Alarm Group Sound

Installation Notes

This document describes how to configure a project so that it has different groups that have different sounds.
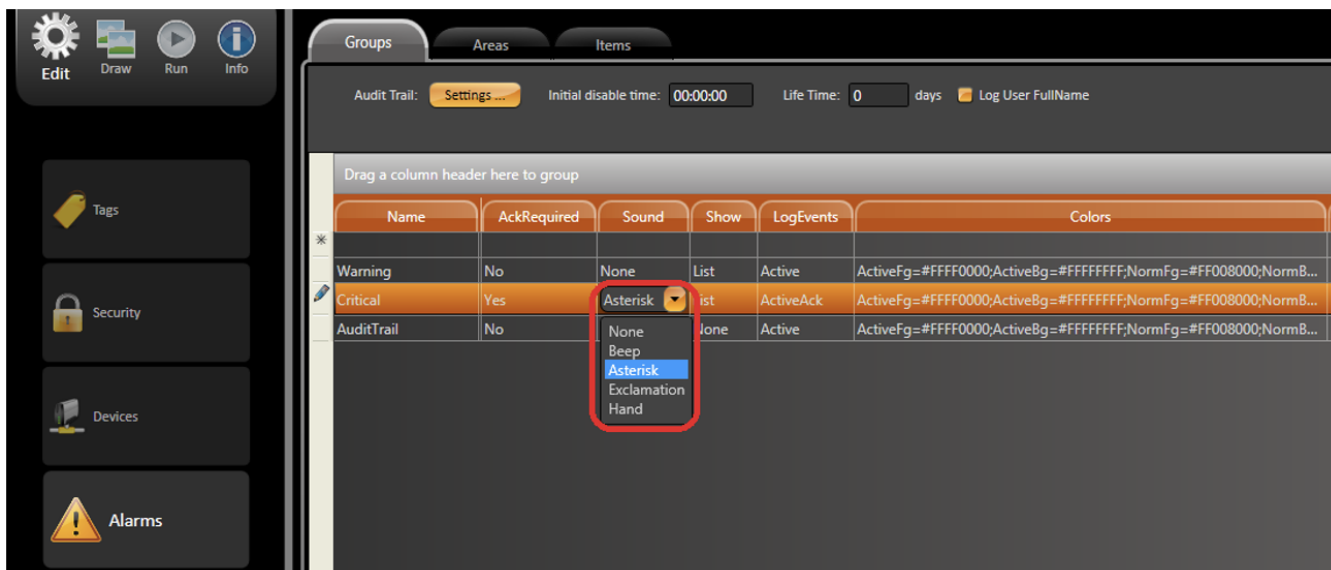
This feature requires the software version 2014.2.12 or newer. Then, you can find the Sound column in the **Alarm> Group** section.

When double-clicking on this Sound column, you choose which kind of windows sound will play when an alarm happens for that specific group.

Configuration example

To configure a project so that a group has a different sound when an alarm is active, use one of these ways:
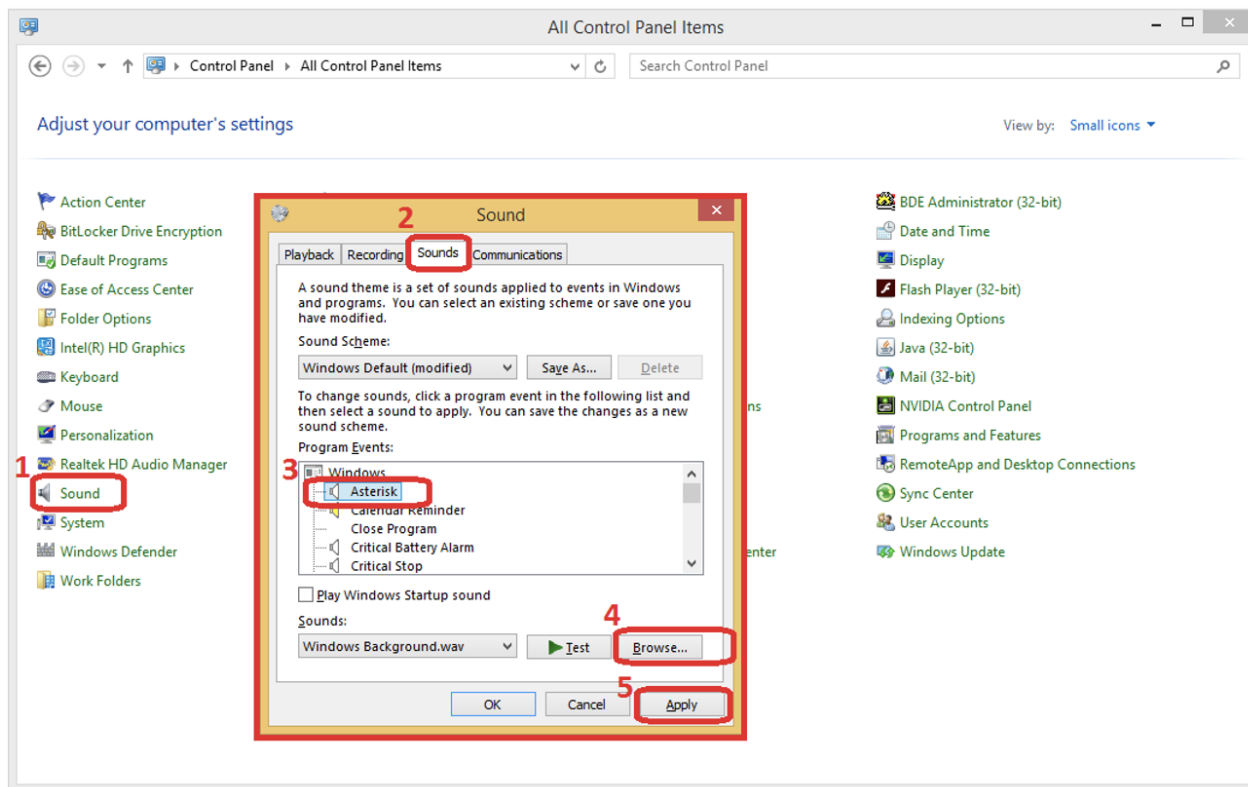
Select one of the default Windows sounds from the Sound column for the specific group.

Use the default Windows sound option as showed above. However, change the WAV file in the Control Panel on the client computers as you want.
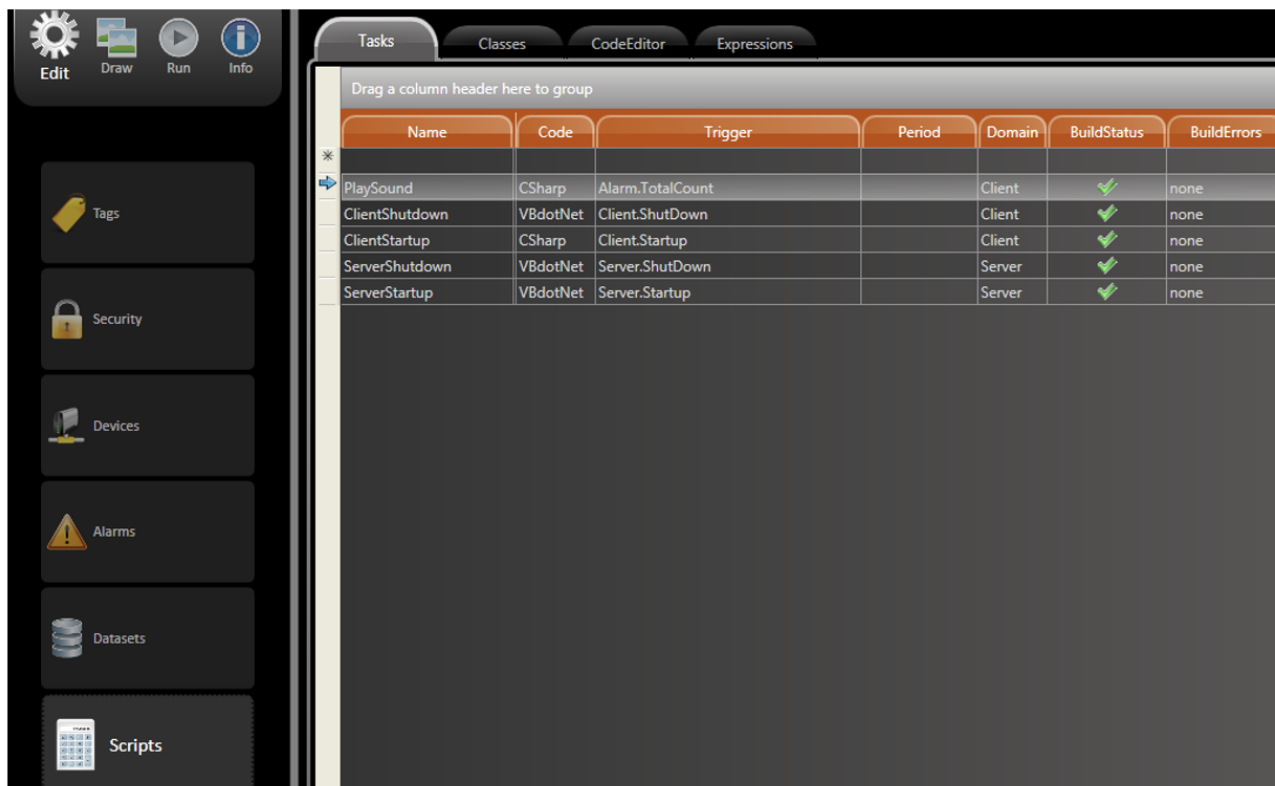
Following the steps below:

1. Open the Control Panel, and click on Sound option.
2. In the Sound dialog, click on the Sounds tab.
3. Choose the Event sound that you want to change. In this example, we are changing the Asterisk event.
4. Click on the Browse... button and select a new .wav file for the event.
5. Click on the Apply button and close the Sound dialog.

In this case, the project uses only the script Task to manage the alarm sounds.

Then, the next step is to configure the **Alarm>Group** Sound column to be None.

After that, create a script Task that will run every time the Alarm.TotalCount property changes. This script Task must be a Client Domain.

Below is an example of code you can create to manage the alarm sounds.

```
//Make sure that the script only runs when there is an active alarm
if(@Alarm.TotalCount <= 0)
        return;

//Make sure that the script only runs when the Alarm.TotalCount increases the number of active alarm
if (@Alarm.TotalCount < @Tag.AlarmTotalCountPrevious)
{
        @Tag.AlarmTotalCountPrevious = @Alarm.TotalCount;
        return;
}
@Tag.AlarmTotalCountPrevious = @Alarm.TotalCount;

try
{
        System.Media.SoundPlayer player = new System.Media.SoundPlayer();
        //Get which group that is active has higher priority
        switch (@Alarm.PriorityItem.Group)
        {
                case "Alarm.Group.Warning":
                        string filePath = @Info.GetExecutionFolder() + @"\alarm1.Wav";
                break;

                case "Alarm.Group.Critical":
                        filePath = @Info.GetExecutionFolder() + @"\alarm2.Wav";
                break;
        }

        //Set the path wav. file
        player.SoundLocation = filePath;

        player.Load();
        player.Play();
}
catch(Exception ex)
{
        @Info.Trace("Error playing audio file: "+ ex.Message);
```

⚠ Since the script Task is a Client Domain, the .wav file needs to be in the correct path that's specified in the code on every client computer.

Functionality

Every time an alarm happens when the project is running, the Product will verify which option was configured in the Sound column and play the .wav file configured in the Control Panel > Sound. Since the alarm sound is played on the client computer, all the Control Panel configurations must be done on all the client computers.