

# Creating Custom Player Hails for NPC Ships

Everyone wants to hail every ship they see. That is the comms officer's job, after all! The generic hailing options aren't very good and you often don't want the NPC to speak first. I'm going to walk you through two scenerios.

1. The NPC should have custom Topics available to the player starting at the beginning of the mission.
2. The NPC's custom Topics must change based on a specific gameplay action.

This tutorial assumes that you are already familiar with creating basic dialog trees and triggering them using actions. You can brush up using the first tutorial in this series [Tutorial: Creating a Sample Dialog Tree](#).

Let's dig in!  
-Jazzy

## 01: Some Rules

Let's review for a second. When adding a Dialog Group, TEXT is what the NPC sends to the Player and TOPIC is what the Player can send to the NPC. TOPICS are linked to the next Dialog Group in the dialog tree. Triggering TOPICS is how you advance through the dialog tree.

All Dialog Groups can have both TEXT and TOPICS but they are not required to have both!

If a Dialog Group only has TOPICS, it will change what the Player can send to the NPC but will not cause the NPC to send a message to the Player.

Dialog Groups are loaded in order from top to bottom. The last Dialog Group with a specific name will override all Dialog Groups loaded previously for that object. If your module is a child to the Horizons module and the object is created inside your module here is how it will load the dialog.

- Horizons Module Level Dialog
- Your Module Level Dialog
- Your Object Level Dialog for the Object at the module level
- Your Object Level Dialog for the Object created at the Mission level (if object has the same name)

The default module-level dialog tree for a ship is set at the Module level in the Vessels menu. Inside a Vessel, you can set the default dialog tree in the NPC menu. This means that neutral and allied ships will pull from the Generic tree and the Daichi will pull from the Daichi tree.

Inside these Module level trees there are several Dialog Groups. Every one should contain a Dialog Group named Main. The Main dialog group usually has no TEXT and multiple TOPICS. Main is what is triggered for an NPC at mission start. These are the base options for what you can send to an NPC when you run across them.

## 02: How Dialog & Object Settings are Loaded

Let's use a Dart as an example. The Dart is a Vessel at the Horizons module level. It pulls its base dialog from the Daichi dialog tree in the Horizons module. The Horizons Module loads first.

In your module, at the module level, you add a Dialog called Daichi. Since your module loads next, all Darts in your missions (not missions inside the Horizons module) will load up your version of the Daichi default dialog.

In your module, you edit the Map Objects to create an object called Dart Commander. It is a Dart so it is pulling the default Daichi dialog as outlined above. If your module has a Dialog Group called Main inside the Daichi dialog, those topics will be the hailing options for this Dart Commander. If you didn't include a Dialog Group called Main inside of your Daichi dialog, that specific dialog group will default to what is at the Horizons level.

Okay but this guy is a Dart Commander. You don't want his dialog to be the same as all other Daichi. Again, at the module level, you edit the Dart Commander Object. In the Dialog tab of the Edit Object menu, you press Add Dialog Group and name it Main. What happens at the mission start is that the Dart Commander initially loads Horizons/Daichi/Main, then it overrides that with YourModule/Daichi/Main, and then overrides that with YourModule/Objects/Dart Commander/Main.

Great, so now you have the Dart Commander in every mission just hanging out with his custom dialog. What happens when you want a specific mission to load an alternative Main for the Dart Commander, overriding the YourModule level one? Go into the mission you are looking to change the dialog for. Under [Mission] Objects create a new object called Dart Commander, giving it the same Name and other general details. In the Dialog tab, add a Dialog Group called Main. Fill out the mission-specific dialog tree. You can change any other settings for this Dart Commander that you want too: Cargo, Spawn Location, Tasks, Properties, etc.

Now when the mission loads it will load the Dart Commander that is at the module level and then when it loads the mission it will replace the Dart Commander from the module with the one outlined in the mission. YOU MUST USE THE EXACT SAME NAME for this to work. Now the Dart commander is loaded with all of the custom settings, including mission-specific dialog.

## 03: Switching Hailing Topics Mid Mission - Part 1: The Setup

Overriding the Main topic is great to set up the first hail that loads at the beginning of the mission. It does no good if your hailing options need to react to what happens in the mission. Here is a scenario:

Our Dart Commander has his normal module-level Main topics at the start of the mission. He appears to be just another extra in this film. When speaking with an informant, the Horizons is told that the Dart Commander is secretly selling off some surplus Daichi ordinance. Since their missiles are compatible with yours, you should go buy some before taking on "Something Big".

Well, this is cool. We want those missiles. Good mission design would require us to be informed about this opportunity before we can ask the Dart Commander about it. We create an object inside our mission that is the Dart Commander, same as the last example in the last section.

We DO NOT need to create a Main dialog group because inheriting the one at the YourModule level is

just fine. Let's add a Dialog Group called "About Those Missiles" with a Topic that says "A friend told me that you may be able to help me get some missiles." Link that topic to the Dialog Group "Maybe" that we are creating next.

Create a Dialog Group called "Maybe" with text that reads, "I may be able to help you for 50 Raw Materials." topics of "We'll send some over." and "No Thanks", and an Action that enables an event titled "Dart Commander". You can link these to some fluff if you want but the **ACTION** is the important part here.

Create a Dialog Group called "Pleasure Doing Business" with text that reads, "Thanks for the retirement funds. Send over a shuttle for your missiles."

Make an **Event** with a **Condition** that if the Dart Commander has 50 Raw Materials two actions will fire the following three **actions**: Add/Remove Cargo:Raw Materials:-50 Add/Remove Cargo:Silkworm [Evil]:+10 Send Topic:Dart Commander:Pleasure Doing Business

## 04: Switching Hailing Topics Mid Mission - Part 2: The Trigger

Now your Dart Commander has his custom tree...but when you hail him you will still see the Main topics that it is inheriting. We fix this in an Event Action or a Dialog Action. To keep this example going, let's edit the dialog for our Informat Object (yep, go create him too).

Create a Dialog Group called "Main" with a topic that reads "What do you have for me?" and link it to "Something Big".

Create a Dialog Group called "Something Big" with text that reads "I have reports from some Colonial Movers about Something Big. It scared them. You should be ready for a fight." with a topic that reads "We are low on ordinance" and link it to "I Know a Guy".

Create a Dialog Group called "I Know a Guy" with text that reads "I know someone who can help you out. An old Daichi Dart Commander who is trying to save for retirement. He is selling off their surplus missiles." and a topic that reads "Send the coordinates!" and link it to "Coordinates".

Create a Dialog Group called "Coordinates" with text that reads "Done", a topic that reads "Horizons Out.", and a pair of **ACTIONS** that "SendTopic:Dart Commander>About Those Missiles" & "SetMissionWaypoint:PLACE:NAME". This will **CHANGE** the available hail the next time you are in the same system as the Dart Commander and set a waypoint to go meet it.

## Wrap Up

This one was pretty deep down the rabbit hole. While the examples add a lot of text to this tutorial, it is helpful to see how something like this can be used and the specifics of building it.

If you notice that you don't see Map Objects in the Module/Maps menu, that is because that isn't currently exposed. These objects are in the XML file though. Look in your .../Starship Horizons/Modules/Horizons/Maps/Milky Way.xml file in Notepad or Notepad++. Scroll past all of the enumerated star systems and check out the <objects> section. Here is where the default Comm

Stations and Spacedock are spawned. Look at some Objects from one of the Horizons Base Missions (.../Starship Horizons/Modules/Horizons/Missions/...) to see how some of the object parameters are overridden. Hopefully, this will be exposed at the module level in the Designer soon.

This is a lot. If you have questions, come ask them on the [Discord Server](#) or DM me on Discord @Jazzy#0858.

[Dialog Tutorials](#) | [Tutorial: Creating a Sample Dialog Tree](#) | [Creating Custom Hailing Dialog](#) | [Back to Tutorials](#)

From:  
<http://wiki.starshiphorizons.com/> - **Starship Horizons Wiki**

Permanent link:  
<http://wiki.starshiphorizons.com/modding:tutorials:dialog-hail?rev=1620412120>

Last update: **2021/05/07 18:28**

