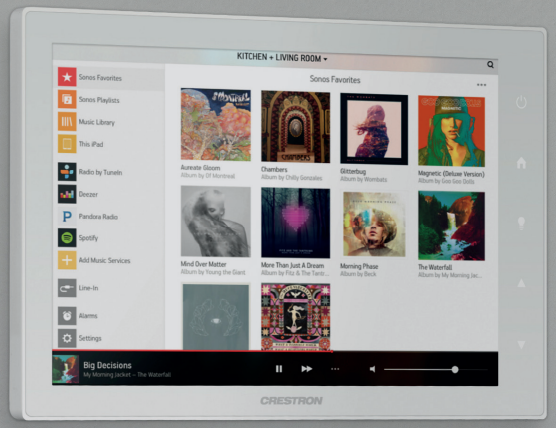
SONOS | CRESTRON



Crestron & Sonos Integration

How-to guide

CRESTRON

The following information will outline the simple process of integrating Sonos into your Crestron system. Never before has Crestron and Sonos been so tightly integrated to give your clients so much control over their Crestron Audio Distribution system.

Note: This document assumes that you already know SIMPL programming and the basic concepts around importing modules and touchscreen projects in VTPro-e.

Studio and Pyng support coming soon.

Getting Started

1. Before you start, make sure you are running all the latest released versions of Crestron software and databases. Either run MasterInstaller, and follow the prompts or go to crestron.com/resources/software

2. Make sure you have downloaded, and installed, the very latest 3-series processor firmware. IMPORTANT: You must use 3-Series Firmware v1.501.0103, or later or this Sonos integration will not work.

3. Go to partners.creston.com/sonos

4. Make sure you are logged in and go to the Resources tab.

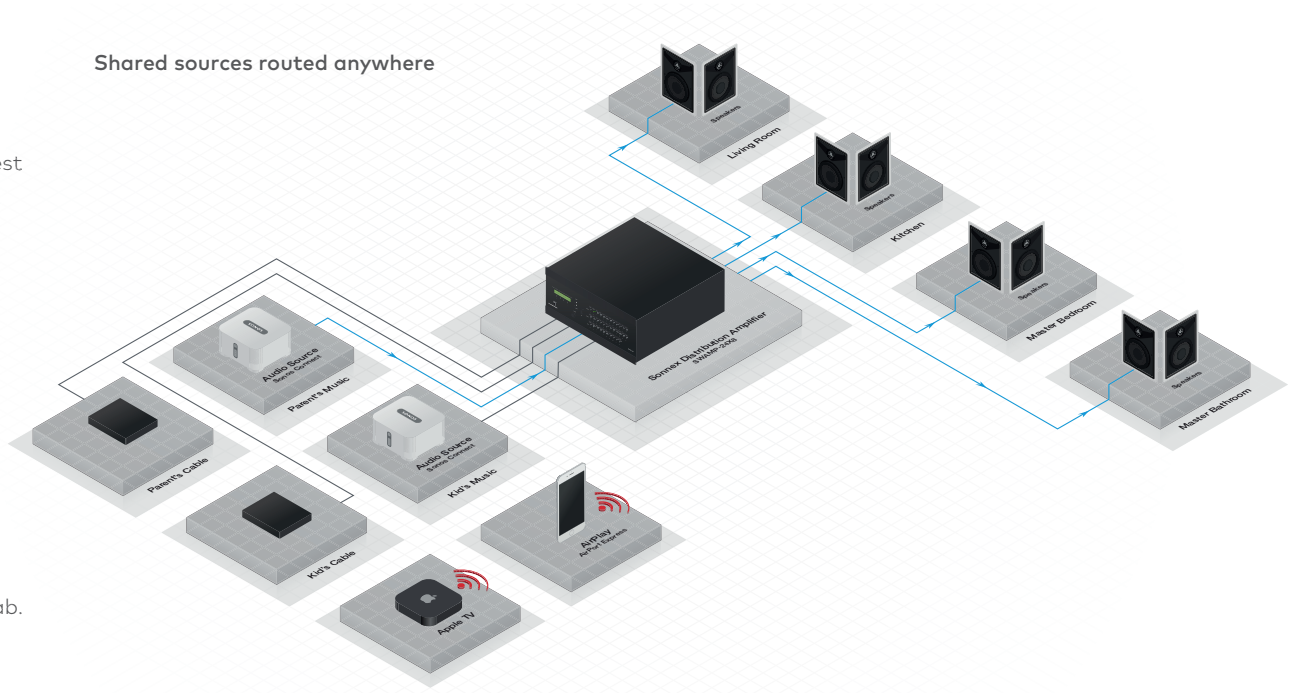
5. From there, download the zip 'Sonos Sample Project'.

Sonos Architecture

Now that we have all of the files, let's examine the overall architecture of the system.

In a traditional Audio Distribution system there are shared sources at the head-end that can be routed anywhere. For example, in this case we have named a CEN-NSP-1 'Parent's Music' and through Sonnex that source can be routed to any room.

Sonos doesn't act like a traditional centralized audio distribution source, instead, it can act as both a source and the room. The Sonos user can use the Sonos app and play different content in each room if they desire, or they can choose to group rooms together to make the content track.



With Sonos Connects now as the sources, through the distribution amplifier you could technically route any Sonos Connect to any room, in the same way. However, for the best user experierience, we recommend a Sonos device dedicated to each room.

Each Sonos device should be named with the room name it is dedicated to. This is so that each room can be completely independent and so the Sonos grouping feature can show the user each Sonos device with the room name that it is assocatied with.

In this example, the Sonos named 'Living Room' even though technically it can be routed through the Sonnex to any other room, the program should only let the Living Room Sonos be routed to the Living Room speakers. The dotted lines in the diagram represent this.

Note:

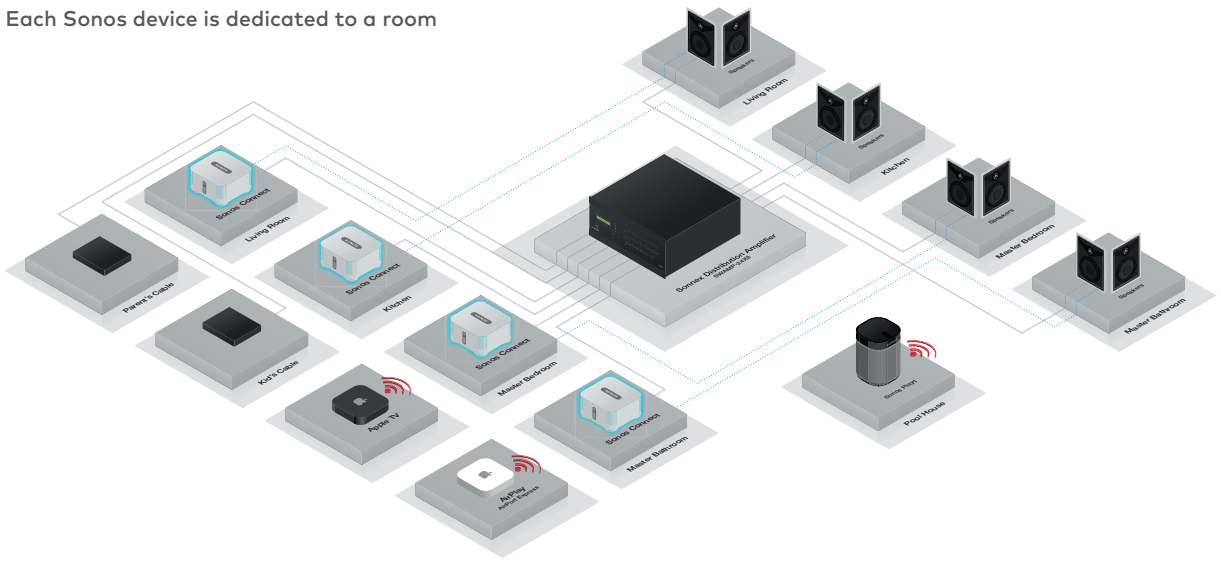
Although Sonos Connect players may be configured as 'shared sources' as shown in the previous diagram, this will result in a confusing user experience and thus Crestron recommends avoiding it. Specifically:

1. Homeowners will not be able to use the Sonos app to select a room and start music playing. They must always use the Crestron mobile app or touchscreen for this.

2. Grouping via the Sonos app will not work. Instead, audio sharing between rooms can only be accomplished from the Crestron mobile app or touchscreen. Grouping via the Sonos app will cause undesired behaviour that will confuse your customers.

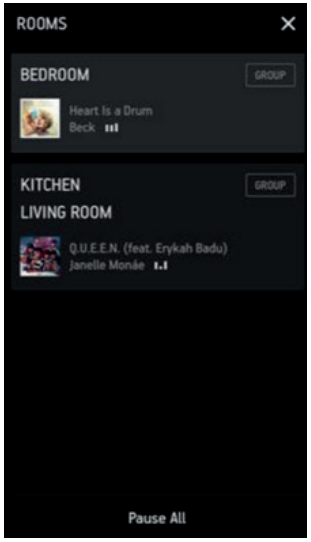
3. If you have Sonos wireless speakers in one or more rooms, your customer will be presented with a combination of 'shared' and 'dedicated' Sonos players (e.g. Dad and Pool House) adding to the confusion.

Each Sonos device is dedicated to a room



This model is also very flexible since if you have remote rooms that may be difficult to get a head-end wire to, you can use a wireless Sonnos speaker, yet can still be controlled via Crestron.

In the Sonos App the end-user sees each Sonos device named with a room name. When they go to the grouping menu they will connect each device as an individual 'room', or zone.



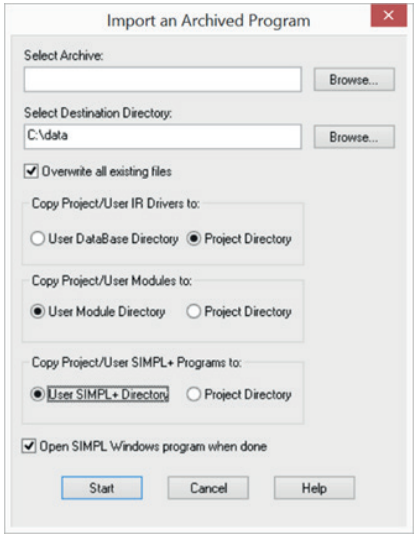
Examining the sample program

1. Unzip the contents of that zip into an empty folder on your computer (keeping the folder structure).

2. Note the file named 'Sonos Demo Release Notes'. Make sure you reference that document for the latest minimum version information. This document also includes written directions and some tips on how to use these modules.

3. Launch SIMPL Windows.

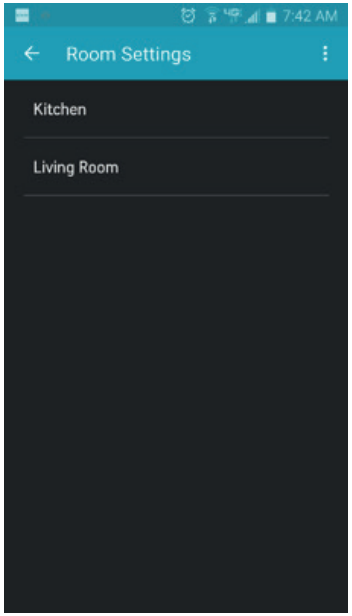
4. Click File → Import.



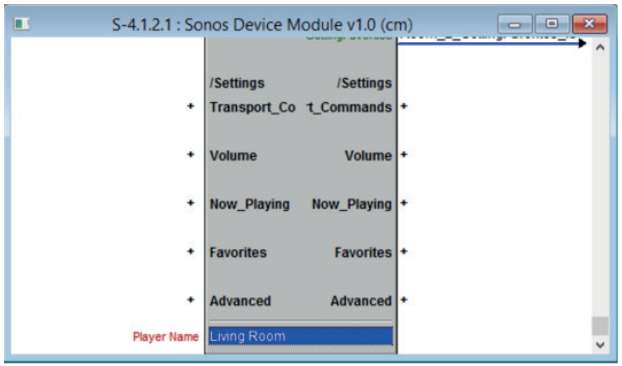
Note:

We are assuming you have already figured out how to get the Sonos device up and running on the network and communicating with the Sonos app. If you haven't yet, pause now and follow the steps that came with the Sonos product.

To see a single touchscreen and one Sonos device in action, start by going to the Sonos app and get the exact name of that Sonos 'Room'. For example, in this example in the Sonos app:



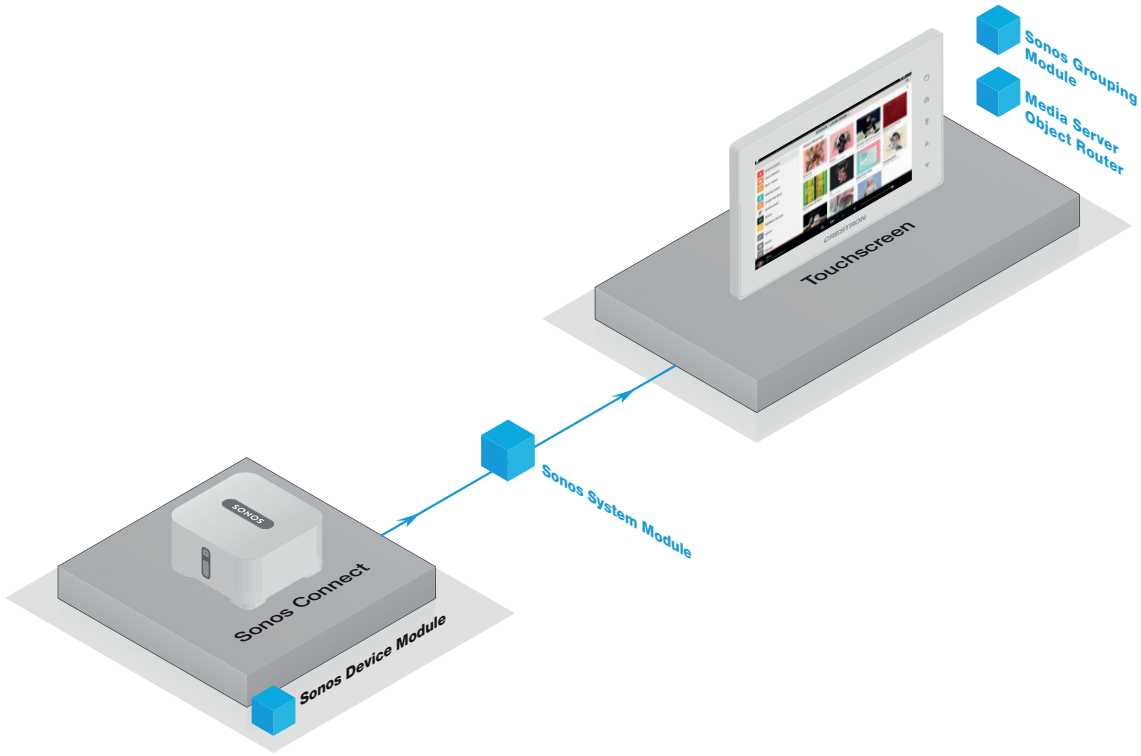
Take that name, in this case, 'Living Room' and type in that exact String into the Sonos System Module parameter:



This name lets the 3-Series processor detect that particular Sonos device on the network and communicate with it automatically. No IP addresses or IP Tables needed. **Just make sure when you discuss this system with the homeowner that you make it clear that they should not change the room names from their Sonos app.** Otherwise, they will break the Crestron integration.

Compile and upload to the processor and touchscreen. Alternatively use XPanel to explore the interface to make sure the code is functioning properly.

Now let's look at the module architecture, so it's easy for you to either modify this sample system, or copy and paste only the pieces you need into one of your own SIMPL Programs.



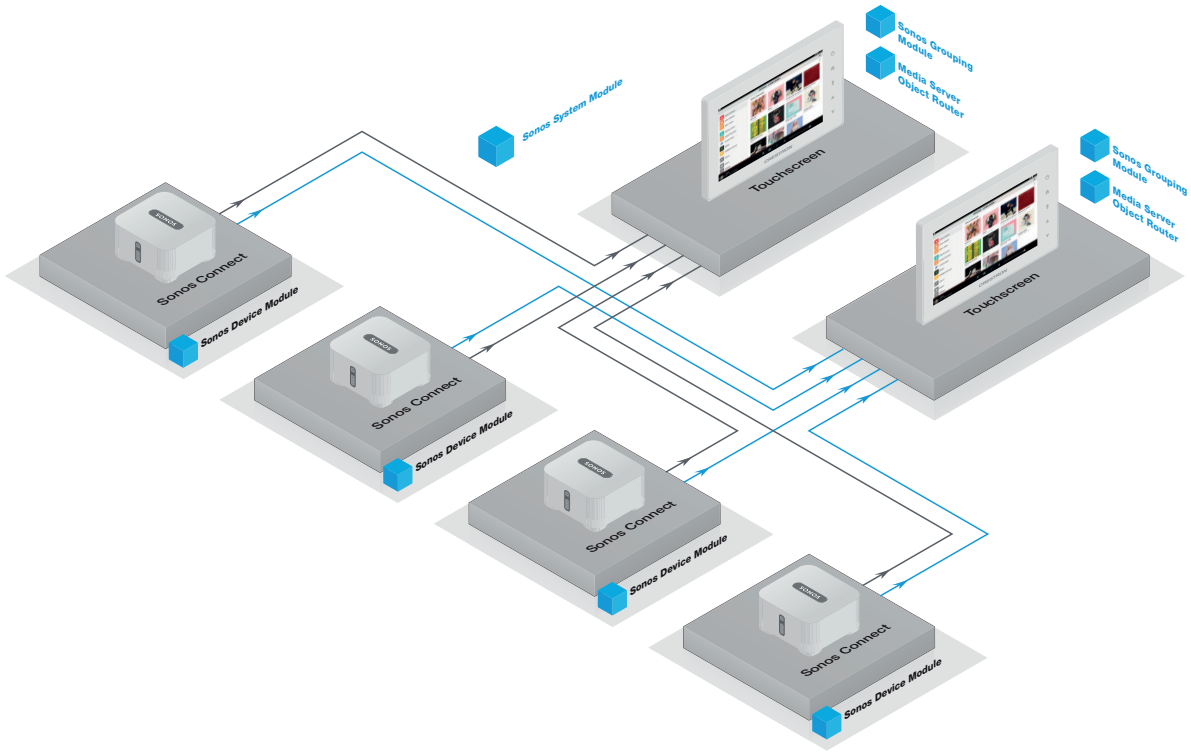
The 'Sonos System' module is responsible for auto-discovery of the Sonos players on the network and manages system-level things such as grouping. **You should have one, and only one of these in your SIMPL program.**

The 'Sonos Device' module is responsible for managing each individual Sonos player. **You will have one instance of this module per Sonos device in your system.**

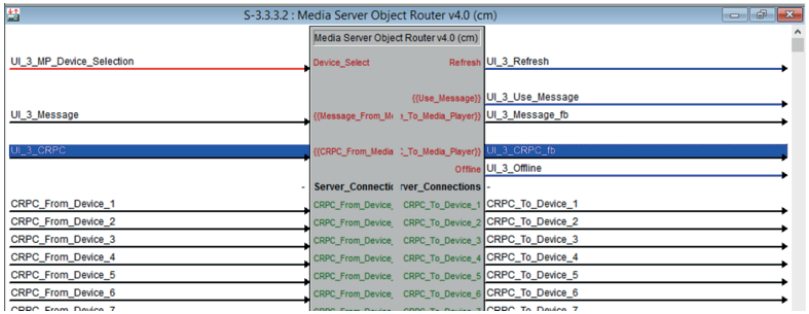
The 'Sonos Grouping' module is responsible for managing grouping related features for each touchscreen. **You will have one instance of this module per touchscreen in your system.**

These modules communicate behind-the-scenes via C# shared memory, so there are no signals for you to interconnect between them in SIMPL.

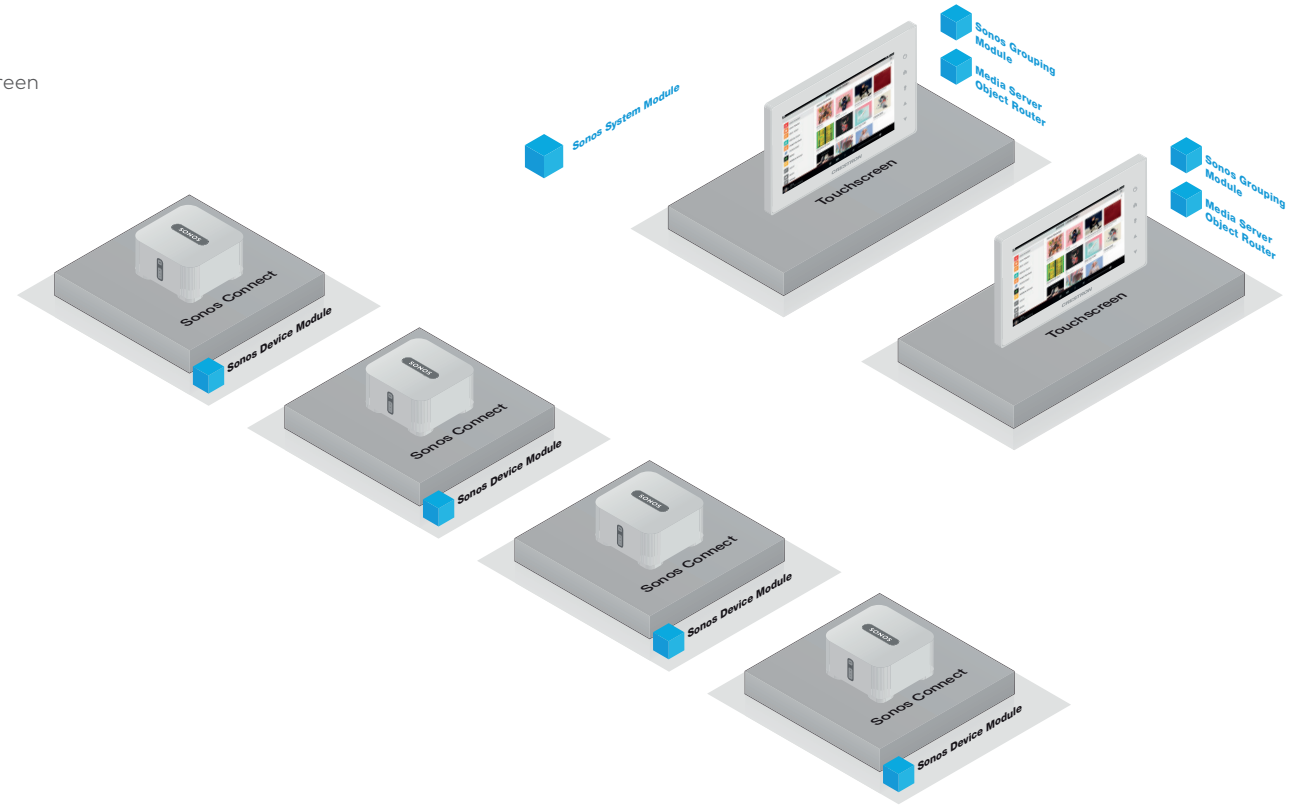
The 'Media Server Object Router' module is responsible for routing CRPC from the Media Player Smart Graphics Application on a touchscreen to currently selected Sonos Device Module. **You will have one instance of this module per touchscreen in your system.** This module lets the user from any of the touchscreens have access to any of the sources:



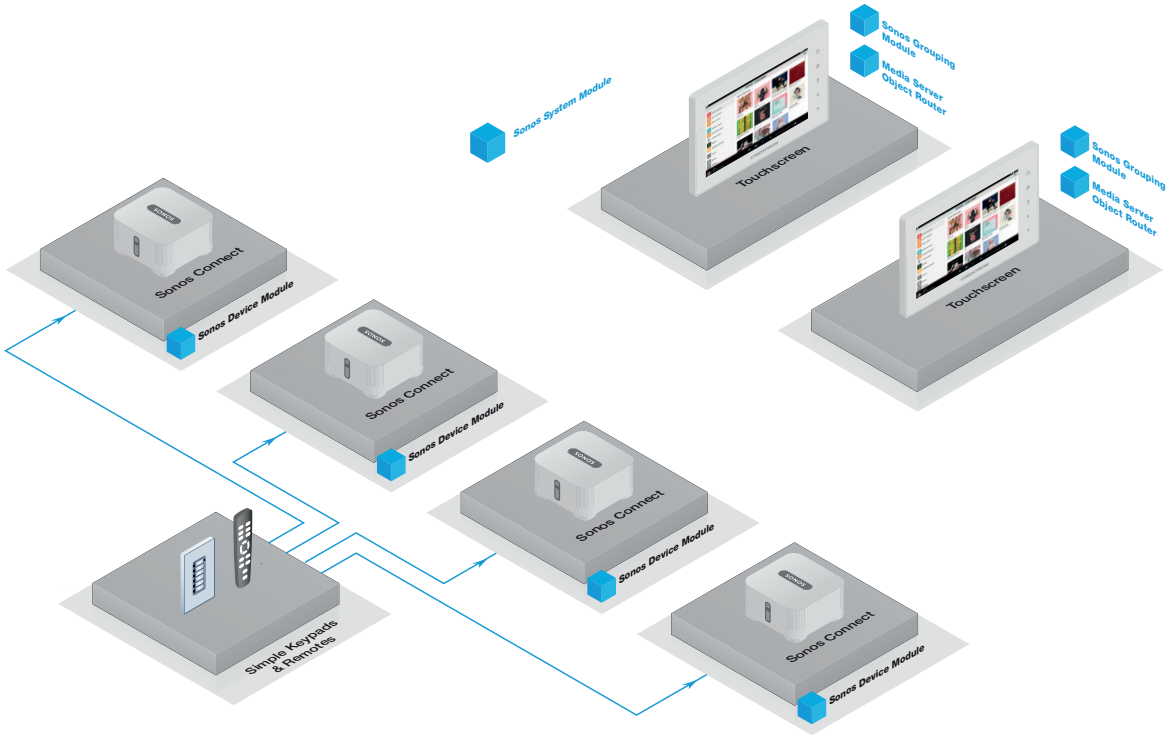
You need to route the CRPC inputs and outputs into the touchscreen's Media Player Application Smart Object. For each of the 'From_Device_#' and 'To_Device_#' signals just route each of those inputs and outputs to the appropriate Sonos Device Module.



Then for each Sonos player and each touchscreen in your system, replicate this pattern:



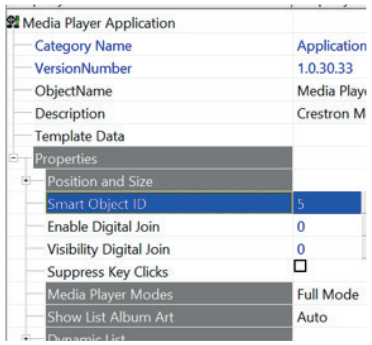
For remote controls, keypads, and the like, you can connect individual commands to the appropriate transport command directly on the 'Sonos Device' Module.

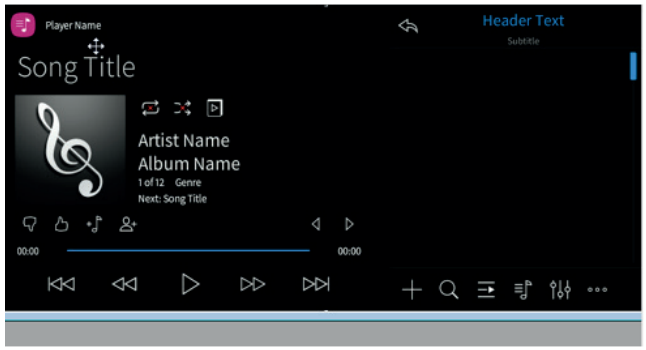


A look at the touchscreen projects in VT Pro-e:

The bulk of the UI is handled in the Media Player Application, via CRPC so there isn't much that you need to do in VTPro-e.

This is already done for you in the sample project. But if you are doing this from scratch, make sure you have a Media Player Application added with a Smart Object ID.

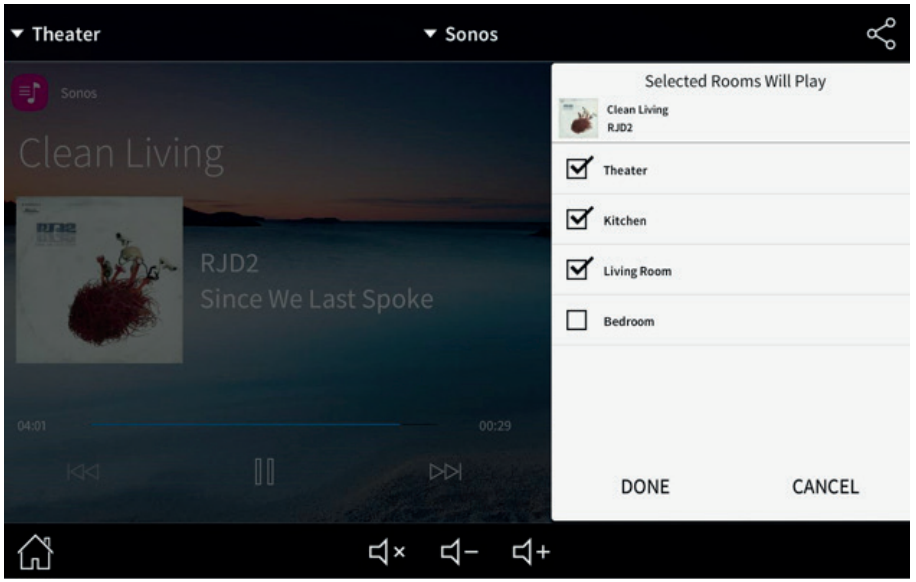




Grouping

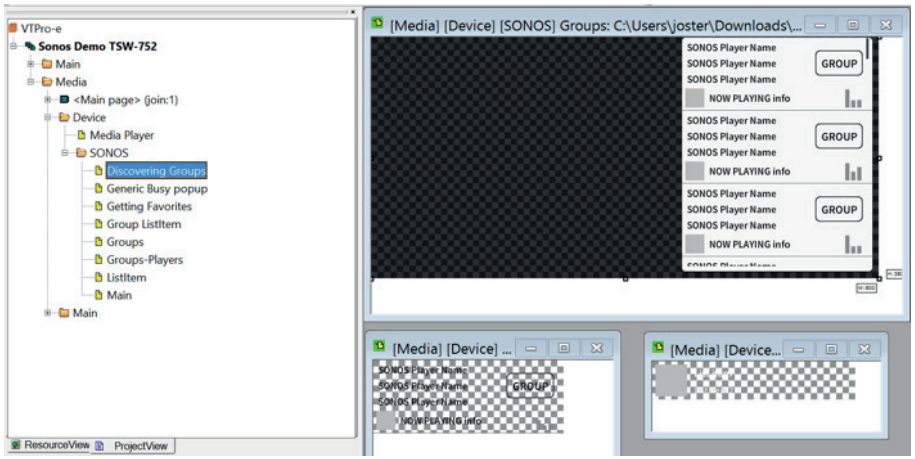
To have the ability to group and ungroup multiple zones from the touchscreen, we have given you the ability to implement Grouping via the Sonos Grouping Module.

To implement the Grouping functionality similar to the native Sonos App, in VTPro-e we have a separate subfolder for this. Grouping is a feature that is OUTSIDE of the Media Player application so that is why it is built with individual elements like buttons, text fields, subpages, and the like. The easiest way to see how all of this works is to see the Grouping feature at run-time in the sample project then trace it back inside of VTPro-e. Have at least two Sonos devices configured, then at runtime press the Grouping icon to be able to group and ungroup rooms:

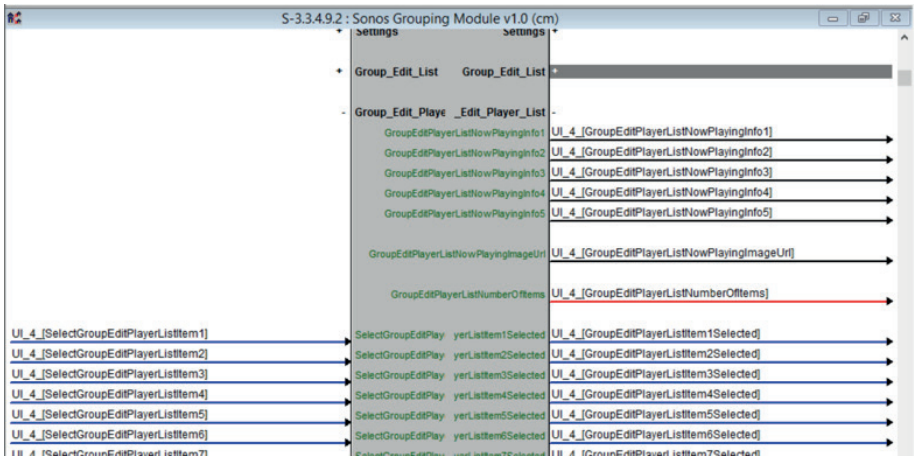


Then look at the related subpages inside of the VTPro-e project. For each of these objects in VTPro-e you can back trace it to see where each of these signals land on the appropriate 'Sonos Grouping Module'

In VTPro-e grouping related objects:



In SIMPL, Grouping related signals:



Armed with this basic knowledge of how these modules communicate with the various components in the Touchscreen project, you can dive into the details in the Sonos Sample Project. You can also use this sample to copy and paste out elements into your own program.