylinder 1000 N R 270 H TB Cylindrical room
###
```

Hint! The cylinder can be made elliptical, using the ***MScale*** statement.

The Cylinder2 statement



Cylinder2 is a cylinder shell of the calotte type. Rather than specifying the radius and revolution angle, ***Cylinder2*** is specified in terms of the ***width*** and ***height***. ***Cylinder2*** is typically used for cylindrical /curved ceilings.

The syntax for Cylinfer2 is:

Cylinder2<Number><NumberOfSurfaces><Width><Height><Length><T/B/N><optional name>

<Width>

Width is oriented in the X direction on the figure.

<Height>

Height of the cylinder shell is oriented in the Y direction on the figure and may be positive (concave shell) as well as negative (convex shell). ***Height*** must be different from zero and less or equal to ½****Width***.

<Length>

Length of the cylinder shell is oriented in the Z direction the figure. If ***Length*** is negative the orientation is inverted.

Insertion point:

The insertion point of ***Cylinder2*** is always foot point of the calotte floor (bottom) surface.

Connection points:

The foot-points in ***Cylinder2*** are stored in ***PlistA***
The top-points in ***Cylinder2*** are stored in ***PListB***

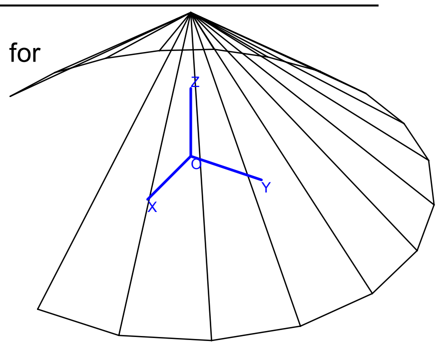
The example shown was generated with the following code:

```
###
Const N 10
Const W 5
Const H 1
Const L 10

Cylinder2 1 N W H L TB Cylinder calotte
###
```

Hint! The cylinder can be made elliptical, using the ***MScale*** statement.

The Cone statement

The ***Cone*** statement models a cone. Typical use of the ***Cone*** statement is for modelling half cone or cone shaped ceilings.



The syntax for ***Cone*** is:

Cone <Number><NumberOfSurfaces><Radius><RevAngle><Height><optional name>

<Number>

A unique number from 1 to 2.147.483647 for identification of the first point and surface in the Cone. Using the same number, but with negative sign defines the surface and its mirrored counterpart in the XZ-pane (Y = 0). A Cone will take up several point- and surface numbers, which must all be unique.

<Radius>

Radius of the Cone must always be greater than zero.

<Revangle>

Revangle must be within the range +/-360° and different from zero. If ***RevAngle*** is 180°, a half-cone is generated, if its 360° a full cone is generated. Positive revolution angles are defined counter clockwise.

<Height>

The height must be different from zero. If the height is less than zero, the orientation of the cone is inverted. Height is oriented in the Z-direction on the figure.

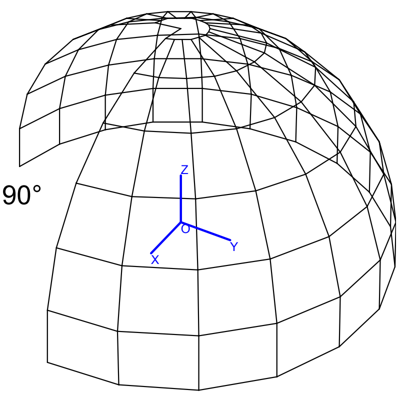
The ***Cone*** example shown was generated with the following code:

```
###
const N 16
const R  15
const  H  10

Cone 1 N  R 270 H Cone shaped ceiling
###
```

Hint! The cone can be made elliptical, using the ***MScale*** statement.

The Dome statement

The ***Dome*** statement generates a full dome (half hemisphere) covering the full 90° vertical angle. In most cases the ***Dome2*** statement is probably better suited.



The syntax for ***Dome*** is:

Dome<Number><NumberOfSurfaces><Radius><RevAngle><optional name>

<Number>
A unique number from 1 to 2.147.483647 for identification of the first point and surface in the ***Dome***. Using the same number, but with negative sign defines the dome and its mirrored counterpart in the XZ-plane (Y = 0). A ***Dome*** will take up several point- and surface numbers, which must all be unique.

<Radius>
Radius of the ***Dome*** must always be greater than zero.

<Revangle>
Revangle must be within the range +/-360° and different from zero. If ***RevAngle*** is 180°, a half ***Dome*** is generated, if its 360° a full ***Dome*** is generated. Positive revolution angles are defined counter clockwise.

Connection points:
The right side vertical points in ***Dome*** are stored in ***PlistA***
The left side vertical points in ***Dome*** are stored in ***PlistB***
In the special case where the revolution angle is 180°, all points are stored in ***PlistA*** and the number of vertical subdivisions is stored in ***ONVert.***

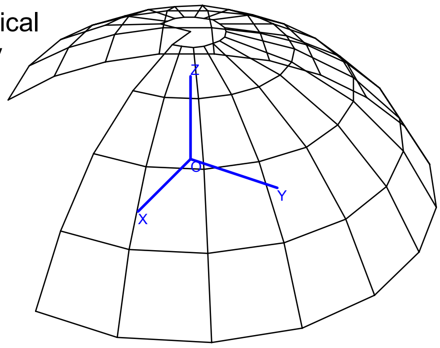
The example shown was generated with the following code:

```
###
const N 16
const R  15

Dome 1 N R 270  This is a dome
###
```

Hint! The dome can be made elliptical, using the ***MScale*** statement.

The Dome2 statement

The ***Dome2*** statement is a Dome shell of the calotte type, where the vertical revolution angle is not necessary 90°. Rather then specifying the dome by a revolution angle, it is specified by the width and height. ***Dome2*** may typically be used for modelling dome shaped ceilings.



The syntax for ***Dome2*** is:

Dome2<Number><NumberOfSurfaces><Width><Height><RevAngle><optional name>

<Number>

A unique number from 1 to 2.147.483647 for identification of the first point and surface in the Cone. Using the same number, but with negative sign defines the surface and its mirrored counterpart in the XZ-pane (Y = 0). A Cone will take up several point- and surface numbers, which must all be unique.

<NumberOfSurface>

Specifies the number of surfaces in one horizontal ring of the dome, around 16 to 24 surfaces per ring is suggested. ODEON will automatically calculate the number of subdivisions in the vertical level. If the revolution angle is 180° the number is stored in the ***ONVert*** variable would have been 9 in the example above. The ***ONVert*** variable may help when connecting a ***Dome2*** to a ***Cylinder2*** in order to specify the correct number of surfaces in the cylinder.

<Width>

Width at the beginning of the dome. The width must always be greater than zero.

<Revangle>

Revangle must be within the range +/-360° and different from zero. If ***RevAngle*** is 180°, a half-cone is generated, if its 360° a full cone is generated. Positive revolution angles are defined counter clockwise.

<Height>

The ***Height*** must be different from zero. If the height is less than zero, the orientation of the dome is inverted. ***Height*** must be different from zero and less or equal to ½****Width***.

Connection points:

The right side vertical points in ***Dome2*** are stored in ***PlistA***
The left side vertical points in ***Dome2*** are stored in ***PlistB***
In the special case where the revolution angle is 180°, all points are stored in ***PlistA*** and the number of vertical subdivisions is stored in ***ONVert.***

The example shown was generated with the following code:

```
###
Const N 16
Const W 10
Const H 3
Const L 10

Dome2 1 N W H 270 Dome calotte
###
```

Hint! The cylinder can be made elliptical, using the ***MScale*** statement.

3.2.2 Creating a new .Par file - time saving hints

The golden rule when creating a .Par file to model a room is to think carefully before you start typing. For very simple rooms, it is not too difficult to keep track of things, but for realistically complex rooms a systematic approach is desirable. You will typically have a set of drawings, which have to be used as the basis for the ODEON model. It pays to spend quite a long time working out how the room can be simplified to a manageable number of sensibly shaped plane surfaces, sketching over the drawings. These ideas will have to be modified when you start to work out the actual coordinates, to ensure that the surfaces really are plane. Here are some ideas that may help you to create correct surface files faster:

- Number corners and surfaces systematically, with relation to their physical location in the room. Group corner and surface numbers according to the part of the room they describe (e.g. floor, 1st balcony, 2nd balcony, ceiling).
- Exploit symmetry: If the room has an axis of symmetry, place a coordinate axis on it. Then use the 'sign'- convention for symmetric /semi symmetric modelling.
- If there are vertical walls and /or features, which repeat vertically (e.g. identical balconies), use the ***CountPt***, ***CountSurf***, ***RevSurf*** statements or indeed ***For..End*** constructs.
- Build the room gradually, testing the .Par file at each stage of growth by loading it into ODEON and have a look at the result.
- Use hybrid statements such as ***Box***, ***Cylinder*** etc.

Where it is difficult to get surfaces to meet properly without either warping or lots of small surfaces to fill the gaps, allow the surfaces to cut through each other a little. This will usually ensure a watertight result, and has only minor drawbacks. These are (i) the apparent surface area will be a little too large, affecting reverberation times estimated using Quick Estimate Reverberation and the room volume estimated by Global Estimate, (ii) crossing surfaces can look odd and hinder clarity in the 3D displays.
Do not try to include small geometrical details at the first attempt. If there are some large surfaces, which are basically plane but contain complex geometrical features (e.g. a coffered ceiling), model them at first as simple planes. Then first when this room has been made watertight, make the necessary alterations to the geometry file. The simplified version can also be used in the prediction exercises, to give some idea of the effect of the feature in question.

3.2.3 Examples on parametric modelling

This section will give some short examples on the modelling of rooms using the parametric modelling language of ODEON. The options in this modelling format are many, ranging from typing the model number by number, to dedicated programming. This section will try to give an idea on how to use the language and its keywords. In the default room directory created at the installation of ODEON you may find several other examples on the .Par format.

Four ways to model a box
These examples show four ways to model a box shaped room; using plain numbers, using constants, using constants plus symmetric modelling and using the ***Box*** statement along with the ***MTranslate*** statement. In each example the dimensions of the room are: (W, L, H) = (4, 6, 2.7).

Below the box shaped room is modelled using plain decimal numbers:

```
Parametric sample BoxFromPureNumbers.par
###
Pt      1       0       2       0
Pt      2       0       -2      0
Pt      3       6       -2      0
Pt      4       6       2       0
:ceiling points
Pt      11      0       2       2.7
Pt      12      0       -2      2.7
Pt      13      6       -2      2.7
Pt      14      6       2       2.7
Surf    1       floor
  1     2       3       4

Surf    2       ceiling
  11    12      13      14
Surf    3       end wall
  1     2       12      11

Surf    4       end wall
  1     2       12      11

Surf    5       side wall
  1     4       14      11

Surf    6       side wall
  2     3       13      12
###
```

Below the box shaped room is modelled using constants for the definition of W, L and H. Some of the advantages of using parameters in modelling rooms are that it makes changes to a model much easier (allowing reuse) and often it will also improve the clarity of a model data.

```
Parametric sample BoxFromParameters.par
The box measures are:
Width  = 4 metres
Length = 6 metres
Height = 2.7 metres
###
const    W      4
const    L      6
const    H      2.7

Pt       1      0      W/2    0
Pt       2      0      -W/2   0
Pt       3      L      -W/2   0
Pt       4      L      W/2    0

Pt       11     0      W/2    H
Pt       12     0      -W/2   H
Pt       13     L      -W/2   H
Pt       14     L      W/2    H

Surf     1      floor
  1>4

Surf     2      ceiling
  11>14

Surf     3      end wall
  1      2      12     11

Surf     4      end wall
  1      2      12     11

Surf     5      side wall
  1      4      14     11

Surf     6      side wall
  2      3      13     12
###
```

Below the box shaped room is modelled using parameters and symmetric modelling syntax (signs on point and surface numbers). The symmetric modelling syntax means less typing and less typing errors:

```
Parametric sample BoxFromParametersUsingSymmetricModelling.par
###
const    W      4
const    L      6
const    H      2.7

Pt       1      0      W/2    0
Pt       2      L      W/2    0

Pt       11     0      W/2    H
Pt       12     L      W/2    H

Surf     1      floor
  1      2      -2     -1

Surf     2      ceiling
  11     12     -12    -11

Surf     3      end wall
  1      -1     -11    11

Surf     4      end wall
  2      -2     -12    12

Surf     -5     side wall
  1      2      12     11
###
```

Below the box shaped room is modelled using the ***Box*** statement, which is the easiest way to create this simple geometry. A ***MTranslate*** statement is used to insert the ***Box*** at the same position as in the three other examples:

```
Parametric sample BoxStatement.Par
###
const L 6
const W 4
const H 2.7

MTranslate l/2 0 0

Box 1 l w h tb  Walls and floor
###
```

Modelling a cylinder

This example shows two different ways to create a cylindrical room with a floor and a ceiling. In the first example the room is modelled using the Cylinder statement:

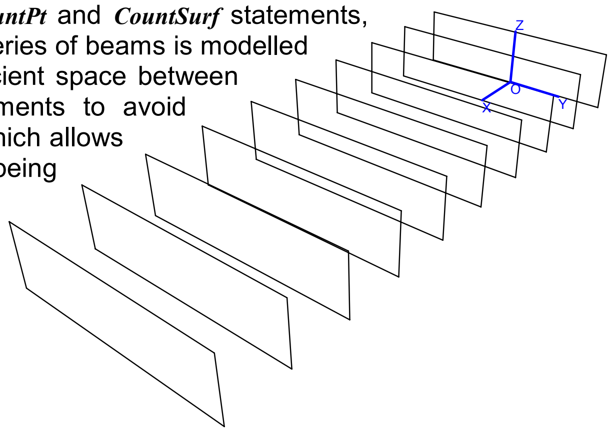
```
Parametric Sample CylinderStatement.Par
###
const N 16
const R  15
const  H  10

Cylinder 1000      N      R      360     H      TB Cylindrical room
###
```

The ***Cylinder*** statement is of course the easiest way to model a cylinder, however sometimes more flexibility is needed (e.g. different radius in top and bottom). In the second example the corners in the room are modelled using the ***CountPt*** statement and the cylindrical surfaces are modelled using the ***RevSurf*** statement. Notice that the number of points created by the ***CountPt*** statement is one higher than the number of sections in the ***RevSurf*** statement. The bottom and top of the room is modelled using the ***Surf*** statement, notice that points used by these surfaces are referenced using the statement ***100>100+Sections-1*** rather than writing each of the sequential points, this is not only a faster way to write things, it also allows a rapid change to the number of sections in the cylinder by simply changing the ***N*** constant.

```
Parametric sample, a cylinder RevSurfCylinder.Par
###
const N     16
const R     15
const H     10

CountPt     100  N+1  R*CosD((PtCounter)*360/N)   R*SinD((PtCounter)*360/N)  0
CountPt     200  N+1  R*CosD((PtCounter)*360/N)   R*SinD((PtCounter)*360/N) H

RevSurf 300  100 200            Sections  cylinder walls

Surf        100       Circular floor
  100>100+N-1

Surf        2 Circular ceiling
  200>200+N-1
###
```

Modelling a number of beams

Many rooms have repeating parts, e.g. construction beams in a ceiling. The following example shows how a number of beams (ten) can be modelled using the ***CountPt*** and ***CountSurf*** statements, together with the symmetric syntax. Each corner in the series of beams is modelled using a ***CountPt*** statement. It is important to leave sufficient space between the series of points generated by the ***CountPt*** statements to avoid repeated point numbers, in this case the space is 100 which allows ***Nbeams*** to take the value 100 without repeated points being generated.




```
###
const BeamHeight          2
const BeamWidth           7
const BeamDist            2
const NBeams              10

:beam corner points;
:first beam has the 4 points          1000    1100    -1100   -1000
;second beam has the four point       1001    1101    -1101   -1001
;third beam has the four point  1002  1102    -1102   -1002
CountPt 1000     NBeams  PtCounter*BeamDist     BeamWidth/2     0
CountPt 1100     NBeams  PtCounter*BeamDist     BeamWidth/2     BeamHeight

CountSurf  1000  NBeams  Front side of beam
  1000     1100     -1100     -1000
###
```

Modelling a box shaped room with columns in two dimensions, using two level For..End constructs

When modelling geometries having more than one level of symmetry it is advantageous to use ***For..End*** constructs. This example shows how to model columns in two dimensions in a room using a two level ***For..End*** construct.

Each column is created using 8 points and 4 surfaces, thus the numbering used by points and surfaces is incremented by 8 each time a column is created. This is done by incrementing the predefined variable ***NumbOffSet*** by eight for each column in order to make surface and point numbers unique. The different positions of the points used for each column are obtained, using ***MTranslate and MReset.***

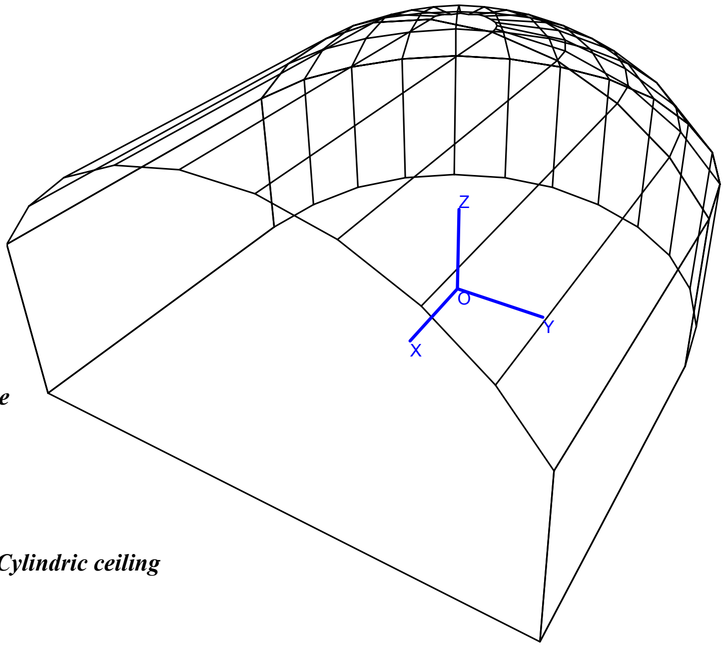
```
Parametric sample BoxColumnRoom.Par
###
 const L 10
 const W 4
 const H 3
 const NumColX 4
 const NumColY 3
 const ColumnW 0.3

mTranslate l/2 0 0
Box 1 l w h tb walls in the room
:modelling the columns

for ColYCnt 1 NumColY
MReset
   MTranslate  L/(NumColX+1) w/2-ColYCnt*W/(NumColY+1)  0
   for ColXCnt 1  NumColX
              NumbOffSet NumbOffSet+8      ;comment: hint! setting NumbOffSet to Auto would do the same job
              Box 1 ColumnW  ColumnW  h n columns in the room
              MTranslate L/(NumColX+1) 0 0
   end
end
###
```

Using hybrid statements and coordinate manipulation
The following example demonstrates an example on how to use the hybrid statements ***Cylinder2*** and ***Dome2*** as well as the coordinate manipulations, which are essential to the use of the hybrid statements.

This example is a rather complex one, so the main parts of the file is explained below:

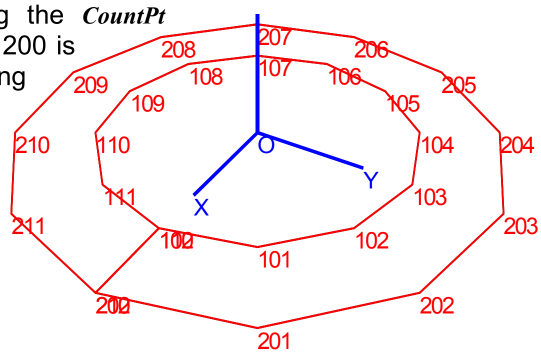
Line 3-7 Defining constants.
Line 8-9 Inserting the cylindrical wall, which needs a rotation of 90° around the Z-axis.
Line 11 The foot-points of the cylindrical wall, which is temporarily stored in ***PlistA*** are stored in ***Plist0*** for later use (definition of the floor).
Line 12-13 Inserting the dome shaped ceiling. The Z-rotation has already been set to 90° when the wall was created.
Line 14-18 Setting the coordinate manipulation for the ceiling and creating the ceiling.
Line 19 Resetting the coordinate manipulation to work in absolute coordinates
Line 20-23 Creating Wall /floor point
Line 24-25 Defining floor, using the 'cylinder points' stored in ***Plist0***.
Line 28-29 Defining side walls using symmetric modelling.
Line 30-31 Defining back wall, using the 'ceiling cylinder points' which is still stored in ***PListB***.

```
1.  Dome2 and cylingder2 x 2 room.par
2.  ###
3.  const H 5
4.  Const L 10
5.  Const W 15
6.  Const N 12
7.  Const HCurve 4

8.  MRotateZ 90
9.  Cylinder 1000 N W/2 180 H N
10. //Storres PListA for later use with
    floor
11. PList0 PListA

12. MTranslate 0 0 H
13. Dome2 2000 N W HCurve 180 Halfdome

14. MReset
15. MRotateZ 90
16. MRotateY 90
17. MTranslate 0 0 H
18. Cylinder2 3000 ONVert W HCurve L n Cylindric ceiling

19. MReset
20. Pt 1  0 W/2 0
21. Pt 2    L W/2 0
22. Pt 3 0 W/2 H
23. Pt 4    L W/2  H

24. Surf 1 Floor
25. 2 PList0 -2

26. //done with PList0 its a good habit to Reset it
27. ResetPList0

28. Surf -2 Side Walls
29. 1 2 4 3
30. Surf 3 BackWall
        2     PListB -2
31. ###
```



Defining surfaces with concave edges
Most surfaces in the geometries used with ODEON will probably be have convex edges (rectangles, cylindrical surfaces etc.), however in ODEON, it is possible to define surfaces with cavities, even surfaces with holes. Such surfaces are defined just like any other surface, by creating a list of corners where the listing is obtained by travelling around the surface's edge (in either direction). Below are two examples; one with a donut shaped balcony floor and another with a cylindrical window opening in a ceiling.

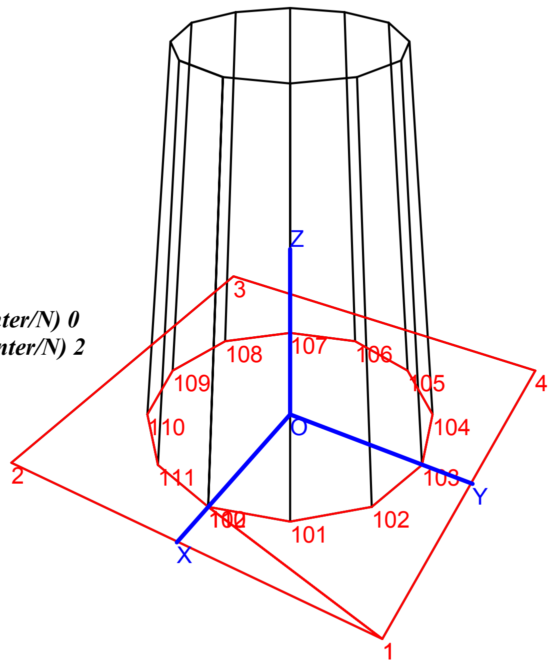
In the donut example two 'rings of corners' are created using the ***CountPt*** statement, notice that the point 100 is equal to point 112 and point 200 is equal to point 212. The donut surface is created, simply by connecting the inner and outer ring of points into one surface. It doesn't matter whether one of the rings are created clock or counter-clockwise. The surface is created from the following list of points: ***100,101,102,…,110,111,112,200,201,202,….,210,211,212***




```
DonutSurface.par
###
Const R1 10
Const R2 15
Const N 12
CountPt      100 N+1          R1*CosD(360*PtCounter/N)          R1*SinD(360*PtCounter/N) 0
CountPt 200  N+1              R2*CosD(360*PtCounter/N)          R2*SinD(360*PtCounter/N) 0

Surf 100 Donut surface
 100>100+N 200>200+N
###
```

The window example shows how a cylindrical window opening is created in ceiling surface. The interesting surface in this example is surface 1, the ceiling surface. The surface is created from the following list of points:1,100,101,102,103….,111,112,1,2,3,4

```
CeilingWithWindowTube.par
###
Const R1 0.75
Const R2 0.5
Const N 12

Pt 1 1 1 0
Pt 2 1 -1 0
Pt 3 -1 -1 0
Pt  4 -1  1  0

CountPt 100 N+1 R1*CosD(360*PtCounter/N) R1*SinD(360*PtCounter/N) 0
CountPt 200 N+1 R2*CosD(360*PtCounter/N)  R2*SinD(360*PtCounter/N) 2

Surf 1 Ceiling
 1 100>100+N 1>4

RevSurf 2 100 200 N Window tube

Surf 100 Window /glass
 200>200+N-1

###
```



3.3 Importing DXF files, guidelines

ODEON 4 and later supports import of DXF files (Drawing eXchange Format) enabling import of CAD models exported from CAD programs such ase AutoCAD. You should be aware that geometries created in a CAD program might not necessarily be suitable for calculations in a room acoustical program such as ODEON. The demands to the CAD model are further described below.

3.3.1 About CAD drawings (and DXF files)

Room models to be used by ODEON must in general be surface models of plane surfaces, no matter if it is modelled using a CAD program or if they are modelled directly (using the ODEON ASCII format). Also general rules for good room acoustical modelling behaviour applies (please refer to chapter 3.1) and you should remember to perform check on the room model once imported into ODEON as the model may look nice in the CAD program while containing serious errors with respect to room acoustical modelling.

For a CAD model exported in the DXF format to be useful using the ODEON Import option, the models must be surface models and surfaces in the room model must be created using the (AutoCAD) entity named 3DFACE or POLYLINE (Mesh, RuleSurf, RevSurf,TabSurf, Cylinder, Sphere, Dome etc.). Finally the surfaces must appear explicit in the drawing.
To support as many CAD programs as possible, elevations of the LINE and LWPOLYLINE entities are partially supported and it can be selected during the import phase to accept closed 3DPoly's (also the POLYLINE entity)

as surfaces. Solid modelling i.e. using the EXTRUDE command is not supported – any drawing entity including texts and dimensioning lines which is not supported is simply ignored by ODEON in the import phase.

The surfaces will when imported to ODEON carry the name of the layer on which they were drawn. Use the layer name to give the different parts of the geometry different names, e.g. draw the stage floor surfaces on a layer named *Stage floor*, the sidewalls on a layer named *Sidewalls* etc.
If you are modelling subdivided surfaces such as *Upper wall* and *Lower wall* because you wish to be able to assign different materials to these parts of a surface, it is a very good idea to model these parts on different layers in order to avoid the ODEON Import DXF files option called Glue Surfaces from adding these surfaces together (described below).

Surfaces that appear implicit in the drawing (AutoCAD)
Surfaces that have been inserted into the CAD drawing, using blocks must be exploded (using the EXPLODE command from within AutoCAD) for the entities to appear explicit in the drawing.

3.3.2 Using the DXF Import option

To import a DXF file:

- Select Files|Import DXF files
- Specify the input file e.g. MyCADRoom.dxf
- Specify the destination file e.g. MyCADRoom.Par

Finally specify the parameters described below as appropriate to begin the translation.

Unit in input file
Unfortunately a dxf-file carries no information on the unit used in a drawing. It is important that the correct unit in which the geometry was modelled is selected in the import dialog. If the correct unit is not specified the import process may fail because the geometry seems to be only few millimetres large or several kilometres in size.

Geometric rules, glue surfaces
3DFACE surfaces are by nature surfaces build using three or four sets of coordinates. In order to create a complex surface (containing a lot of points) using the 3DFACE surface in AutoCAD, it is necessary to build it combining a number of 3DFACE surfaces each containing 3 or 4 points. This will lead to an increased number of surfaces, thus a model that is visually unclear and which doesn't combine well with the image source theory.
When importing DXF models with the Glue surfaces option turned on, ODEON will try to combine the surfaces into a reduced number of surfaces (having bigger areas). ODEON will not combine surfaces with each other, which are situated on different layers in the CAD drawing, thus if you wish that certain surfaces are not 'glued' together, e.g. if upper and lower part of a wall should be assigned different materials, either draw the surface on different layers in the CAD program (preferable) or turn off the glue surface's option.

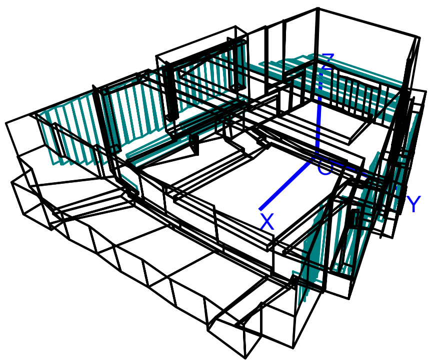
Max. point margin
If points in the DXF file are within this margin, the points will be considered the same. Allowing a certain amount of point margin will allow the Glue function to perform better if the coordinates in the model are not exact. However if you have modelled both sides of a surface (e.g. outside and inside a surface of a balcony front edge) the Max point margin should be smaller than the distance between these surfaces otherwise the points on either side of the surface will be considered the same, with disastrous results.

Max warp
ODEON will split four point surfaces into 2 three-point surfaces if the surface's warp exceeds this warp. The glue option on the other hand will try to glue surfaces as long as this warp is not exceeded.

3.3.3 Editing the imported geometry

It will often be necessary or at least desirable to make changes to a geometry once it has been imported. The 3DView, which display the geometry one it has been imported, may be useful for this purpose, please see the context sensitive help available from within Odeon for further details.

Example: Importing the supplied ElmiaDXFSample.dxf
To make the operation of Odeon as smooth as possible it is desirable to move the ***Origo*** of this geometry. In this geometry the front of the stage is located at:
(x,y,z) = (10.5, 0.0, 24.0)



We may want to locate ***Origo*** at the front of the stage. This can be done using the ***Mtranslate*** statement in the geometry file – open the .par file, clicking the Odeon editor icon the just after the ### sign type:

MTranslate –10.5 0 -24

At the end of the file just before the ### sign, type ***MReset*** in order to make the coordinate system neutral – this is desirable when adding new surfaces to the geometry.

Click the Odeon icon inside the editor save the modified geometry and reload it in Odeon.

Other coordinate manipulations to the geometry may desirable; in particular the ***CoordSys*** and ***MRotate*** statements described in section 3.2.1 may be useful.

3.4 Model check in ODEON

The geometry file is the first file used by ODEON when assigning a file from Files|Open Room model. When assigning a new or modified room its validity is checked.
The check performed by ODEON involves checking whether data is consistent and in the correct format, but not whether a meaningful geometry is being defined. If the geometry passes the test you may start checking if the geometry is meaningful and without errors, this may involve:

- Viewing the room in a 3DView
- Viewing the room in the 3DOpenGL display
- Analysing the geometry for unacceptable surface warps, using the 3DGeometry Debugger option.
- Analysing the geometry for unacceptable surface overlap, using the 3DGeometry Debugger option.
- Checking for missing surfaces in the room (forming holes in the geometry). The Unique edge's function available from the 3DView may help you (shortcut E).
- Testing water tightness of the room, tracing rays in the 3D Investigate ray tracing window.

3.4.1 Viewing the room in a 3DView

The 3DView displaying your room once loaded into ODEON has a number of large of facilities which can be useful when creating and verifying geometries for Odeon.

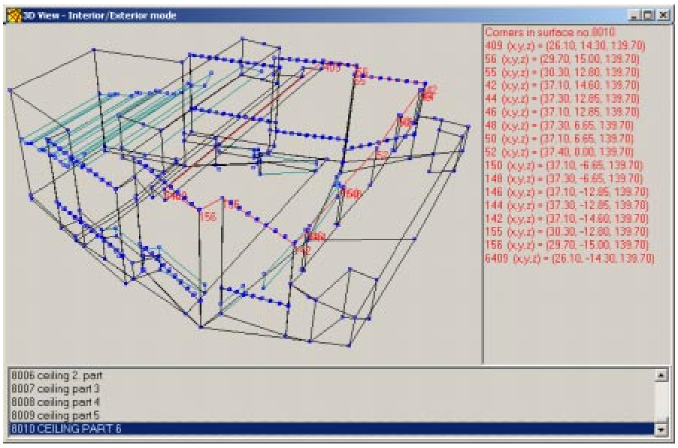


Figure 1 Viewing corners and coordinates in a selected surface using the N shortcut and highlighting corner and displaying the coordinates of the corner closest to the mouse pointer using the M shortcut.

The perspective option (shortcut P) allows you to turn off the perspective of the room, to get an isometric display of the room. This may prove valuable when investigating warped surfaces.

The Unique edge's option in the 3DView display shows edges, which only occur on one surface. Such an edge is "free"; it might be the edge of a free-hanging reflector, but it could be the result of an error whereby two surfaces, which should join along an edge, do not.

> Example:
> Modelling a box shaped room consisting of 6 surfaces, but forgetting to define the 6th surface in the geometry file. This room will have a hole where the 6th surface is missing. The unique edge's option will show where the missing surface should have been.

Using the 3DOpenGl display for model verification

This display is very useful for detecting holes in the geometries. Especially if stepping outside the model (Arrow-back shortcut) and rotating the model using the Ctrl+Arrow shortcut. See the corresponding dropdown menu for more shortcuts. When materials have not been assigned to all surfaces the room, surfaces will appear in random colours making holes appear more visible. If materials have been assigned, the colours will by default reflect the acoustic properties of the surfaces, however it is possible to turn on the random colouring, using the R shortcut.

3.4.2 3DGeometry debugger

Overlapping and warped surfaces should be avoided in the room model specified in the geometry file, but a certain amount of overlap and warp (by default 50 mm) is allowed without generating a warning. By overlapping surfaces is meant surfaces, which define a part of the same plane in space. In the simple case this can be because the surfaces are simply duplicates, another case could be a door, which has been defined in the same plane as the wall in which it is mounted. Overlapping surfaces should be avoided because it will not be clear which absorption coefficient should be applied at a reflection in case of overlapping surfaces.

Warps can lead to "holes" in rooms at edges of joining surfaces, with erroneous results as a consequence and the surfaces will not be well defined.
Using the 3DGeometry debugger in ODEON, ODEON will generate a list of warnings and a corresponding illustration in a 3D display, whenever an overlap or a warp exceeds the value specified in the Room setup|Model/Air conditions dialog.

Overlapping surfaces is a tricky problem because it is usually invisible on 3D projections of the geometries, however such errors in the model may lead to unpredictable results, so *always* check models of some complexity for overlapping surfaces.

3.4.3 Testing Water tightness using 3D Investigate trace rays

Testing a new model for "watertightness" (i.e. whether it is completely closed) may be done using a 3D Investigate trace rays window.

The room model may not be watertight if:

- Surfaces are missing from the model.
- Surfaces are unacceptable warped.
- Surfaces are overlapping.
- Boundary surfaces have been assigned transparency coefficients greater than zero.
- Boundary surfaces have been assigned Material 0 transparent.
- Sources are located outside the room.

Before investigating ray tracing, you will have to:

- Make the boundary surfaces of the room "solid" by assigning materials to them. For the moment it does not matter what the materials are, as long as they are not transparent (Material 0) or fully absorptive (Material 1). Go into the Materials List and assign, e.g. 20% absorption to all surfaces (use Ctrl+Ins to do this in one keystroke).
- Place a source somewhere inside the room. Sources are defined from the Source-receiver List. At first it may be a good idea to define a point source somewhere in the middle of the room.

Open a 3D Investigate ray tracing display and run it with, e.g. 1000 rays, with a Max reflection order of zero. This tests whether any holes can be seen from the source position, and should reveal any gross problems. The tracks of lost rays will show outside the room boundaries, and indicate whereabouts in the room problems occur.
If rays are being lost, and you have an idea of which part(s) of the room is /are "leaky", a number of things may be done:

Reduce the value of Max. accept. warp or the Max. accept. wall overlap in the Room setup at the Model|Air conditions page. Then run the 3DGeomtry Debugger. Warnings will appear if surfaces have a warp or an overlap above the acceptable range. This may reveal slight warps or overlaps in surfaces in the leaky region of the room, which then have to be reduced as far as possible by revisions to the geometry file.

Use the 3DView or 3DOpenGL for inspection of the model to study the region(s) under suspicion. It may turn out that a surface is missing or does not join to its neighbours in the expected manner. It may help to zoom regions in question with the Highlight surfaces, Show corner numbers and coords and Modelling options switched on.

4 Materials

This chapter covers material properties and the facilities available from within the Materials List.

4.1 Special Materials

Material 0 (transparent)
Assigning Material 0 to a surface corresponds to removing the effect of the surface completely from all calculations. Hence surfaces with this material assigned:

- Offer no hindrance to rays, either in energy or direction
- Are excluded from the calculated active surface area of the room, and therefore do not affect the estimate of the room's volume produced by Global Estimate or Quick Estimate Reverberation

This facility can be used to temporarily "remove" surfaces such as doors or reflectors from the room or to define a phantom surface over which an energy map (a grid) is to be plotted.

Material 1 (totally absorbent)
The totally absorbent material (Material 1) may be used for modelling outdoor situations, e.g. an open roof. This is the only material, which will stop the rays during ray tracing and no reflections are generated from surfaces assigned this material.

The material list consists of a window containing two lists, the surface list and the material library. When selecting a surface in the surface list, the surface is automatically highlighted in the corresponding 3D Materials window.

The Material list window consists of two parts:
- Surface List (left part of the window)
- Material List (right part of the window)

Some of the functions available, at the local toolbar as well as from the toolbar dropdown menu:

- **Assign Material**, assigns the material selected in the material list to the surface(s) selected in the surface list.
- **Assign Material for all surfaces**, assigns the material selected in the material list to all the surfaces in the room.
- **Global Replacement** replaces all appearances of the material assigned to the selected surface in the surface list, with the material selected in the material list. This is useful if you wish to replace all appearances of a material with another, e.g. using two layers of plasterboard instead on one.
- **Assigning Scattering coefficient** is done a little different. You simply select the field at the surface and enter the scattering coefficient using the keyboard.
- **Assigning Transparency coefficients**, select field at the surface and enter the transparency coefficient directly using the keyboard.
- **Repeat Scattering coefficient** assigns the scattering coefficient last entered in the surface list to the current selected surface.
- **Repeat Scattering coefficient** for all surfaces, assigns the scattering coefficient assigned to the surface selected in the surface list to all the surfaces.
- **Quick Estimate** for fast evaluation of reverberation times and listing of summarized absorption areas, while assigning materials etc.
- **Edit /Add a material** allow to create new materials or to edit existing ones. The material editor available assists in mixing different materials into one.

Surface List (Material List window)
The surface list, lists the material specifications assigned to the surfaces, starting from the left to the right:

Surface number
The unique number assigned to the surface in the geometry file.

Material number
The number of the material assigned to the surface (from the material library). This number and material correspond to the number listed in the material display, except when:

- The material has been edited in the material library after the material was assigned to the surface (e.g. its absorption coefficients have been changed).
- The materials were assigned on another computer where another material library was available, with different definitions of the material having this number.

Note! Once a surface has been assigned a material, this material sticks to the surface, even though the material has been changed in the material library (Material.Li8), thus calculated results stays in consistency with the materials assigned to the room. To make such a new material take effect in the room **please** reassign the material, e.g. using the Global Replacement option.

Scattering coefficient (or diffusion coefficient)
A scattering coefficient is assigned to each surface. This coefficient is taken into account during the ray tracing if Room setup|Calculation parameters|Scattering method is set to Lambert. The scattering coefficient can be assigned values between 0 and 1 (and -1 and 0):

- 0.1 should be assigned to large surfaces, smooth /rigid surfaces.
- 0.7 should be assigned to very scattering surfaces like the audience area in a concert hall.

In general it is recommended to keep the scattering coefficients between 0.1 and 0.7. For rooms such as offices and class rooms, there will often be a lot of small items which are not accounted for in the geometrical models of the rooms and interior in the rooms will often be modelled as being perfectly parallel or perpendicular, which in the real world they are not. A minimum scattering coefficient of 0.3 is recommended in rooms such as classrooms and offices in order to take into account for these scattering effects.

Transparency coefficient - semi transparent surfaces
A transparency coefficient is assigned to each surface; this is a way to make the surface semi transparent. This feature may be used for modelling many small surfaces in real rooms. E.g. a reflector panel built from many small surfaces with space in-between can be modelled as one large surface having a transparency coefficient of, e.g. 0.5. The transparency coefficient can be assigned values between 0 and 1:

- 1 are assigned to all solid walls. This value should always be assigned to the boundary walls of the room; otherwise rays will escape from the model.
- Very small transparency coefficients, e.g. 0.05 should be avoided. Instead consider modelling the surface as solid. Using a transparency coefficient greater than zero will cause the Image source method to be discarded for rays hitting such surfaces (only relevant for point sources). Another problem is that only very few rays will be transmitted, making the results on the other side of the surface statistically unreliable.
- Very large transparency coefficients, e.g. 0.95 should also be avoided. Instead consider removing the surface from the model. An easy way to do this is to assign Material 0 (transparent) to the surface. Using a transparency coefficient greater than zero will cause the Image Source Method to be discarded for rays hitting such surfaces (only relevant for point sources).

Surface name
Lists the name given to the surface in the surface file (if given any name)

Area
Lists the calculated area for each surface.

4.2 Editing and extending the Material Library

The materials displayed in the left side of the Materials List window resides in an ASCII file called Material.Li8. This library, provided with ODEON, may be altered and extended at will by the user, using the material editor available from the Material list. If you should wish to add several materials e.g. by coping them from some other file, this is possible by editing the file using the OdwEdit editor which is also available from within the Materials list and following the ODEON material format.

Special Materials
There are three special materials in the library

- Material 0, transparent
- Material 1, totally absorbent
- Material 2, totally reflective

Although the material library Material.li8 may be edited, materials 0, 1 and 2 must remain as originally defined.

Data format for materials in Material.Li8
The data format for a material in Material.Li8 is very simple; each material is described by to lines:

ID_Number Descriptive text up to rest of line
α63 α125 α250 α500 α1k α2k α4k α8K

ID_Number must be a unique number between 0 and 2.147.483.647.
Absorption coefficients on second line must be floating point within the range 0 - 1 (the line containing 8 floating point values).

5 Auralization

(Combined and Auditorium editions only)

Although much effort has been made to make it as easy as possible to use the auralization capabilities available in ODEON, its felt that a separate chapter on the use of the auralization in ODEON is needed, as this is where all the threads from room acoustics modelling, signal processing, wave signal files, transducers, psycho acoustics, recording techniques etc. meets.
In the description of auralization techniques special words are frequently used, please refer to appendix C; Vocablary for a short description. In this chapter it is assumed that you have tried the short-guided tour in chapter 2.1.

The basis for auralization in ODEON is the Binaural Room Impulse Responses (BRIR's), which can be calculated as part of the Single Point Response, in the Job List - if the Auralization Setup|Create binaural filters option is turned on. In short terms the BRIR is a two channel filter trough, which a mono signal passes from the sound source(s) to the left and the right ear entrance of the listener (receiver). Using convolution techniques to convolve a mono signal with the BRIR, a binaural signal is obtained which, when played back over headphones should give the same listening experience as would be obtained in the real room. Mixing such binaural signals created with different source positions and signals, but with the same receiver position and orientation, multi channel simulations is possible (e.g. simulating a stereo setup, background noise versus loudspeaker announcements or singer versus orchestra).

Listening to Binaural Room Impulse Responses

As mentioned above the basis of auralization in ODEON is the BRIR's, which are calculated as a part of the Single Point Response. Once a Single point response is calculated, it is possible to play the BRIR, clicking the Play Single Point BRIR button. The BRIR may give a first clue as to how the room sounds and it also allows some evaluation of the quality of the calculated point response, e.g. whether to use a higher number of rays or a higher Reflection density in the Room setup.
Although the BRIR may sound a little ‘rough’, it may work quite realistic when convolved with a signal less transient than an ideal impulse. To get a more realistic presentation of a BRIR as it would sound in the real world you might want to convolve it with the Clapping signal file, an anechoic recording of hands being clapped, which is eventually a less transient signal than an ideal impulse.

A note on equalisation of natural point sources

When using natural point sources in simulations aimed for auralization, one should show caution not to include the overall equalisation of the source (signal) twice in the simulation. E.g. if you use a point source assigned the directivity file TALKNORM (person speaking at a normal level), an overall equalisation for this type of signal is included in the directivity file, however the overall equalisation is also included in the anechoic recording of voice to be used in the auralization. Clicking the Linear Equalisation button in the Point Source Editor, when you define the point source will produce a flat overall equalisation for the source, thus making it suitable for auralization. You may need to make two sets of calculations one with a source, which has been equalised for auralization, and another un-equalised for derivation of parameters.

Head Related Transfer Functions and digital filtering

To create binaural simulations, a set of HRTF's (see Appendix C; Vocabulary) is needed. The HRTF's are different from subject to subject and in principle you may measure your own ones and import those into ODEON using the Tools|Create filtered HRTF menu entry in the ODEON program. Measuring HRTF's is however a complicated task so you will probably be using the supplied ones. If you should be interested in creating own sets of HRTF’s for Odeon, additional information can be found in the help available in the Odeon program.
The imported HRTF's to use for auralization are pre-filtered into octave bands in order to reduce calculation time. The octave band filter parameters for the selected filter bank can be seen on the filter bank name at the Auralization Setup menu. The filter parameters are:

diffuse

If the file name contains the word *diffuse*, the Diffuse field filtering (or equalization) was applied to this set of HRTF's in order to obtain an overall flat frequency response; that is, the average frequency response of all the filters was calculated and all the HRTF 's were filtered with the inverse of this filter. If using headphones, which are diffuse field equalized (most headphones are) or indeed loudspeakers for reproduction of auralized sound then diffuse filtered HRTF 's should be preferred.

M

If the file name contains the 'word' *Mddd* where *ddd* is a floating-point number, is included in the filename, then localization enhancement was applied to the HRTF 's [38]. This means that frequency dip and notches in the individual HRTF's has been exaggerate in order to improve the directional cues in the HRTF 's. The M-factor determines how much the dips and notches has been exaggerated; If M is 0 then the effect is neutral, a M factor of 0.6 is suggested as a good compromise between improved localization and undesired coloration, a M value greater than 1.3 is not desirable.

Sample rate

The sample rate of the HRTF. This sample rate should be the same as the sample rate of the signal files (anechoic recordings) to be used. The supplied HRTF's are sampled at 44100 Hz.

Apass

Ripple of octave band filters in dB. Smaller is better, 0.5 dB is probably sufficient.

Astop

Maximum possible attenuation of octave bands. To allow complete attenuation of all reflections of a 16 bit signal (96 dB dynamic range), A_{stop} should be 96 dB, however due to auditory masking we are not able to hear such differences so 40 dB is probably sufficient. Smaller A_{stop} leads to shorter calculation time (of the BRIRs).

Band overlap in percent

Octave bands implemented using FIR filters are not completely rectangular, it takes some frequency span before they attenuates completely. An overlap between the filters of 100 percent gives a smooth transition between the filters, which is probably a more realistic representation of real world reflections than shorter overlaps. At the same time long, overlap gives shorter calculation time (of the BRIRs).

Points per HRTF's

Length of the individual HRTF's. The supplied filters are 128 points (samples) long, which is around 3 ms at a sample rate of 44100 Hz or 1 metre.

If you should need to filters with other filter parameters e.g. A_{stop} being 96 dB you should create a filtered set of HRTF's with these parameters, use the File|Create filtered HRTF's option. Then from within your room, select the new filter bank from the Auralization Setup.

If you should need to import other HRTF's than the Kemar or CIPIC ones [32,41] supplied with ODEON, you should create a text file following the same format as used in the files Unity.hrtf and Kemar.hrtf. These two files can be found in the \DirFiles\ directory.

Adjusting levels

Sound Pressure Level is one of the most important room acoustical parameters, so it is important that levels at which auralization samples are presented are realistic. If playing a simulation of voice at an unrealistic high level, the speech intelligibility may be over judged - it does not help that Clarity or Speech Transmission Index is satisfactory if the Sound Pressure Level is too low. If play back levels are too high, echo problems may be exaggerated, because echoes that would be below audible threshold (or at a very low level) are made audible.

The levels presented in auralization samples created by ODEON are influenced by:

- The HRTF's
- Level in input signal file, e.g. the RMS value or Leq_A
- Calculated Sound Pressure Level, which is based on geometry, sources, receiver positions, materials etc.
- Overall recording level in the Auralization setup
- Rec. Lev. in the Auralization display of the JobList -if off-line convolution is used.
- Mixer levels (Mix. Lev. in the JobList) if off-line convolution is used.
- Gain in the Streaming convolution dialog if the real-time convolution option is used
- Output gain of the soundcard, the volume setting
- Sensitivity of the headphones
- Coupling between headphone and the subjects ear

Maximised play back levels for maximum dynamic range

If you are only interested in the best sound quality in your auralization files you may focus on getting an Output Level (Out Lev in the auralization display within the Job list) as close to but below 0 dB as possible in the Convolve BRIR and Signal file table and in the Mix convolved wave results into one wave file table in order to obtain the highest dynamic range. If using the Streaming convolution option available from the main display in the Joblist, Odeon will maximize the auralization output level, if changing input signal or BRIR from within this display you may press the Maximize Gain button to maximize the gain for the new setup.

Relative play back levels
In some cases you'll be interested in obtaining correct relative levels e.g. for comparisons between different seats in a concert hall. In this case you should remember to use the same recording level (convolver level and mixer level) in the samples to be compared, it is a good idea to use the same input Signal file to make sure that levels are the same at this point. If you wish to compare across different rooms you should also be careful to remember that source gains in the rooms corresponds. If using the Streaming convolution option available from the main display in the Joblist, Odeon will maximize the auralization output level, so if you wish to compare different setups you should make sure to set the Gain in the Streaming convolution display to the same value.

Absolute play back levels
Setting the level to an absolute level so the subject presented to the auralization sample experiences the same level as would have been the case in the real room is a bit tricky as it involves every part in the signal chain.

To obtain a reasonable correct level a first approach is to adjust the auralization output against levels of some kind of sound in the room in which you are, e.g. if you are simulating voice, try to compare the level of the playback with the level of somebody speaking in your room. This method should make it possible to make a rough adjustment - and it's certainly better than none.

A more precise method is to use the calculated SPL_A as a reference (if it is calculated at an absolute level):

- Present the auralization signal over a loudspeaker in the room in which you are sitting
- Measure the sound pressure level in the room at the position where you will be sitting when listening to the auralization and adjust the output level of the loudspeaker-amplifier until the measured Leq_A corresponds to the calculated level. At this point you have a physical reference level, which can be used for calibration of you auralization playback level
- Change between playing your auralization sample over headphones and over the loudspeaker while adjusting the level of the auralization playback until you are satisfied that the levels are the same.

This method is somewhat inspired by Bachausen and should in principle allow perfect calibration of the level (the resulting level being within one subjective limen).

Headphones
The auralization results created in ODEON are binaural signals which should be presented over headphones, the objective being to reproduce the same sound pressure at the entrance of the ear canals (and at the eardrums for that matter) of the subject as would be obtained in the real room, if it exists).

Soundcards
A sound card is required in order to play back the auralization results and may also be useful if you wish to transfer (anechoic) signals to the harddisk. As a minimum the sound card should be capable of handling signals in stereo, in a 16-bit resolution at a sampling frequency of 44100 Hz. To transfer signals, without loss in quality to /from a DAT recorder the soundcard should be equipped with digital input and output and the soundcard should be able to handle a sampling frequency of 48000 Hz. It should also be considered whether the card is immune to electromagnetic noise, which is always present in a PC and whether its analogue output for headphones is satisfactory.
The soundcard should of course be of a reasonable quality to allow realistic presentations of the simulated room acoustics.

At www.digitalexperience.com/cards.html a listing of soundcards and their manufactures can be found.

Input signals for auralization - anechoic recordings
For auralization you will be using input signals to convolve with the calculated BRIR's. Usually the signals will be anechoic signals although it may also be other types of signals, e.g. if you are simulating an ordinary stereo setup in a room you will probably be using a normal stereo recording. The input signals to be used with ODEON are stored in files following the Windows Wave format (having the extension .wav) in a 16-bit resolution at a 44100 Hz sampling frequency.
The ODEON program comes with a few anechoic samples, which are installed to the \odw...\WaveSignals\ directory. If you wish to extend the library of input signals you should put your new signal files here.
A few audio CD's containing anechoic recordings are commercially available, namely the Archimedes CD [36], which contains some recordings of solo instruments and the Denon CD [37] which contains (semi) anechoic stereo recordings of orchestral music. The easiest way to transfer recordings on an audio CD into wave files on the harddisk on a computer is probably to use a software application called a CD ripper, this

also ensures the transfer is without loss in quality. Signals ‘ripped' from CD audio tracks, will always be in two channels - if you know that signals in fact are mono signals it will be a good idea to convert the resulting wave files into mono signals, this will save space on the harddisk and avoid confusion whether a signal is stereo or in fact mono. A standard wave file editing software, which is usually included with the soundcard should be capable of doing the job.
If you have recordings, which you have created yourself, e.g. using a DAT recorder, you should use a wave file recording software in connection with your soundcard in order to transform the recordings into wave files. Most soundcards comes with a software program for recording and editing wave files, which should be capable of this job. Please note that the connection between the CD-ROM drive and you soundcard is often an analogue one, so if you record from this drive you'll not benefit from digital inputs on your soundcard, resulting in a loss in quality.

Output signals
The output signals are all binaural signals stored in two channel wave files and will have the same leading name as the room. The result files, being in the wave format, makes it easy to edit and publish the results e.g. on the Internet or on audio CD's.
The binaural impulse responses files are created the first time they are played back from within ODEON and will have the extension .Jnn.Wav where nn refer to the relevant job number.
The wave files created as results from the Convolve BRIR and Signal file table, will have the extension .ConvAuralnn.Wav where nn refer to the row in the table (Conv. no.).
The wave files created as results from the Mix convolved wave results into one wave file table will have the extension .MixAuralnn.Wav where nn refer to the row number in the table (Mix. No.).

Publishing audible results on the Internet
To publish calculated demonstration examples on the Internet, it is a good idea to convert the result files into compressed .mp3 files or .wav Mpeg Layer3 files, as download times for wave files are extreme. One minute of compressed stereo signal will (depending on the compression rate) take up approximately 1 MB. To create compressed files, a mp3-encoder software is needed. Shareware versions of mp3-encoders can be found at www.mp3.com. When publishing examples, make sure that copyrights are not violated. You are free to publish examples, which are calculated using the anechoic examples supplied with ODEON, you may also redistribute the same anechoic examples for comparison. Remember to inform the end-user to use headphones when listening to the samples.

Publishing results on an ordinary audio CD
If a CD-R drive is installed on your PC is quite easy to transfer the wave result files into an ordinary audio CD (most CD-R drives comes with the necessary software for this purpose). Most people have access to a CD audio player, so publishing results on an audio CD makes is easy to send demonstrations to clients etc. without worrying about whether they have a PC with a soundcard of a reasonable quality. Again when publishing examples, make sure that copyrights are not violated. You are free to publish examples, which are calculated using the anechoic examples supplied with ODEON, you may also redistribute the same anechoic examples for comparison. Remember to tell the end-user to use headphones when listening to the samples.

6 Calculation Principles

6.1 Global decay methods

ODEON features two methods for calculating the Global decay of rooms:

- Quick Estimate, which is available from the Materials List, is the fastest method allowing quick evaluation of the effect of changes to materials. This method should be considered only as a tool for preliminary results.
- Global Estimate is the most precise of the methods allowing high quality results.

6.2  Quick Estimate

This method estimates a mean absorption coefficient, which is inserted in the Sabine, Eyring and Arau-Puchades formulas to give an estimate of the reverberation time. Instead of simply taking the areas of the surfaces and multiplying by the corresponding absorption coefficients to obtain the total absorption in the room, ODEON also sends out 'particles' from the source, reflecting them in the room using Snell’s law as reflection method (see section 6.4), keeping a count on how many times they hit each surface. Surfaces that are hit very often then carry greater weight in the overall mean absorption coefficient of the room. Surfaces, which are not detected at all in the ray-tracing process, are left out of all calculations and surfaces which are hit on both sides are included twice in the calculation. As a result the estimated reverberation time corresponds to the sub-volume in which the selected source is located. Note however that if a part of the area of a surface, which is present in the sub-volume, is located outside that sub-volume (e.g. if two sub-volumes share the same floor surface) then area and surface estimates for the statistical calculations may not be entirely correct.

The classical mean absorption coefficient is given by:

$$\bar{\alpha} = \frac{\sum_i S_i \alpha_i}{\sum_i S_i}$$

where S_i and α_i are respectively the area and absorption coefficient of the i^th room surface.

The modified mean absorption coefficient as experienced by the particles is:

$$\bar{\alpha}' = \frac{\sum_i H_i \alpha_i}{\sum_i H_i}$$

where H_i is the number of hits on the i^th room surface.

In ODEON, both of these mean absorption coefficients are inserted in the Sabine and Eyring formulae to calculate reverberation times; the classical values are labelled Sabine and Eyring, and the values using the modified mean absorption coefficient are labelled Modified Sabine and Modified Eyring. The mean absorption coefficients used for the Arau-Puchades formula are derived in similar ways except that separate values for surface hits, area and the corresponding mean absorption coefficients are calculated as projections onto each of the main axis of the room.
The Sabine, Eyring and Arau-Puchades formulae require a value for the room volume, which ODEON estimates from the mean free path experienced by ray tracing, using the well-known relation:

$$l = \frac{4V}{S}$$

where V is the room volume and S the total active surface area. From version Odeon 6.5, the ray-tracing process carried out in order to estimate the room volume assumes scattering coefficients of 0 for all surfaces rather than using the coefficients assigned to the surface in the materials list as this is the mean free path formula is based on diffuse field assumptions.

The value of S used here is the sum of the areas of non-transparent surfaces, taking into account whether one, two or indeed none of the sides of a surface are visible inside the room.

Convergence criterion
A certain number of particles must be sent out and followed around the room for a stable estimate to be obtained. More and more particles are sent out in random directions until the value of the reverberation time has remained within 1% for at least 50 particles. At the end of a run, the data on how many times each surface was hit is stored. Then, if new materials are assigned to the surfaces, the reverberation times can be recalculated instantaneously, without repeating the particle tracing.

6.2.1 Global Estimate

This method estimates the global reverberation times T_{20}, T_{30}, the room volume, and the mean free path and generates estimates of decay curves. Particles are sent out in random directions from the source (see section 6.6) and reflected using the 'Late ray' reflection method (see section 6.4). ODEON records the loss of energy in each particle as a function of time occurring because of absorption at room surfaces and in the air. Summing over many particles, we obtain a global energy decay function for the room. The decay curve is backwards integrated and a correction for energy, which is lost due to the truncation of the decay curve, is applied.
This is analogous to an ordinary decay curve, except there is no specific receiver. The summation process may be carried on for as long as desired.

Evaluating results

When the reverberation curve seems smooth, derive the results. If T_{30} values are shorter than T_{20} it is likely that the number of rays used were too small, thus press the Recalculate button. If the reverberation times are 0, the Impulse response length defined at the Room Setup page is properly too short.

6.3 Calculation of Response from Sources to Receivers

This section describes the methods used to predict the response from a source to a receiver, being used to predict Single Point, Multi Point and Grid Response results from the Job list.

Source types - calculation methods
Responses from Point sources are calculated using a hybrid calculation method, where ‘early reflections' are calculated using a mixture of the Image source model and ray-tracing and late reflections are calculated using a special ray tracing process generating diffuse secondary sources. Responses from line and surface sources are carried out using the special ray-tracing method simulating diffuse reflections.

The calculations carried out are divided into a two-step process, a receiver independent part and a receiver dependent part.

Trace-rays - the receiver independent part
Trace-rays is the receiver independent part of the Response calculations; rays are being used to trace down possible reflection paths; the result of this process can be reused for any receiver position in the room. Whenever running a Single Point, Multi Point and Grid Response calculation, the necessary Trace-rays calculation is automatically carried out, if this has not been done already.

Single Point, Multi Point and Grid Response - the receiver dependent part
The response calculations are the receiver dependent part of the calculations; at this point the contributions of direct and reflected sound are collected at the receiving point allowing the calculation of the results known as Single Point, Multi Point and Grid Response.
When more than one receiver is involved, the receiver-dependent part of the process is simply repeated for each source. When more than one source is involved, the response at a given receiver is simply the sum of the responses from the individual sources, each delayed appropriately, if a delay is applied to the source.

ODEON automatically takes care of handling which of the calculation and result files are currently consistent with user-entered data, erasing those that are no longer valid. Thus in some situations you may experience that Trace Rays calculation files have already been done/ are still valid, in other cases they have to be recalculated.

Trace rays - detecting possible reflection paths using ray-tracing
For point sources rays are sent in all directions, for line and surface sources the rays are sent out in random directions following the Lambert distribution (see section 6.6 Sending rays from a source). Once sent from a source, rays are followed around the room as they become reflected and the geometrical data is stored (id numbers of walls hit, points of incidence, etc.). The criterion for stopping the trace of a given ray is normally a geometric one, either the path length travelled (set by the Impulse Response Length) or number of reflections experienced (Max. reflection order). The geometrical data produced, is written continuously to the harddisk, and used later in the determination of reflections received at a point. The early reflections of rays from points sources are treated a little bit different from the rest of the rays because they are reflected speculary, if the reflection order is less or equal to the Transition order allowing the detection of image sources. Above this order the rays are reflected due to the ‘Late ray’ (see section 6.4) method of ODEON.

Determination of Reflections at a Receiver
Having traced rays around the room and stored the data of ray-histories, the next step is to place the receiver at a specific point and so to speak 'collect' the reflections there.

The Early Reflection method

Early reflections in ODEON are reflections generated by point sources while the reflection order is less than or equal to the Transition order specified in the Room setup. The early reflections are treated as modified Image Sources (modified because the limited size of the surfaces as well as scattering properties are taken into account). Every time a ray is reflected at a surface the position of an image source, which may or may not give a contribution to the response at the receiver, is found. The position of this image is defined by the incident direction and the path length travelled from the source to the surface (via other surfaces, in the case of higher order reflections). ODEON checks each image source to determine, whether it is visible from the receiver. Images may be hidden because walls in the room block the reflection path to the receiver or because the receiver falls outside the 'aperture' formed between the image source and the surface generating it. Figure 1 illustrates the concept of visible and hidden image sources. If an image is found to be visible, then a reflection is added to the reflectogram, if another ray has not already detected it. In this 'early' part of the point response calculation, rays are only used indirectly to detect Image Sources that are likely to be valid. From Odeon version 4.2 the Image sources are split into a specular contribution and a 'scattering tree', which consists of secondary sources on the image source surfaces allowing a realistic calculation of early scattering.

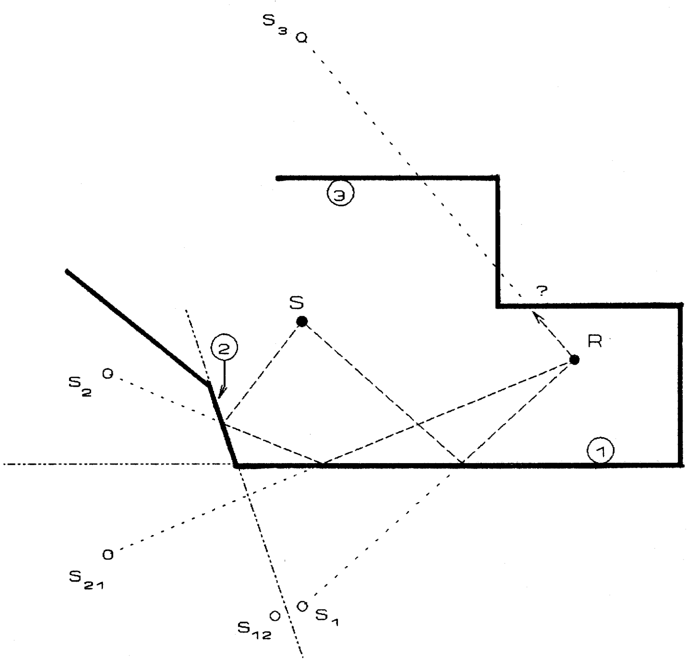


Figure 1 Visible and invisible images. Images S1 and S12 are visible from R, while S2, S3 and S12 are not.

The attenuation of a particular Image Source is calculated taking the following into account:

- Directivity factor of the primary source in the relevant direction of radiation
- Reflection coefficients of the walls involved in generating the image
- Air absorption due to the length of the reflection path
- Distance damping due to the distance travelled from the primary source to the receiver
- Diffraction loss due to limited size of the surfaces generating the reflection and the lengths of the reflection paths etc. for interior surfaces [10].
- Scattering loss, due to the scattered energy which is handled by a 'scattering tree'

Early scattering

In short each time Odeon detects an image source, an inner loop of (scatter) rays (not visualised in the Investigate ray tracing display) is started, taking care of the scattered sound which is reflected from this image source /surface.

Example: If all scattering coefficients in a room is 0.5, then the specular energy of a first order IMS is multiplied (1-0.5) - and the specular energy of a second order IMS is multiplied by (1-0.5)*(1-0.5). The scattering rays handle the rest of the energy.

The early scatter rays are handled in way, which is indeed inspired by the way in which Odeon simulates surface sources, actually each time an image source is detected, Odeon will simulate a surface source, which will emit Number of early scatter rays times the scattering coefficient of the image source surface. The early scatter rays will be traced from the current reflection order and up to the transition order. At each reflection point of the early scattering rays, including the stating point, a secondary scattering source is created.

The Late Reflection method

All reflections that are not treated by the early reflection method are treated by the late reflection method. Every time a late ray is reflected at a surface a small secondary source is generated, having the directivity of $4\cos\theta$ (according to Lamberts Law). ODEON checks each secondary source to determine, whether it is visible from the receiver. The late reflection process does not produce an exponential growing number of reflections (with respect to the time) as would be expected in the real room, but keeps the same reflection density in all of the calculation in order to keep down calculation times.

The attenuation of a Secondary Source is calculated taking the following into account:

- Directivity factor of the primary source in the relevant direction of radiation (point sources only)
- Reflection coefficients of the walls involved in generating the image
- Air absorption due to the length of the reflection path
- Distance damping due to the distance travelled from the primary source to the receiver is inherently included in the ray-tracing.
- Directivity factor for secondary source $4\cos\theta$, due to the Lamberts Law.

Summarising the calculation method used for point response calculations in ODEON
As described above, the point response calculation in ODEON is divided into a receiver independent and a receiver dependent calculation part. The division into two calculations is solely done in order to save calculation time by reusing parts of the calculation where possible.

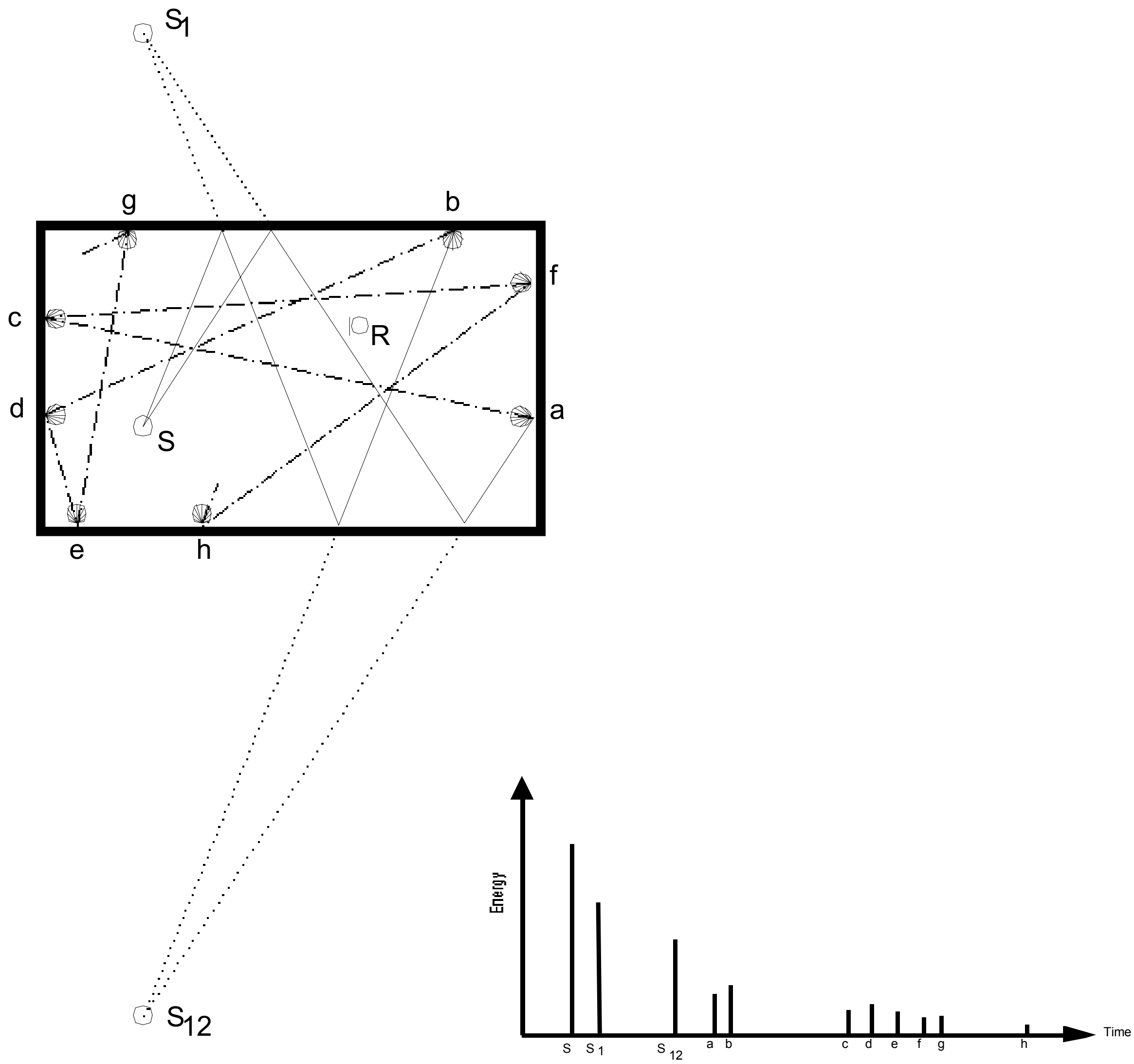


Figure 2 Summary of the model for response calculation. Inset shows resulting reflection sequence at receiver *R*.

Looking at the calculation as a whole, only with respect to one receiver may help understanding the concept. In figure 2 reflections generated by a point source at a certain receiver is illustrated, taking into account only two neighbouring rays up to the sixth reflection order. Because we are dealing with a point source, this figure illustrates the hybrid calculation method. The calculation is carried out using a Transition order of 2 and all surfaces are assigned Scattering coefficients of 1. Thus rays will detect image sources up to second order and above this order they will detect secondary sources.

Both rays detect the image sources, which will both contribute a reflection to the receiver, because the specular reflection path between the source and the receiver is free and reflection points falls within the boundaries of the surfaces.
Although more than one ray detected the image sources they only contribute the detected image sources once, this is obtained by having ODEON build an Image tree keeping track on this. Each Image source being unique is one of the major advantages of the Image Source Model.

Above order 2, each ray generates independent secondary sources situated on the surfaces of the room. The time of arrival of the contribution from a given secondary source is proportional to the ray path length from *S* to the secondary source plus the distance from the secondary source to the receiver. The intensity of a contribution from a secondary source is attenuated as listed above.
One of the advantages of the ray tracing method used in ODEON compared to more traditional methods is that rays does not even have to come near to the receiver to make a contribution. Thus even in coupled

room, it is possible to obtain a reasonably number of reflections at a receiver (which is required to obtain a result that is statistically reliable) with only a modest number of rays. This results in a fine balance between reliability of the calculation results and calculation time.
A complete histogram containing both early and late energy contributions is generated and used to derive Early Decay Time and Reverberation Time. The other room acoustical parameters are calculated on basis of energy collected in time and angular intervals.

For surface and line sources a number of secondary sources are placed randomly on the surface of the source, each emitting one ray and radiating a possible contribution to the receiver. The rays emitted from these source types generate an independent secondary source each time they are reflected. Compared to the calculation principle applied to the point sources one might say that only late energy contributions are collected for these source types or rather that calculations are based on a sort of ray tracing.

Processing reflection data for auralization use in Single Point Response Calculations
A typical point response calculation in ODEON includes some 100000 reflections per source /receiver. The reflections are calculated in terms of time of arrival, strength in 8 octave bands and angle of incidence. The information on size of the reflecting surfaces and absorption coefficients are also available as a part of the calculation. When the Auralization setup|Create binaural impulse response file option is turned on the reflections are post-processed in order to create a binaural impulse response (BRIR). First of all it is determined whether a phase shift should be applied to the reflection, based on surface size and absorption coefficients of the last reflecting surface [31]. Then the reflection is filtered /convolved through 9 octave band filters (Kaiser-Bessel filters, the ninth being extrapolated) and finally the reflection is filtered /convolved through two corresponding directional filters, one for each ear (Head Related Transfer Functions), creating a binaural impulse response for that reflection. This process is carried out for each reflection received at the receiver point and superposing all the reflections, a resulting Binaural Room Impulse Response (BRIR) for that particular receiver point is obtained. The actual order in which the filtering is carried out in ODEON differs somewhat from the description above (otherwise the calculation time would be astronomic), but the resulting BRIR contains the full filtering with respect to octave band filtering in nine bands as well as directional filtering.

6.4 The 'Late ray' reflection method of ODEON

The 'Late ray' reflection method is applied for all rays used in Quick Estimate and Global Estimate. For point response calculations, rays send out from a line or a surface use the 'Late ray' reflection method from the first reflection of a ray (reflection order); rays sent from a point source is handled a little different; in order to combine with the hybrid calculation method, rays are reflected specular as long as the reflection order is less or equal to the Transition order specified at the Room Setup page, this is done in order to allow the detection of image sources up to the specified order, above this order the rays are also reflected using the 'Late ray' reflection method.

The 'Late ray' reflection method is applied in order to allow the inclusion of scattering of sound in the reflections. The scattering method used in ODEON is refereed to as Lambert because it is partly based on Lambert's cosine law; In any direction (θ,δ) the intensity of scattered sound is proportional to $\cos\theta$, i.e. proportional to the projection of the wall area. The Lambert scattering method should always be switched on to obtain optimum results, however it is possible to switch it off at the Room Setup page, in order to illustrate or investigate the effect of scattering.

6.5 Scattering and frequency dependence

When using the Lambert scattering method the scattering coefficients applied to the surfaces of the room from the Materials List are used in the calculations; otherwise reflections are made due to Snells law. The scattering coefficient entered for a particular surface is assumed to be valid for all the frequency bands - this scattering coefficient is the product of scattering due to the surface material itself and due to the geometrical properties of the surface. The geometrical scattering at low frequencies is dominated by diffraction because of the limited size of the surfaces, at high frequencies it is dominated by the irregularities of the surfaces.

Reflecting a 'Late ray'

In ODEON scattering coefficients between 0 and 1 can be applied to surfaces where a scattering coefficient of zero indicates a completely smooth and infinitely large surface reflecting the rays specular, following Snell's[7] law. Having scattering coefficients of one, the surface reflects the ray ideally scattered and the reflected direction is calculated as a random direction following the angular Lambert distribution of ideal scattered reflections; sin2θ.

For scattering coefficients between 0 and 1, the reflected direction is calculated as a weighted direction between the specular and the scattered direction; where a scattering coefficient of zero means that the scattered direction is not taken into account. This is a statistical way of simulating scattering and the methods only holds for a reasonably large number of rays.

From empirical studies it has been found that scattering coefficients should normally be set to around 0.1 for large, plane surfaces and to around 0.7 for highly irregular surfaces (e.g. the audience area in a concert hall). Scattering coefficients as low as 0.02 have been found in studies with extremely smooth surfaces without any diffusing elements. The extreme values of 0 and 1 should be avoided, except for research or demonstration purposes.

6.6 Sending rays from a source

In ODEON (Combined and Industrial version only) there different kind of sources are available the point, the line and the surface source. Knowing a little bit about how ray directions and starting points are generated by ODEON may avoid confusion, and help using tools like Investigate Ray Tracing at its optimum.

Point Sources

For Single, Multi and Grid response calculations and for the investigate ray tracing display, rays are sent in directions distributed as evenly as possible over a solid angle. Ray directions are arranged in rings and ray 1 is sent out almost vertically downwards and the last ray is sent almost vertically upwards. The total number of rays used is usually a few more or less than requested to ensure an even distribution. For Quick Estimate and Global Estimate the send directions are chosen randomly allowing the calculation to be finished after any ray without getting a very uneven distribution of send-directions.

Surface Sources and Line sources

(Combined and Auditorium versions only)

For these source types the send directions and send points are the same no matter the calculation type. For each starting ray a random starting point is chosen at the line or surface source. From this point a ray is send out in a direction following the laws of the 'Late ray' method using a specular direction based on the 'Normal' of the source and a scattered direction. The method used here is similar to the 'Late ray' reflection method of ODEON, however the scattering coefficient used for weighting between the normal direction of the source and the scattered direction is one assign to the particular source from within the appropriate source editor (Line Source Editor or Surface Source Editor). With the present knowledge a scattering coefficient of 1 is suggested for these source types.

6.7 Calculation method for Reflector Coverage

25000 rays are send out from the selected source, if the rays hit one of the surfaces defined as reflector surfaces at the Define reflector surfaces menu, a cross is painted where the reflected rays hits the room surfaces. Note that the value of the Transition order is taken into account; if it is zero and the Lambert scattering is active, the chosen reflectors will exhibit a degree of scattered reflection corresponding to their scattering coefficients. Sound from line and surface sources will always reflect scattered, if the Lambert scattering is on.

---

[7] Snells law is the law of Billiard saying that the reflected angle equals the angle of incidence

7 Calculated Room Acoustical parameters

This chapter will shortly describe the derivation of energy parameters for Single Point, Multi Point and Grid response calculations (for the Industrial edition only EDT, T_{30}, SPL, SPL_A and STI are available). All the parameters are derived on the assumption that the addition of energy contributions from different reflections in a response is valid. This manual will not cover the use of the individual parameters in depth and suggestions on ideal parameters choice should only be sought of as a first offer; instead refer to relevant literature e.g. some of the following references for a further discussion on parameters and design criterions:

Auditorium acoustics as Concert Halls, Opera Halls, Multipurpose halls, etc. are dealt with in [26] and [27], where different halls around the world are presented along with judgement of their acoustics and guidelines for design.

Short guidelines on which values to expect for Clarity and G in concert halls based on some simple design parameters as width, height, floor-slope, etc. are given in [28].

Some recommended values for room acoustical parameters

Objectiv parameter	**Symbol**	**Recommended (symphonic music)**
Reverberation time	T_{30}	1.7 - 2.3 seconds
Clarity	C_{80}	- 1 to 3 dB
Level rel. 10 m free field	G	> 3 dB
Early Lateral Energy Fraction	LF_{80}	> 0.25
Early Support	ST_{early}	> -13 dB
Total Support	ST_{total}	> - 12 dB

Recommended values for objective room acoustical parameters in huge music rooms with audience [24].

Early Decay Time and Reverberation time
The energies of all the reflections received at the receiver point are collected in histograms, with class interval specified in the Room setup |Impulse response resolution. After completion of the response calculation, early decay time and the reverberation time are calculated according to ISO 3382 [6]:

The Reverberation time T_{30} is calculated from the slope of the backwards-integrated octave band curves. The slope of the decay curve is determined from the slope of the best-fit linear regression line between -5 and -35 dB, obtained from the backwards-integrated decay curve.

Early Decay Time (EDT) is obtained from the initial 10 dB of the backwards-integrated decay curve.

Sound Pressure Level, Clarity, Deutlichkeit, LF, ST_{early}, ST_{late} and ST_{total}
The energy of each reflection is added to the appropriate terms in the formulas for all the energy parameters, according to its time and direction of arrival. After the response calculation, Clarity, Deutlichkeit, Centre Time, Sound Pressure Level, Lateral Energy Fraction, ST_{early}, ST_{late} and ST_{total} is derived.

In the following formulae, $E_{a\text{-}b}$ is the sum of energy contributions between time *a* and time *b* after the direct sound, time *t* is the end of the calculated response, and *t* is, for the reflection arriving at time *t*, the angle between the incident direction and the axis passing through the two ears of a listener.

Below two definitions are shown for Clarity, Deutlichkeit, Lateral Energy Fraction, ST_{early}, ST_{late} and ST_{total}, corresponding to the Room Setup|Calculation parameters|Smooth early late ratios option being checked or not. The smoothing is turned on by default. The averaging is equivalent to a 'smoothing' of the transition between 'early' and 'late' energy, and attempts to make up for the facts that:

- Reflections in ODEON are point-like in time, but in reality they are smeared out both physically and by filtering during measurement.
- Inaccuracies in geometrical modelling lead to inevitable displacements of reflections backwards and forwards from their 'true' positions.

Clarity (Early-late averaging OFF):

$$C_{80} = 10\log\left(\frac{E_{0-80}}{E_{0-\infty}}\right) \quad (dB)$$

Clarity (Early-late averaging ON):

$$C_{80} = 10\log\left(\frac{E_{0-72} + E_{0-80} + E_{0-88}}{E_{72-\infty} + E_{80-\infty} + E_{88-\infty}}\right) \quad (dB)$$

Deutlichkeit (Early-late averaging OFF):

$$D = \frac{E_{0-50}}{E_{0-\infty}}$$

Deutlichkeit (Early-late averaging ON):

$$D = \frac{1}{3}\frac{E_{0-45} + E_{0-50} + E_{0-55}}{E_{0-\infty}}$$

Centre time:

$$T_s = \frac{\sum_{0}^{\infty} tE_t}{E_{0-\infty}} \quad (ms)$$

Sound Pressure Level:

$$SPL = 10\log(E_{0-\infty}) \quad (dB)$$

The value of SPL becomes equal to the value of G (the total level re. to the level the source produces at 10 m in free field as defined in ISO 3382 [6]), when an OMNI directional source type and a power of 31 dB/Octave band is selected from within the appropriate Point Source Editor.

Lateral Energy Fraction (Early-late averaging OFF):

$$LF_{80} = \frac{\sum_{t=5}^{80} E_t \cos^2(\beta_t)}{E_{0-80}}$$

Lateral Energy Fraction (Early-late averaging ON):

$$LF_{80} = \frac{\sum_{t=5}^{72} E_t \cos^2(\beta_t) + \sum_{t=5}^{80} E_t \cos^2(\beta_t) + \sum_{t=5}^{88} E_t \cos^2(\beta_t)}{E_{0-72} + E_{0-80} + E_{0-88}}$$

It should be noted that the original definition of 'Lateral Energy Fraction' [6] assumes an ideal microphone having cosine directivity for energy. Real 'figure 8' microphones have cosine directivity for pressure. In order that ODEON's predicted LF_{80} values can be compared with measured results, ODEON uses the modified definition shown above, equivalent to cosine pressure sensitivity. The LF_{80} parameter has a high correlation with the apparent source width (ASW) as shown in [29].

A-weighted Late Lateral SPL (Early-late averaging OFF):

$$LLSPL_{A\ 80}^{\ \infty} = 10\log\left[\sum_{1}^{NBands}\sum_{t=80}^{\infty} E_t \cos^2(\beta_t) \times AWeighting_n\right] \qquad (dB)$$

A-weighted Late Lateral SPL (Early-late averaging ON):

$$LLSPL_{A\ 80}^{\ \infty} = 10\log\left[\sum_{1}^{NBands}\left[\sum_{t=72}^{\infty} E_t \cos^2(\beta_t) + \sum_{t=80}^{\infty} E_t \cos^2(\beta_t) + \sum_{t=88}^{\infty} E_t \cos^2(\beta_t)\right] \times AWeighting_n\right] \quad (dB)$$

The value of $LLSPL_{A\ 80}^{\ \infty}$ becomes equal to the value of Late lateral G (A-weighted total late lateral level re. to the level the source produces at 10 m in the free field), when an OMNI directional source type and a power of 31 dB/Octave band is selected from within the appropriate Point Source Editor. This parameter is suggested in [29] and has a very high correlation with the subjective parameter Listener envelopment (LEV).

Stage Parameters
Stage parameters are calculated as a part of the Single Point response (Auditorium and Combined versions only), if the job only contains one active source, the active source is a point source and the distance between receiver and source is approximately 1 metre (0.9 to 1.1 metre). The parameters are called Support for early, late and total energy and are described in more detail in [24]:

Early Support or ST1:

$$ST_{early} = \frac{E_{20-100}}{E_{0-10}} \quad (dB)$$

Late Support:

$$ST_{late} = \frac{E_{100-1000}}{E_{0-10}} \quad (dB)$$

Total Support:

$$ST_{total} = \frac{E_{20-1000}}{E_{0-10}} \quad (dB)$$

ST_{early} or ST_1 is used as a descriptor of ensemble conditions, i.e. the ease of hearing other members in an orchestra, ST_{late} describes the impression of reverberance and ST_{total} describes the support from the room to the musicians own instrument. If the early late averaging is turned ON, averaging in time is performed as for the other parameters. In case of the stage parameters the following limits of time intervals are used: 9 ms, 10 ms, 11 ms, 18 ms, 20 ms, 22 ms, 900 ms, 1000 ms and 1100 ms.

Warnings displayed with the room acoustical parameters
When the calculated reverberation curves appears very uneven, ODEON may come up with the following warning:

Warning: Fitted reverberation curve not monotonic, results may be unreliable.
This message appears when ODEON is not able to perform smoothing of the decay curve, when the option Room Setup|Smooth late decay is switched on. This may indicate that not enough rays were used in the calculation or that the reflection density was too low. However for large rooms or if the receiver is very close to the source, the fitted decay curve may be non monotonic and in such cases the message should not be taken too serious. It is however not recommended to use the decay curve smoothing – the option is only included for backwards compatibility.

Warning: Direct sound not found, C, D, LF....may not be reliable.
When ODEON calculates the parameters including time intervals in the parameter definition e.g. the C_{80} parameter, the origin of the time axis are set due to the closest source from where direct sound is received. If no source is visible from the receiver or if a hidden source acts significantly earlier at the receiver, the time

origin may come somewhat after the beginning of the actual reflectogram sequence. The warning may of course also indicate an erroneous position of the receiver.

STI - Speech Transmission Index
Speech Transmission Index, known as STI is calculated according to [7]. The STI parameter takes into account the background noise, which may be adjusted from the Room Setup. For the STI parameter to be valid, it is very important to adjust the background noise accordingly, remember that background noise must be set in a relative level if relative source gains are used. It should be mentioned that it is not stated in [7] what kind of directivity the source in the STI measuring system should have, so if using a source with directivity different from the one used in the real measurements in the simulations, results may not be comparable. The subjective scale of STI is given below:

Subjective scale	STI value
Bad	0.00 - 0.30
Poor	0.30 - 0.45
Fair	0.45 - 0.60
Good	0.60 - 0.75
Excellent	0.75 - 1.00

DL_2 - Rate of Spatial Decay
Rate of spatial decay is the decay of sound pressure level per distance doubling. DL_2 is calculated according to ISO 14257 [2]. The DL_2 parameter is intended to characterise the acoustic performance of workrooms. The values to be expected for the DL_2 parameter is according to [1]; 1 - 3 dB for reverberant rooms and 2 - 5 dB for ideally treated rooms. The design criterion for DL_2 is set to 3.5 dB or better according to ISO 11690-1 [3].

The DL_2 parameter is calculated as a part of the Multi Point response, if the job only contains one active source, the active source is a point source, more than one receiver is defined and the distance between the source and the receivers are not the same for all receivers. Please notice that one misplaced receiver may ruin the entire DL_2 calculation, thus it is a good idea to check the receiver positions or even better to check the individual results of the Multi Point calculation.
DL_2 is given for the frequency bands 63 Hz to 8 kHz and $DL_{2,Co.}$, is the A-weighted Rate of Spatial Decay for the frequency bands 125 Hz to 4 kHz. For DL_2, as well as $DL_{2,Co}$ the correlation coefficients are calculated. If the correlation coefficients are low, this may indicate bad locations of source and or receivers, however it may also indicate a very low damping in the room (the Spatial Decay Curve being almost horizontal).
The measuring points (Receiver points) and the source position are of course essential to the DL_2 parameters and should follow ISO 14257 [2]. As an example a path of receivers may be chosen in the following distances from the source (using logarithmic increment):

1, 2, 4, 5, 6.3, 8, 10 metres

The positions should also follow the standard with respect to distance from floor and reflecting surfaces.

ODEON will use all the receivers defined in the receiver list. In some cases the positions of the receivers will not combine with the receiver positions that should be used for the receiver path in the DL_2 calculation. In this case the following solution is recommended:

- Make a copy of the room using the File|Copy files option, e.g. copy a room called MyRoom to MyRoomDL2Path and load the new copy when prompted for during the copy process.
- Delete receivers that are not wanted in the receiver path.
- Define the receivers needed.
- Finally make the Multi Point response calculation with the appropriate point source activated in the particular job.

8 Calculation Parameters - Room Setup and Define Grid

Most calculation parameters are by default set by ODEON, leaving you the choice of the essential parameters such as surface materials, surface scattering coefficients, source and receiver definitions. The only parameter that should always be specified by the user is the Impulse response length. If many surfaces are added to the room model in-between calculations, it is also recommended to re-specify the Number of Rays.

For most of the parameters on the Room Setup page ODEON suggest values that can be considered safe if there are no special demands and the room model does not contain decoupled rooms or very uneven distribution of the absorption area. When this is the case, it may be desirable to increase the Number of Rays (and uncheck Decimate late rays /increase the Desired reflection density).

In some cases it may also be desirable to change certain parameters in order to conduct special investigations or to speed up the calculations in preliminary studies of a room. In either case the parameters are described below.

Scattering method (Job calculations, Global Estimate and Quick Estimate)
If the Scattering Method is set to Lambert, all directions of 'late' reflections are calculated using the scattering coefficients assigned to the surfaces in the Materials List. E.g. if the scattering coefficient is 10 %, the new ray direction will be calculated as 90 % specular and 10 % scattered (random direction due to a Lambert distribution).
If the Scattering method is set to None, scattering is not taken into account, thus all reflections are calculated as specular and if it is set to Full scatter, 100 % scattering is applied to all surfaces. These settings are not a recommended, except for initial tests, demonstration or research purposes.

Decimate late rays (Job calculations only)
For surface and line sources the number of rays is simply decimated, for point sources the rays are decimated above the reflection order set by Transition order. In short terms fewer rays than Number of rays are traced for the late reverberant tail, but still a sufficient number to enable a good estimation of the reverberant behaviour. The reasons for doing this are to enable faster calculations to be carried out without compromising the resolution of early reflections, and to generate smaller ray history files. If you are using ODEON in a research context or if you have rooms with strong decoupling or uneven distribution of absorption area, you might wish to switch this setting off. Decimate late rays is by default on.

Number of Rays (Job calculations only)
The Number of rays to be used for the calculations is automatically set by ODEON; the number is specific for the room loaded and will usually be sufficient for reliable results. The number of rays specified is used for each source in a calculation.
To improve the reliability of the results, increase this number and switch off the Decimate Late Rays option. To decrease the calculation time used for job calculations decrease the number; this may be OK for rough "sketch" calculations.

Max. reflection order (Job calculations only)
Max. Reflection order is a stop criterion, which determines how many times a ray can be reflected. Under normal conditions it should be as big as possible; then the Impulse response length will be the actual stop criterion and Max. Reflection order is only taken into account when stopping rays that has been trapped between two very narrow surfaces. The number may be decreased to speed up calculations if you are only interested in the very early reflections, e.g. if designing the delays of a loudspeaker system. If the Max. reflection order is set to zero, then only direct sound is calculated.

Impulse response length (Job calculations and Global Estimate)
Determine how many milliseconds of the "decay curve" should be calculated. This an important parameter, if it is shorter than approximately 2/3 of the reverberation time, the T_{30} cannot be calculated (because the dynamic range of the decay curve is less than 35 dB). For reliable result it is recommended to use an Impulse response length, which is comparable to the reverberation time.

Impulse response resolution (Job calculations and Global Estimate)

The Impulse response resolution is the width of the steps in the Impulse response histogram in which the energy of the reflections are collected during a point response calculation. The histogram is used for calculation of EDT and T_{30}. A resolution of a approximately 10 ms is suggested.

Transition Order (Job calculations only)
Applies only to point sources. Below the transition order, calculations are carried out using the "Image Source Method", above the transition a special ray-tracing algorithm is used (see section 6.4).
Currently our 'safe' recommendation on the transition order is a transition order of 1. However the transition order can be optimised in order to improve predictions, if the nature of the room geometry and how it has been modelled is taken into account.

Transition order = 0
For rooms dominated by curved surfaces, e.g cathedrals, mosques or rooms modelled using many surfaces. If a transition order of 0 is chosen, then there is no limitation to the size of the surfaces thus domes and other curved surfaces may be subdivided into as many surfaces as desired.

Transition order = 1
Complicated rooms like opera halls, fan shaped concert halls etc. The number of surfaces should not exceed 1000 – 3000 surface.

Transition order = 2
For relatively simple rooms e.g. auditoria, which can be modelled using less than say 50 to 1000 surfaces.

Transition order = 3 or 4
For basically rectangular rooms modelled from no more that 20 to 50 surfaces.

Transition order > 4
A transition order greater the 4 should not be used in order to obtain reliable results. However for special occasions (Auditorium and Combined editions only), you may want to use a higher value of the Transition order parameter, e.g. if examining echo problems as the Reflectogram and 3DReflection paths displays, will only display reflections with a reflection order less or equal to the Transition order. If you have calculated results with the optimum Transition order, which you wish to keep, create a copy of the room with a different name for the investigation, using the Files|Copy files option.

Smooth early late rations (Job calculations only)
A smoothing procedure is normally applied when calculating C_{80}, D, ST_{early}, ST_{late} and ST_{total} and LF_{80}, to simulate the filtering in real measurements as well as the smearing that happens to real life reflections. The Smooth early/late ratios option is by default ON.

Smooth late decays (Job calculations only)
Causes a curve fitting and smoothing procedure to be applied to the reverberant decay, giving better appearance to the late part of the decays. The smoothing is a way of simulating that the number of reflections increases with respect to time, as it would it in real rooms. The Smooth late decays option is OFF by default and doesn't improve the quality of results (except for the visual appearance of reverberation curves).

Desired late reflection density (Job calculations only)
Determines the reflection density, which ODEON will attempt to achieve in the late portion of the decay for Single and Multi Point Response calculations. The higher this value is, the smaller is the chance that unrealistic peaks will disturb the late part of the decay curve, using the default value of 100 /ms will usually be sufficient. To achieve the highest possible density; turn off the Decimate late rays option, use a high number of rays and a high Desired reflection density. You will find a separate value for the Desired late reflection density on the Define Grid page, which is used for the grid response calculations, as it is likely that you will want to speed up grid calculation.

9 Achieving good results

The following section discusses how to obtain good results and indeed what is a good result. It is not a straight answer as to how the best result is obtained, merely a discussion that may provide some ideas as to what can be done in order to obtain reliable results in a program such as ODEON.

The desirable precision - subjective limen
Before discussing how to achieve good results, it is a good idea to outline just what a good result is. The subjective limen (or just noticeable difference - *jnd*) on room acoustical parameters should give a good suggestion as to the desirable precision. If the error between the 'reel' (measured with some precision) and the simulated room acoustical parameter is less the one subjective limen, then there is no perceivable difference and the result is really as good as can be, so it would be senseless to look for more precise results. In many cases it will be difficult or even impossible to obtain results at this precision and a poorer one will probably also be satisfactory for most purposes.

Parameter	**Definition (ISO/DIS 3382)**	Subj. limen
T_{30} (s)	Reverberation time, derived from –5 to – 35 dB of the decay curve	5 %
EDT (s)	Early decay time, derived from 0 to – 10 dB of the decay curve	5 %
D_{50} (%)	Deutlichheit (definition), early (0 - 50 ms) to total energy ratio	5 %
C_{80} (dB)	Clarity, early (0 – 80 ms) to late (80- ∞) energy ratio	1 dB
Ts [ms]	Centre time, time of first moment of impulse response or gravity time	10 ms
G (dB)	Sound level related to omni-directional free field radiation at 10 m distance	1 dB
LF (%)	Early lateral (5 – 80 ms) energy ratio, $\cos^2$(lateral angle)	5 %
STI (RASTI)	Speech Transmission Index	0.05

Room acoustical parameters and their subjective limen as given by Bork [39] and Bradley [40].

Example 1:
If the real *G* value is 1 dB and the simulated is 1.9 dB then the difference is not noticeable.

Example 2:
If the real *LF* value is 12 % and the simulated is 16 % the difference is just noticeable.

Note! When comparing measured parameters to the ones simulated it should be kept in mind that the measured parameters are not necessarily the true ones as there are also uncertainties on the measured results as well. These errors are due to limited tolerances in the measuring equipment as well a limited precision in the algorithms used for deriving the parameters from the measured impulse response (or similar errors if results are not based on an impulse response measuring method). There may also be errors due to imprecise source and receiver positions.

9.1 Sources of error

There are many sources of errors in a room acoustical simulation, leading to results, which are less than perfect (within one subjective limen). Sometimes this is quite acceptable because we are just interested in rough results, at other times we are interested in results as good as possible. In any case being aware of the sources of error may help getting the maximum out of ODEON. The sources of error (or at least some of them) are:

- The approximations made in the ODEON calculation algorithms
- Inappropriate calculation parameters
- Material /absorption coefficients are imprecise
- Material /scattering coefficients are imprecise
- Geometry definition may not be accurate
- The measured reference data to which simulations are compared may not be accurate

9.1.1 Approximations made by ODEON

It should be kept in mind that algorithms used by a program such as ODEON are but only a raw representation of the real world. In particular, the effect of wave phenomena are only to a very little extend included in the calculations. There is very little to do with this fact for you the user, except to remember that small rooms and rooms with small surfaces are not simulated at high precision.

9.1.2 Optimum calculation parameters

A number of calculation parameters can be specified in ODEON. These setting may reflect reverberation time, a particular shape of the room or a trade of between calculation speed and accuracy.

Decimate late rays
To use all the reflections found in the ray-tracing process; the Decimate late rays option should be switched off and the Late ray density should be set to its maximum.

Number of rays
ODEON by default specifies a suggested number of rays to be used in point response calculations. This number is derived taking into account the aspect ration of the room as well as the number of surfaces in the geometry. In short this means that Odeon will suggest more rays for very long room with many surfaces, than for a basically cubic room with few surfaces. This suggested number of rays will be sufficient for many rooms, however in some cases more rays may be needed in order to obtain good results, in particular in rooms with:

1) Strong decoupling effects
2) Very uneven distribution of the absorption in the room, in particular if the quantity of absorption is very different in the x, y and z dimensions

From 1)

If a dry room is coupled to a reverberant room, then more rays may be needed in order to estimate the coupling effect well. An example could be a foyer or a corridor coupled to a classroom. If the room where the receiver is located is only coupled to the room where the source is located through a small opening, then more rays are also needed.

Form 2)
In some rooms the reverberant field in the x, y and z dimensions may be very different. An example of this could be a room where all absorption is located on the ceiling while all other surfaces are hard. Another example could be an opera theatre.
In particular if surfaces are all orthogonal while having different materials in the x, y and z dimensions of the room and if low scattering properties on the surfaces are used, then more rays should be used.

More rays needed?
There are no way of telling if more rays are needed for a certain calculation, but to get an idea whether a room has strong decoupling effects, you may try to run the Global Estimate calculation. If:

- Global Estimate coverage slowly
- The Global decay curve make sudden jumps, like steps on a stair
- The Global decay show ‘hanging curve’ effect

This could be an indication that more rays are needed. Let the Global Estimate run until the decay curve seems stable, then use say 1/10 – 1 times the number of rays used in the Global estimate to specify the number of rays to be used in the calculation of the point responses (specified in the room setup).

Transition order
The ‘Transition order’ can be optimised taking into account the basic properties such as room shape into account, se section 8 for suggestions on Transition order.

9.1.3 Materials /absorption data

Wrong or imprecise absorption data are probably one of the most common sources of error in room acoustical simulations. This may be due to lack of precession in the measurements of the absorption data or because the material construction assumed in the simulations are really based on guesswork – in any case it is a good idea to remember this and to estimate the size of error on the material data as well as the impact on the simulated results.

Solution if materials data are uncertain:

There is really not much to do about the uncertainty of material data if the room does not exist except taking the uncertainty of the materials into account in the design phase. If the room does indeed exist and is being modelled in order to evaluate different possible changes it may be a good idea to tweak (adjust) such uncertain materials until the simulated room acoustical parameters fits the measured ones as good as possible.

Absorption properties in a material library are often, by users, assumed to be without errors. This is far from being the truth. For high absorption coefficients and high frequencies the values a probably quite reliable however; low frequency absorption data and absorption data for hard materials will often have a lack of precision.

Low frequency absorption
At low frequencies the absorption coefficients measured in a reverberation chamber are with limited precision because:

- There are very few modes available in a reverberation chamber at lowest frequency bands.
- Low frequency absorption occurs partly due to the construction itself (e.g. the large wall has the low frequency absorption it has because it is large) rather than is surface structure, and its not possible to reconstruct a complete building construction is a reverberation chamber.

There is no current solution to these problem, but one can hope that new measuring techniques will to some extend overcome these problems.

Hard materials
Hard materials such as concrete are often listed as being 1 % or 2 % absorbing. It may sound like a difference of 0.5 % or 1 % is not a significant difference. However if a room is dominated by this material (or if one of the dimensions of the room is) a change from 1 to 2 % is a relative change of 100 %.

9.1.4 Materials /scattering coefficients

The knowledge on scattering coefficients is currently rather limited. Hopefully in the future, the scattering coefficients will be available for some materials. Meanwhile the best that can be done is to make some good guesses on the size of the scattering coefficients and to do some estimates on the effect of uncertainty.

9.1.5 Measurements

Eventually the reference data, which you may compare with simulated room acoustical parameters are not perfect. We must accept some tolerances on the precision of the measured parameters.

9.1.6 Receiver position(s)

Common errors are:

- to base the room acoustic design on simulations in one or only few receiver positions
- to place the receiver close to a surface.
- to use too short source-receiver distance

9.1.7 Source-Receiver distance

Point response calculations made in ODEON are to be compared with point response measurements and as such the ISO 3382 standard should be followed:

To obtain good estimates of reverberation time, the minimum source-receiver distance should be used in order to avoid strong influence from the direct sound. The minimum source –receiver distance according to ISO 3382 is:

$$d_{\min} = 2\sqrt{\frac{V}{c*T}}$$

where :

V	is the volume of the room in cubic metres
c	is the speed of sound, in metres per second
T	is an estimate of the expected reverberation time, in seconds

Thus for a typical concert hall a source-receiver distance less than 10 metres should be avoided in order to get good predictions (measurements) of the reverberation time.

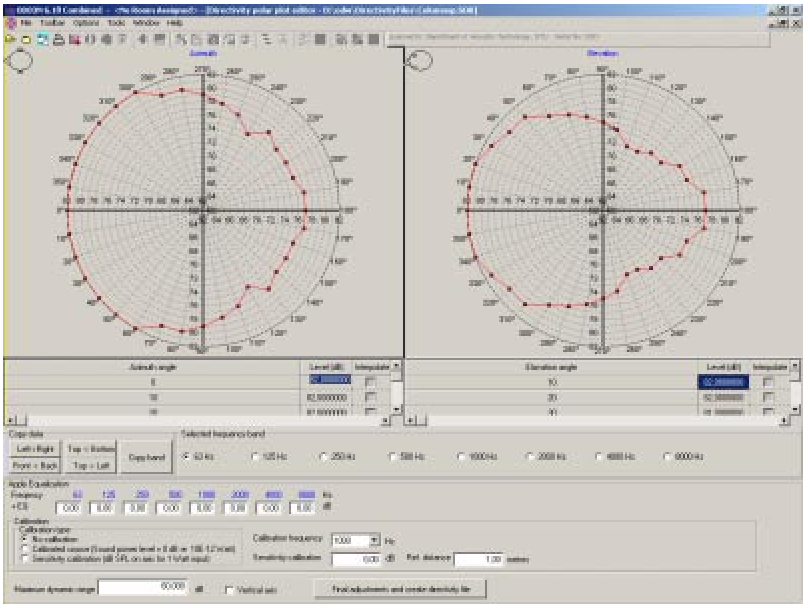
9.1.8 Minimum distance from the receiver to the closest surface

If a receiver is place very close to a surface then results will be sensitive to the actual position of the secondary sources generated by ODEON. If such a secondary source happens to be very close to the receiver e.g. 1 to 10 centimetres this may produce a spurious spike on the decay curve, resulting in unreliable predictions of the reverberation time – indeed if the distance is zero then in principle a contribution being infinitely large would be generated. To avoid this problem is recommended that distances to surfaces are kept greater than say 0.3 to 0.5 metres. Anyway for measurements it is, for other reasons, recommended to keep distances greater than a quarter of a wavelength, i.e. 1.3 metres at 63 Hz – a distance of 1 metre is required by ISO 3382.

10 Creating new directivity patterns for point sources

Tools for creating directivity patterns can be found at the Tools|Creating directivity patterns menu entry inside the ODEON program. The tools allow you to expand the set of source directivity pattern files available for point sources in ODEON. The ODEON directivity pattern file (Version 3 or later) contains information on the sound levels for the eight frequency bands 63 Hz to 8 kHz in dB for each 10° azimuth and 10° elevation. These files are binary and have the extension .SO8. An example on a directivity pattern is the pattern stored in OMNI.SO8.

10.1 Entering a directivity plot using the Directivity plot editor



The easiest way to enter a new directivity plot is to use the built-in plot editor which allows building a directivity plot from a vertical and a horizontal plot.
Enter the dB values for the horizontal and vertical plots at the selected frequency band in the corresponding tables. The angular resolution is 10º degrees. It data are not entered for all angles e.g. if the data are not available, Odeon will do interpolation between the angles entered. For angles between the polar plots, Odeon will perform elliptical interpolation.

Calibration
Three different options are available:

- If No calibration is selected, Odeon will use the dB values as entered in the table, adding the equalization values entered.
- If Calibrated source is selected Odeon will add the equalization entered, then shift the resulting SPL's of the source in order to obtain a sound power level of 0 dB re. 10E-12 Watt at the selected Calibration frequency band. This calibration type is typically used for generic source types such as the OMNI or SEMI directivity pattern.
- If Sensitivity calibration is selected the SPL's of the source will be shifted in order to obtain the SPL is Sensitivity calibration dB on axis of the source at the distance specified as Ref. distance metres.

Maximum dynamic range
The 'minimum level' will be 'Max level' minus 'Maximum dynamic range'. If the range is large the display may not be optimum in the directivity viewer. The source will usually have its max level at its polar axis.

10.2 Creating a new directivity pattern using a text file as input

Another way to create new patterns is to enter the data describing the directivity pattern into an text input file. Depending on the data available and the complexity of the source one of three different text formats may be used. Once the text input file has been created in one of the formats specified below (e.g. in the Odeon editor; OdwEdit), it can be translated into an ODEON Directivity file, which can be applied to any point source from within ODEON.

To translate the created text file into an ODEON directivity file:

- Select Tools|Create directivity (.So8) from ASCII file (.DAT)
- Open the input file you have created.
- Specify the name of the directivity file pattern you wish to create.
- Select whether you wish a Calibrated source or not.
- Apply calibration data as prompted for.

Applying Calibration
Creating a new directivity file you will be prompted whether to create a calibrated source or not:

Calibrated source (Sound Power Level = 0 dB re 10E-12 W at 1 kHz) ? [YES / NO]

Calibrated Sources
Press [YES] if an absolute level is not relevant to the directivity pattern; an example on this could be the OMNI or SEMI directional directivity pattern. When selecting a calibrated source, no data apart from the ASCII input file are required. The directivity represented by this file is preserved, but the values are simply shifted by a constant amount (the same for all bands), such that the sound power level of the source is 0 dB re. 10-12 Watts at 1 kHz. Please do note that the power in the other bands may differ from 0 dB. You may still alter the overall power response of the source by applying an EQ, however the power at 1 kHz will always end up as 0 dB, the other bands shifted accordingly.

NON-calibrated sources - Electro acoustical sources, machinery, natural sources etc.
Press [NO] to preserve the sensitivity of an electro acoustical source or the absolute level of natural source, e.g. a human voice. When selecting the NON-CALIBRATED source you are allowed to enter equalising, electric losses (zero for natural sources) and a sensitivity at a selected frequency band (zero for natural sources).
The addition of electrical sensitivity, electrical input power and electrical loss values completes the data necessary to generate a source directivity file directly readable by ODEON 3 or later.
The DirectivityFilesUtillity program can perfectly well be used to generate a .SO8 file for an electro-acoustic source. Various approaches may be used. In the simplest case, the ASCII input file should contain relative SPL values as for an electro acoustic source. The sensitivity should be given as the true SPL obtained for the calibration band (e.g.1 kHz) at a 1-metre distance (transformed to that distance if necessary using the 6 dB per doubling of distance rule). The EQ values should all be zero. Such a .SO8 file can be used without further Gain or EQ settings in ODEON. In certain cases, the form of the available data or the mode of usage may make this approach inappropriate. Sensitivity and EQ within DirectivityFileUtility.exe, and Gain and EQ within ODEON may be combined in many ways to achieve the desired result.

Text format
The data presented to the Odeon should be in relative calibration across frequency, but need not be in any absolute calibration (this calibration is applied from within the program). Thus the data for the forward on-axis direction constitutes a relative frequency response for that direction, which is used to calculate the frequency-dependency of the source's on-axis sensitivity.

The first non-comment line of the input file indicates whether the data is for:

- FULL set, for complex sources where directivity data is known for each 10° Azimuth and 10° Elevation.
- SYMMETRIC set, for symmetric sources, e.g. a trumpet.
- POLAR set containing only horizontal and vertical polar plots, for sources where only a horizontal and a vertical plot are known, e.g. a loudspeaker.

Each of the subsequent lines of the input file should contain sound levels in dB for a complete 180° of elevation (from the forward axis to the backward axis). The resolution must be 10°, hence each line contains 19 values (0°, 10°, 20°.....160°, 170° and 180°). Comment lines are allowed anywhere in the ASCII input file(s).

When only horizontal and vertical polar plots are known (POLAR)
The first non-comment line of the file should start with the word POLAR. In the polar case, there are four lines of data for each frequency band. The first four lines are for 63 Hz, the next four for 125 Hz, and so on.

For a given frequency, the first and last values must agree on all four lines, since all the polar plots meet at the polar axe. The first line of a group of four is the upward vertical polar plot as seen from in front of the source (12 o'clock plot). Then come the left horizontal plot (9 o'clock plot), downward vertical plot (6 o'clock plot) and finally the right horizontal plot (3 o'clock plot).

As a minimum there must be 1 + 4 * 8 lines in a polar input file.

Elliptical interpolation
When the DirectivityFileUtility translates the polar input file it has to interpolate values between the four polar planes given in the input data. This is done using elliptical interpolation independently for each frequency band, creating the 8 x 4 plots missing between the four input plots.

An example; Polar_Omni.dat on the polar input format can be found in the DirFiles directory, created at the installation of ODEON.

When the complete directivity characteristics are known (FULL)

The first non-comment line of the file should start with the word FULL. In the full case, there are 36 lines of data for each frequency. The first 36 lines are for 63 Hz, the next 36 lines for 125 Hz, and so on.

As a minimum there must be 1 + 36 * 8 lines in a full input file.

- 1. line is vertical upper plot 0° (12 o'clock plot, when looking at the source, e.g. at a loudspeaker membrane)
- 10. line is horizontal left plot 90° (9 o'clock plot)
- 19 line is lower vertical plot 180° (6 o'clock plot)
- 28. line is right horizontal plot 270° (3 o'clock plot)

An example; Full_Omni.dat on the full input format can be found in the DirFiles directory, created at the installation of ODEON.

When the directivity pattern is rotationally symmetric
The first non-comment line of the file should start with the word SYMMETRIC. In the SYMMETRIC case, there is one line of data for each frequency.

As a minimum there must be 1 + 8 lines in a symmetric input file.

Examples of SYMMETRIC sources are a Trumpet and the Omni directional source.

An example; Symmetric_Omni.dat on the SYMMETRIC input format can be found in the DirFiles directory, created at the installation of ODEON.

Samples on directivity patterns (TLKNORM, TLKRAISE and Soprano ref. 42)
The TLKNORM source type corresponds to a male talker with a normal vocal effort. The gain and EQ fields in the Point source editor (inside ODEON) should be set to zero. This source is also a reasonable approximation to a female talker, except that the 63 and 125 Hz band should be ignored.
To simulate a trained talker addressing an audience in a raised voice, use the TLKRAISE source. This has the same directivity as TLKNORM, but the levels in the eight octave bands are respectively 2, 2, 5, 7, 9, 8, 6 and 6 dB higher. The directivity pattern of Soprano ref. 42 is the directivity of a soprano singing opera [42].

Comments and empty lines
Lines containing comments, and empty lines, may be inserted anywhere in the file, as long as they do not come between data items, which should occur on one line. Comment lines must begin with a colon (:), an semicolon (;) or an asterisk (*).

10.3 Compatibility with previous versions of ODEON

The directivity files of ODEON 2.6D and earlier (having the file extension .SOU) is based on the six octave bands (125 to 4000 Hz), where ODEON version 3 or later is using the eight octave bands (63 to 8000 Hz). Thus, new files have to be created containing the information for the eight bands. The levels for 63 and 125 Hz will be equal (just copied) and the levels for 4 and 8 KHz will be equal (just copied).

To create a directivity pattern for ODEON 3 or later (e.g. OMNI.SO8) from an ODEON 2.6D directivity file (or earlier):

- Select File|Translate 6 band into 8 bands.
- Open the old directivity file (e.g. OMNI.SOU)

The new directivity pattern file OMNI.SO8 will automatically be saved at the default path for directivity pattern files (e.g. C:\ODW6_0\Dirfiles). This path is specified from within the ODEON program (at the toolbar dropdown menu: Options|Program setup).

9.3 Making a readable text file from a directivity pattern file (.SO8)

To read the contents of a directivity file:

- Select File|Create ASCII file from .SO8 file.
- Open the .SOU file you which to read.
- Open the newly created ASCII file (e.g. C:\ODW\DirFiles\Omni.Asc) using a text editor like NOTEPAD.

This may be useful for instance to see how the interpolation of the polar plots worked out.

The generated ASCII output file will use the separating character specified from within the ODEON program (OdwCombined.Exe, OdwAuditorium.Exe or OdwIndustrial.Exe) from the dropdown menu at Options|Program setup|ASCII

output. The character will be inserted between each value in the output file. The default separator is a single space.
The file generated is an ASCII file containing the values in SPL at 1 metre defined in the input file. The format of the file is the same as that of an input file of the FULL type.

Appendix A: Mathematical expressions available in the .Par modelling format

Constants, variables, point numbers, surface numbers and coordinates may be defined using mathematical expressions. Where integer numbers are expected (Counter ranges in for..end loops, point, surface numbers, etc.), the results of mathematical expressions are automatically rounded to the nearest whole number.

Operation	Syntax	Example
Addition	+	2+5 = 7
Subtraction	-	3-1 = 2
Multiplication	*	2*3 = 8
Division	/	4/2 = 2
Power	Base^ Exponent or Power(Exponent,Base)	2^3 = 8 or Power(3,2) = 8
Root	Root(Y,X)	Root(3,8) = 2
Round	Round(X)	Round(2.67676) = 3
Truncation	Trunc(X) or Int(X)	Trunc(1.7) = 1
Sine of an angle in radians	Sin(X)	Sin(0) = 0
Cosinus of an angle in radians	Cos(radians)	Cos(PI/4)= 0,707106781186547573
Tangens of an angle in radians	Tan(radians)	Tan(PI/4) = 1
Cotangens of an angle in radians	Cotan(radians)	Cotan(180) = 0
Hyperbolic Sine to angle in radians	Sinh(radians)	Sinh(0) = 0
Hyperbolic Cosine to angle in radians	Cosh(radians)	Cosh(0) = 1
Sine to angle in degrees	SinD(radians)	SinD(90) = 1
Cosine to angle in degrees	CosD(degrees)	CosD(0) = 1
Tangens of an angle in degrees	TanD(degrees)	TanD(45) = 1
Cotangens of an angle in degrees	CotanD(degrees)	CotanD(90) = 0
Inverse Sine in radians	ArcSin(Y)	ArcSin(-Sqrt(2)/2)*180/PI = -45
Inverse Cosine in radians	ArcCos(X)	ArcCos(Sqrt(2)/2)*180/PI = 45
Inverse Tangens in radians	ArcTan(Y)	ArcTan(1)*180/PI= 45
Inverse Tangens II in radians	ArcTan2(X,Y)	ArcTan2D(1,-1)*180/PI = - 45
Inverse Sine in degreess	ArcSinD(Y)	ArcSin(-Sqrt(2)/2)*180/PI = -45
Inverse Cosine in degrees	ArcCosD(X)	ArcCos(Sqrt(2)/2)*180/PI = 45
Inverse Tangens in degrees	ArcTanD(Y)	ArcTan(1) = 45
Inverse Tangens II in degrees	ArcTan2D(X,Y)	ArcTan2D(1,-1) = - 45
Exponential	Exp(X)	Exp(1)= 2.71828182845904509
Natural Logarithm	Ln(X)	Ln(2.718281828459045091) = 1
Logarithm base 10	Log10(X)	Log10(100) = 2
Logarithm base 2	Log2(X)	Log2(8) = 3
Square	Sqr(X)	Sqr(2) = 4
Square root	Sqrt(X)	Sqrt(2)= 1.41421356237309515
Radius	Radius(A,B)	Radius(3,4) = 5
Absolute value	Abs(X)	Abs(-2342) = 2342
Sign	Sign(X)	Sign(-2) = -1; Sign(0) = 0; Sign(3) = 1
Minimum number of two munbers	Min(X,Y)	Min(23,12) = 12
Maximum of two numbers	Max(X,Y)	Max(23,22) = 23

Appendix B: References

[1] Ondet A.M. and Sueur, J., Development and validation of a criterion for assessing the acoustic performance of industrial rooms, J. Acoust. Soc. Am. 97 (3), March 1995 p 1727 - 1731.

[2] ISO/ DIS 14257:1999, Acoustics - Measurement and modelling of spatial sound distribution curves in workrooms for evaluation of their acoustical performance.

[3] ISO 11690-1, Acoustics:1996 - Recommended practice for design of low-noise workplaces containing machinery - Part 1: Noise control strategies.

[4] ISO 11690-2:1996, Acoustics - Recommended practice for design of low-noise workplaces containing machinery - Part 2: Noise control measures.

[5] ISO/TR 11690-3: 1997, Acoustics - Recommended practice for design of low-noise workplaces containing machinery - Part 3: Sound propagation and noise predictions in workrooms.

[6] ISO 3382:1997 Acoustics - Measurement of the reverberation time of rooms with reference to other acoustical parameters.

[7] CEI/IEC publication 60268-16 first edition:1998. Sound system equipment, part 16: Objective rating of speech intelligibility by speech transmission index.

[8] G.M. Naylor & J.H. Rindel, Odeon Room Acoustics Program, Version 2.5, User Manual. Publication No. 49, The Acoustics Laboratory, Technical University of Denmark, Lyngby, 1994. (103 pages).

[9] J.H. Rindel, Computer Simulation Techniques for Acoustical Design of Rooms. Acoustics Australia 1995, Vol. 23 p. 81- 86.

[10] J.H. Rindel, Computer simulation techniques for the acoustical design of rooms - how to treat reflections in sound field simulation. ASVA 97, Tokyo, 2- 4 April 1997. Proceedings p. 201-208.

[11] Cremer, L. and H. Mller, 'Principles and Applications of Room Acoustics', Applied Science Publishers, London, 1982.

[12] Barron, M. and A.H. Marshall, "Spatial Impression Due to Early Lateral Reflections in Concert Halls: The Derivation of a Physical Measure", Journal of Sound and Vib., 77, pp 211-232, 1981.

[13] Kristensen, J., "Sound Absorption Coefficients - Measurement, evaluation, application", Note no. 45, Statens Byggeforskningsinstitut, H rsholm, 1984 (in Danish).

[14] Knudsen, V.O. and C.M. Harris, 'Acoustical Designing in Architecture', John Wiley, 1953.

[15] Bobran, H.W., 'Handbuch der Bauphysik', Verlag Ulstein, Berlin, 1973.

[16] Ingerslev, F, Lærebog i bygningsakustik for Ingeniører', Teknisk forlag, Copenhagen, 1949.

[17] Petersen, J., 'Rumakustik', SBI-anvisning 137. Statens Byggeforskningsinstitut, Hrsholm, 1983.

[18] Parkin, P.H., H.R. Humphreys and J.R. Cowell, 'Acoustics, Noise and Buildings', Faber and Faber, London,1979.

[19] Fasold, W. and H. Winkler, 'Bauphysikalische Entwurfslehre, Band 4: Bauakustik', VEB Verlag f r Bauwesen, Berlin, 1976.

[20] Meyer, E., D. Kunstmann and H. Kuttruff, 'ber einige Messungen zur Schallabsorption von Publikum', Acustica 14, 119-124, 1964.

[21] Beranek, L.L., 'Music, Acoustics and Architecture', John Wiley, 1962.

[22] Ingerslev, F. and J. Petersen, 'Lydabsorberende Materialer', Arkitektens Ugehefteno. 3, 1953.

[23] Dührkop, H. et al, Mørtel, muring, pudsning. Teknologisk h ndbog', SBI anvisning 64, 2nd. edition, Statens Byggeforskningsinstitut, Hrsholm, 1981.

[24] Gade, A.C., Rumakustisk måleteknik, særtryk 14 til kursus 5142. Department of Acoustic Technology, Technical University of Denmark, Lyngby, 1990.

[25] Gade, A.C., 'Practical Aspects of Room Acoustic Measurements on Orchestra Platforms', ICA 14 Proceedings Vol. 3, paper F3-5, Beijing 1992.

[26] Beranek, Leo, How they sound, Concert and Opera Halls. Acoust. Soc. of Am. 1996.

[27] Barron Michael, Auditorium Acoustics and Architectural Design. E& FN Spon, and imprint of Chapman & Hall, 2 - 6 Boundary Row, London SE18HN, UK 1993.

[28] Gade, A.C., The Influence of basic design variables on the acoustics of concert halls; new results derived from analysing a large number of existing halls. Proceedings I:O:A. Vol 19 Part 3 (1997).

[29] Bradley John S., Gilbert A Soulodre. The influence of late arriving energy on spatial impression, Acoust. Soc. Am. 97 (3), April 95 p 2263 - 2271.

[30] Leo L. Beranek, Takayuki Hidaka. Sound absorption in concert halls by seats, occupied and unoccupied, and by the hall's interior surfaces, Acoust. Soc. Am. 104. December 1998 p 3169-3177.

[31] Rindel, J.H, The Acoustics Laboratory, Technical University of Denmark, Modelling the Angel-Dependent Pressure Reflection Factor. Applied Acoustics, 38 p 223 - 234.

[32] Bill Gardner and Keith Martin. MIT Media Lab., HRTF Measurements of a KEMAR Dummy-Head Microphone, http://sound.media.mit.edu/KEMAR.html.

[33] William H. Press Brian P. Flannery, Saul A. Teukolsky, William T. Vetterling, Numerical recipes in Pascal, The Art of Scientific Computing, Cambridge University Press 1990.

[34] Affronides, Sophocles, Introduction to Signal processing, J.Prentice Hall International, 1996.

[35] Oppenheim, Alan V. Schafer, Ronald W., Descrete-Time Signal Precessing. Prentice Hall International 1989.

[36] Music for Archimedes, CD B&O 101, 1992.

[37] Anechoic Orchestral Music Recordings, Denon PG-6006, 1988.

[38] Ming Zang, Kah-Chye Tan, M. H. Er, Centre for signal processing, School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798, Three-Dimensional Sound Synthesis Based on Head-Related Transfer Functions.

[39] Bork, Ingolf. A Comparison of Room Simulation Software – The 2nd Round Robin on Room Acoustical Computer Simulation, Acta Acoustica, Vol. 86 (2000), p. 943-956.

[40] Bradley J. S., Predictors of speech intelligibility in rooms, J. Acoust. Soc. Am., Vol 80, No. 3, pp 837-845 (1986).

[41] The CIPIC HRTF Database http://interface.cipic.ucdavis.edu/

[42] Linda Parati*, Filipe Ortondo**, Comparison of Directional Sources in Simulating a Soprano Voice. Procedings of the Stockholm Music Acoustics Conference, August 6-9, 2003 (SMAC 03), Stockholm, Sweden. *Dipartimento di Ingegneria, Università, Via Saragat 1, 44100 Ferrara, Italy. **Ørsted-DTU, Acoustic Technology, Technivcal University of Denmark, building 352, Ørsteds plads, DK-2800, Kgs. Lyngby, Denmark.

Appendix C, Vocabulary

The techniques of auralization make use of many of technologies and a lot of technical terms and abbreviates are commonly used in the literature. Here is a short vocabulary to some of the most used expressions - the vocabulary is not a complete description of the individual words - the context under which the words are used are many and the subjects are rather complex.

Anechoic recording
Anechoic recordings are recordings of sound sources made without any reflections from the surroundings contributing to the recordings. A common problem with anechoic recordings are that they may often include to many high frequency components, because they are usually near field recordings and because they are recorded 'on axis' where these components usually dominate. When using such recordings with auralization systems this may often result in unrealistic sharp 's'-sounds especially in case of long reverberation times. Anechoic recordings are usually recorded in an anechoic room, but semi anechoic recordings may also be acceptable for use with auralization systems, this could be out door recordings of machinery, trains etc. or studio recordings of music.

Auralization, auralization
The term auralization was invented by Mendel Kleiner who gives the following definition: Auralization is the process of rendering audible, by physical or mathematical modelling, the sound field of a source in a space, in such a way as to simulate the binaural listening experience at a given position in the modelled space.
In the way auralization is used in ODEON, one may think of auralization as the art of creating digital simulations of binaural recordings in rooms (which may not be build yet). The aim is to provide the same three-dimensional listening experience to the listener as would be achieved in the real room at the given receiver position with the simulated source position(s) and signals.

HRTF's - Head Related Transfer Functions
In short terms the HRTF describes how an impulse arriving at a person /dummy head is smeared out by diffraction phenomenon's from head and torso of the 'person'. While an incoming impulse is only 1 (sample) long, this will result in an impulse response arriving at the right and an impulse response arriving at the left ear, which may typically have a length (of interest) of some 2 -3 milli seconds (approximately 100 samples at a 44100 Hz sample rate) - this is what is described by the HRTF's. A set of HRTF's used for auralization will typically contain a library for many different angles of incidence. The HRTF's that comes with ODEON are those made available by Bill Gardner and Keith Martin at MIT Media Lab. at http://sound.media.mit.edu/KEMAR.html. If you have the capability of measuring HRTF's it is possible to import new sets for use with ODEON.

Binaural (recording)
Humans (usually) listens using two ears. This allows us to perceive sound as a 3D phenomenon. To create a binaural recording, its not enough to create a two-channel recording (stereo), also the colouration created by diffraction from the human body has to be included. This is usually done by using a dummy head with a microphone mounted at the entrance of each ear canal - this recording may be recorded using an ordinary stereo recorder - but is now refereed to as binaural. Binaural recordings are usually played back through headphones to avoid colouration from the room in which it is played as well as avoiding diffraction from the human body to be included twice (at the recording and at the playback). If one has measured or indeed simulated the BRIR's (see below) in a room, it is possible to 'simulate' a binaural recording.

BRIR - Binaural Room Impulse Response
The BRIR is the key to binaural room acoustic auralization. The BRIR is a set of impulse responses detected at the left and right entrance of the ear canals of a dummy head (or indeed at blocked entrenches of the ear channels of a (living) person residing in a room, when a sound source (or some sound sources) has emitted an impulse. The BRIR should include all the (necessary) information on receiver position and orientations, source(s) position(s) and orientations, room geometry, surface materials and the listener’s geometry (described by the HRTFs). Convolving the left channel of the BRIR and the right channel of the BRIR with a mono signal, a binaural signal is created, which when presented to the listener over headphones gives the impression of the three dimensional acoustics at a particular position in the room. It is also possible to simulate the recording of the BRIR's, which is what ODEON does.