# Fluid Network Tutorial

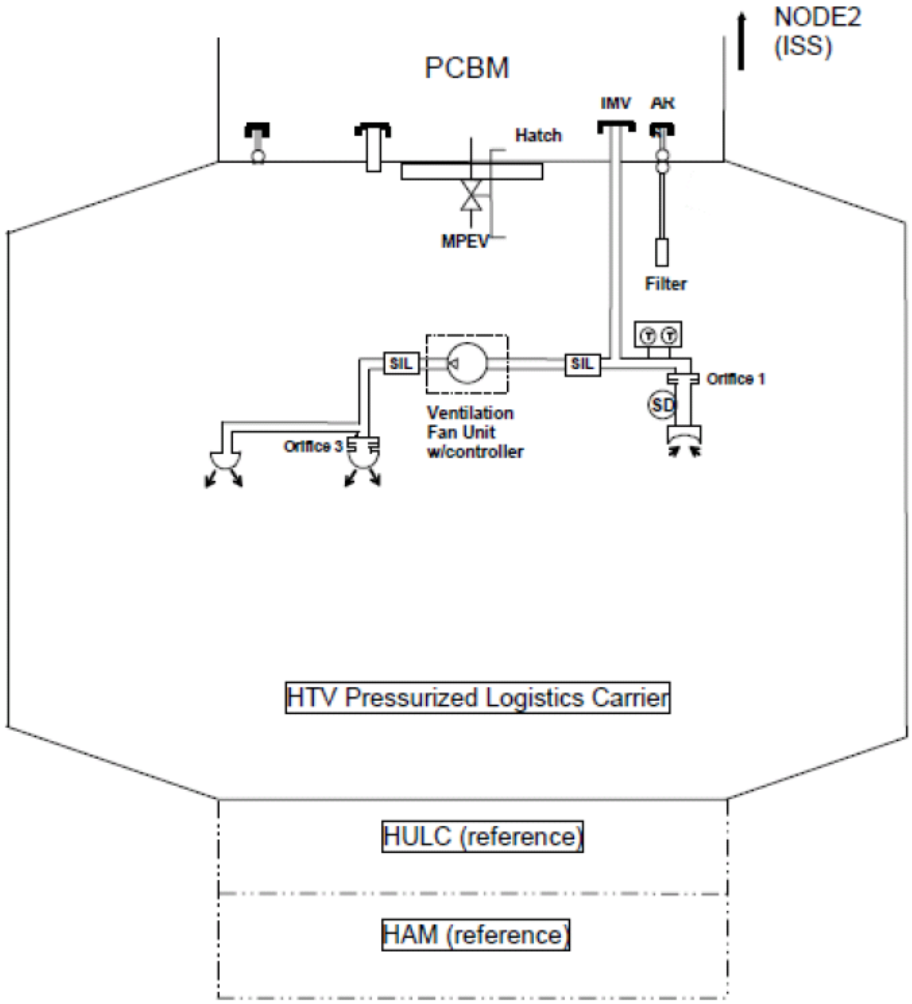
This guide will walk you through creating a GUNNS fluid network of slight complexity. It will cover all the fundamentals you should need to begin creating more complex fluid networks. In general, fluid networks are slightly more complicated than the others. This is mostly due to the need for nodes to keep track of a fluid state. Fluid states hold quantities like fluid mass fractions, pressure, temperature, etc. For this reason, fluid nodes are fundamentally different than the nodes you will find in other networks. Because of this, the fluid node shape in GunnShow is a little bit different than the electrical node shape. This distinction, and others, will be elaborated upon later in this tutorial.

This tutorial will assume you have a small amount of prior knowledge to GUNNS and GunnShow. If you do not, it is probably best that you start with the Getting Started Tutorial to get a brief introduction to GunnShow. If you are not well versed in GUNNS, you can check out the Electrical Refresher document.

Once you feel comfortable with these pre-requisites, we can begin to start building up a GUNNS fluid network. This example will start with an HTV ECLSS design schematic, and show you how to go from the schematic to a GUNNS subsystem model of HTV's ECLSS.

## HTV's ECLSS

Below is a schematic of the Japanese H-II Transfer Vehicle. This has been taken from the ECLSS subsystem section of the HTV Subsystems Manual (JSC-36476). Our goal will be to go from this schematic to an initial baseline software model of the subsystem as quickly as possible. This tutorial will focus on completing a first cut model of the subsystem, and will avoid focusing on specific model parameters that must be tuned correctly to get a model of higher fidelity. While we avoid focusing on these specific parameters in this tutorial, we will point out ways to get more information on individual model parameters, and it will become clear how you can begin to tune these when you are ready to take your first cut model and begin building in higher fidelity.




**HTV ECLSS Overview**

Looking at the above schematic, we can see that HTV's ECLSS is primarily a simple air loop. There is both a hatch and an IMV (Inter-Module-Ventalation) duct connecting up to ISS Node 2's PCBM. When the hatch and IMV are open and the cabin fan on, air is circulated between the HTV and ISS Node 2 cabins. Our GUNNS model of this system will focus on the following key components:

- HTV Hatch
- HTV MPEV (Manual Pressure Equalization Valve)
- HTV IMV Duct
- HTV Cabin Fan
- HTV Smoke Detector
- HTV PLC Volume (Pressurized Logistics Carrier)
- HTV Vent Relief Valves (4)
- ISS Node 2 Stub Model

You will notice a few additional models were added to the above list that do not appear in the HTV ECLSS schematics. On the real HTV their are four Vent Relief Valves. These are responsible for venting air should the HTV become over-pressurized during it's trip to ISS. These valves are not depicted in the above schematic, but we will implement them in this tutorial anyways in order to illustrate some features of GUNNS and GunnShow. Secondly, we will add in a stub model of the ISS Node 2 Cabin.

This is just done for completeness.

## Side Note on External Interfaces in GUNNS

An ISS Node 2 stub model is being used for the purpose of keeping this tutorial simple. However, in the TS21 Visiting Vehicle simulation that was originally developed with GUNNS and GunnShow, we did not have to model ISS Node 2. A model already existed within the Space Station Training Facility that we could interface with. A method for handling these types of external interfaces was developed for GUNNS and GunnShow. This tutorial will not cover this more advanced topic. Sometime soon the GUNNS team will create some documentation to describe how this interfacing was accomplished, and we will link to it here.

## Getting a Start on the HTV ECLSS Model

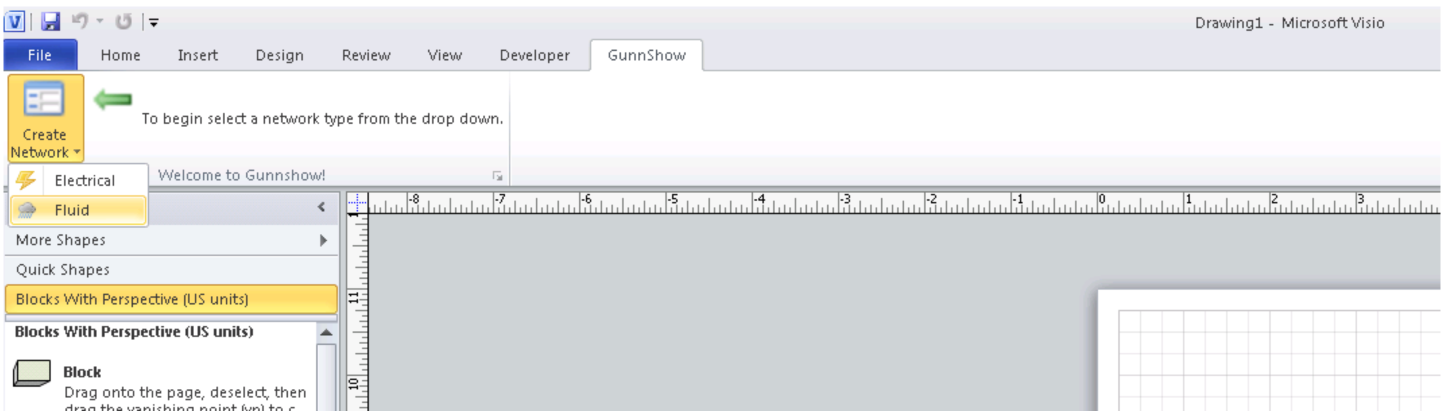
In the early days of GUNNS, the typical process for realizing this HTV ECLSS model would start with a white board drawing of how this system might be represented in software. You would end up with a drawing on the board that had all the key models and how they were connected up in a nodal network. Once this big picture of the system was agreed upon, ECLSS developers would try to determine if GUNNS currently had all the software models needed to build the system. Perhaps their was an exisiting fan model, but for HTV's particular application, it needed to be modified in some way. If that were the case, a team member would be assigned the task of adding the feature to the core fan model, or deriving a subclass of the existing fan model that added the necessary implementation.

In parallel with this development effort, the team might have had a GUNNS network modeler. This person was responsible for integrating all the components into the overall model of the system being developed. This person spent a lot of their time interfacing with the GUNNS core solver class, tuning individuals models, and incrementally building up the connections between components as they became ready.

Nowadays, GUNNS and GunnShow have simplified this process, and made it so that users with a somewhat limited knowledge of nodal analysis can build up an ECLSS model and manage it relatively quickly and easily. To see how it's done, let's dive into GunnShow.

## Starting a New Fluid Network In GunnShow

We start by opening up Visio and creating a new drawing (see the Getting Started Tutorial if there is any uncertainty here). The GunnShow Ribbon will have a create network button. Select this button, and then select Fluid Network.

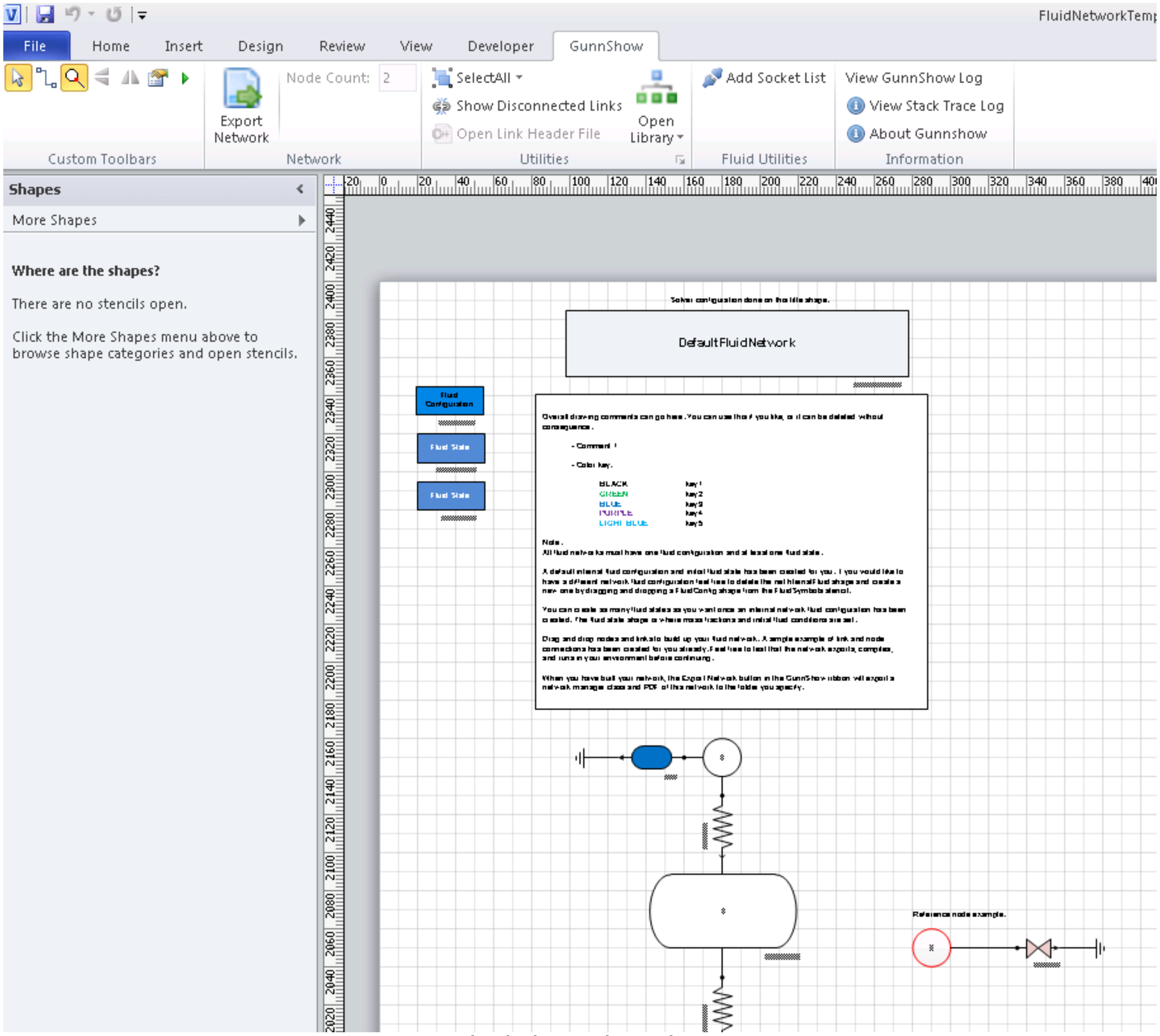



**Starting A New Fluid Network**

This will open up what is known as the standard fluid network template. A template is used here in order to help quickly setup your drawing so that you can begin working and exporting a network as soon as possible. Let's go ahead and try to export the network as it stands right now. First, let's do a Save As and name the drawing whatever you would like. You can also place the drawing wherever you would like. Next, follow the steps for a network export. See the getting started tutorial if there is any confusion on how to do this.

If all goes well, the network should export without any problems. At this point, you have two options. First, you could go through the steps outlined in the Simulating a GUNNS Network Tutorial. This will show you how to pull in your baseline GUNNS network into Trick, and verify that everything works before continuing on. The other option is to put off the integration with the Trick framework and save it for later. This second option allows you to first focus on modifying your current baseline to match the model you are working towards.

Which option you choose is usually a matter of preference. For GunnShow beginners, however, the GunnShow team advises the first option. It is better to first get something that should be very simple working in your environment. Once you have the simple network going, you can then move on to adding and changing components to your system.




**The Fluid Network Template**

There are a few things to note about the fluid network template:

1. Title Block: This is where you specify both a label for you network drawing, and the actual name of the C++ GUNNS network manager class that will be generated upon export. See Getting Started for more info.
2. Legend/Info Block: This is just a block that contains some general information and a template for a legend should you want to use one in your drawing. None of it is necessary and can be deleted immediately if you'd like.
3. Fluid Configuration Shape: This is a default fluid configuration shape. By default it is an internal fluid configuration for air. If this is what you need for your network, great. If not, hang on, and this tutorial will go through the process of deleting and replacing it with a user defined one.
4. Fluid State Shape: This is a default fluid state shape. By default it is given all of the properties of air at a standard temperature and pressure. This shape goes hand in hand with the fluid configuration shape. If you need to change this, hang on, and this tutorial will go through that process in a later section.
5. Default Nodes/Links: There are some default nodes and links configured for your convenience. These can be copy-pasted if you need some of the same ones for your own model, or they can all be immediately deleted.
6. Example Reference Node: An example of a reference node and how they are set up is included. See the shape data, and take note of how it is connected should you need to use reference nodes in your network drawing. This can also be deleted.

## The Fluid Stencil

Now that you have your baseline GUNNS Fluid Network, we can begin the process of customizing it. To do this, you will need access to the GUNNS fluid models. In GunnShow terms, we call this the Fluid Stencil. The GunnShow Fluid Stencil has a shape for all or very nearly all the generic fluid models that have been developed thus far for GUNNS. To open the fluid stencil, follow the image below and go to Model Libraries -> Fluid.

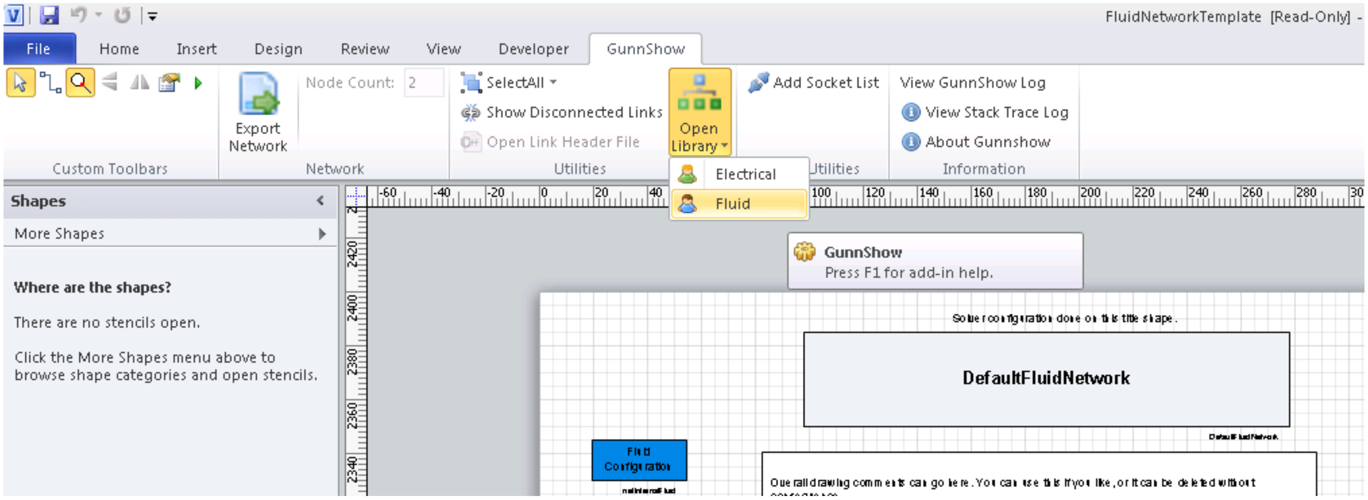
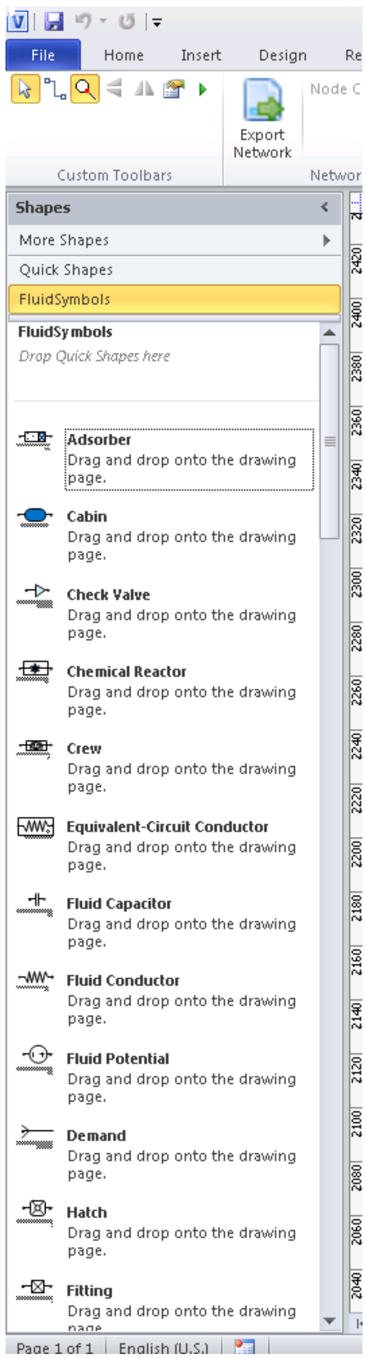


**Fig 3 The Fluid Stencil**

Take a moment to scroll through this library and appreciate all the fluid models you have access to as a GunnShow user. These models were all developed by many different people on the TS21 ECLSS and TCS team. To pull a particular model into your GUNNS network, it is as simple as dragging the shape from the Fluid Stencil and dropping it onto your drawing.

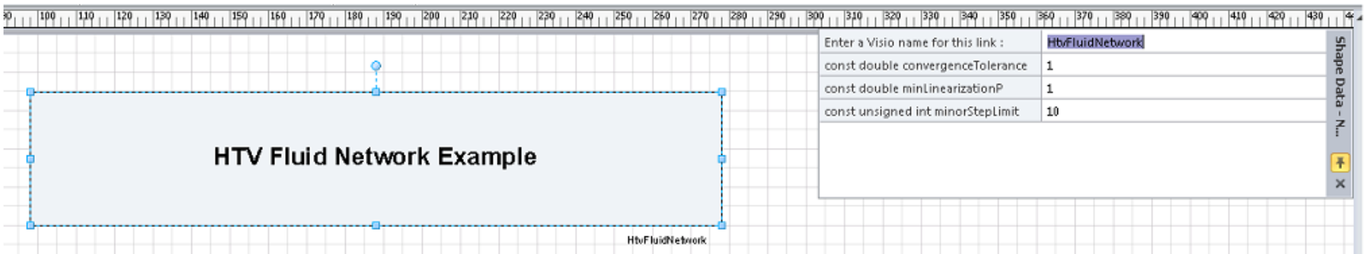


**The Fluid Stencil**

## The Title Block Shape

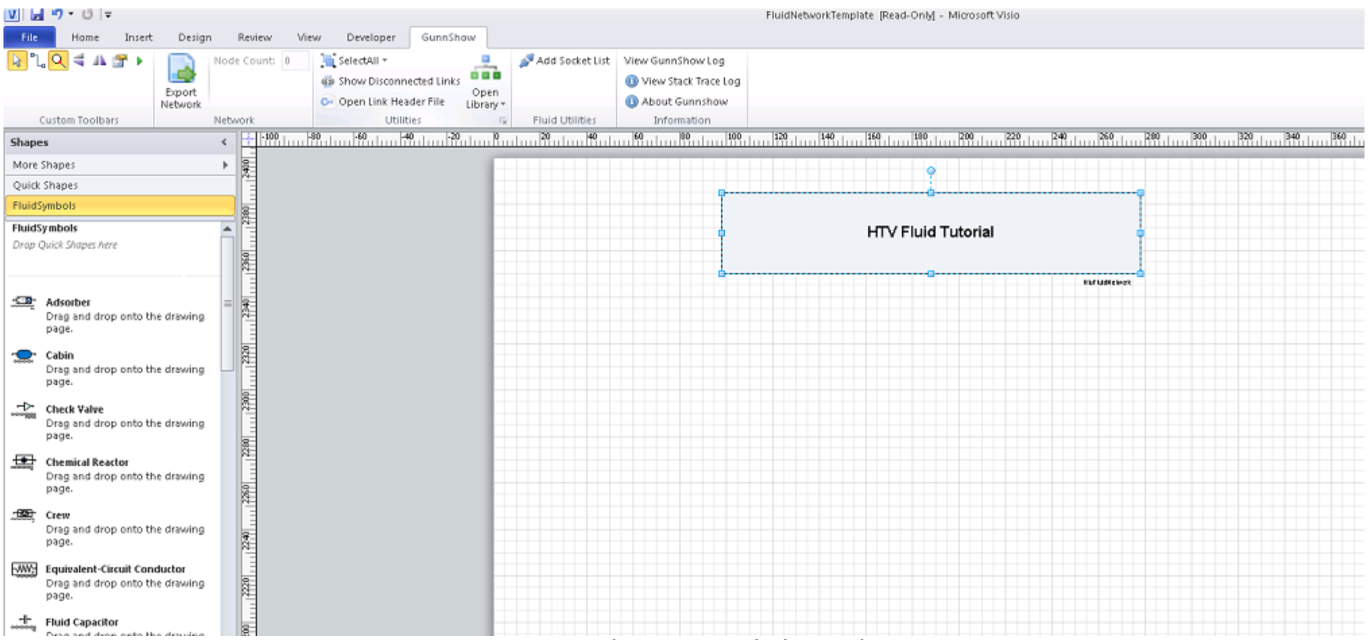
Let's start by giving our fluid network a name. To do this, click on the already existing Title Block shape, and edit the shape data term that has the "Enter a Visio name for this link : ". This field is used by all GUNNS links to give their corresponding C++ object instance a name. Let's name our network HtvFluidNetwork. Editing the Visio Name shape data field in the Title Block is the way to set the name of the C++ class that will be auto-generated by the network export. The names of the files containing the auto-generated class will be HtvFluidNetwork.hh and .cpp. In addition, you can double click the Title Block to enter the text that is displayed within the Title Block. In this example I placed HTV Fluid Tutorial in the shape. GunnShow ignores this information completely, so feel free to put whatever you want in there. Note that a GUNNS network drawing MUST have a Title Block. The network will not have a defined name, and the solver cannot be properly configured without one.

It is important to note that in the case of the Title Block, the object that the shape represents isn't actually a GUNNS link. It is an instantiation of the Gunns solver itself. You can think of the Title Block as a shape that represents an instance of the GUNNS solver to be used for the particular network you are currently building. Take a moment to view the other terms in the shape data window of the Title Block. There is a convergance tolerance, minimum linearization potential, and a minor step limit. It is beyond the scope of this tutorial to discuss what these terms mean. It is enough for our purposes to know that they are configuration parameters for the solver instance. For those that are curious you can read more by consulting the official Gunns solver documentation.




**The Title Block Shape**

Now, before we go any further, let's start fresh on our drawing. To do this, delete every shape but the Title Block shape which we have already configured. You can do this by using the mouse to select all the shapes at once and then hit the delete key. In your normal process of creating GUNNS networks, you might find it quicker to make use of some of the shapes the fluid network template provides you with. GunnShow supports the copying and pasting of links. It also has limited support for copying and pasting nodes. It is very useful to copy and paste in GunnShow. Many times, real-life systems utilize the engineering concept of system redundancy. In these cases, you can often use GunnShow to literally copy and paste nodes and links to build the redundant pieces very quickly. Please take caution when doing this for nodes! A quick check on the node numbers is usually sufficient. GUNNS networks will not successfully export if you have multiple nodes with the same name or number. Sometimes copying and pasting of nodes can run into glitches where your pasted nodes do not get auto-numbered correctly. Just use caution in these cases in order to avoid GunnShow node related headaches.




**Starting Fresh On Our HTV Fluid Network**

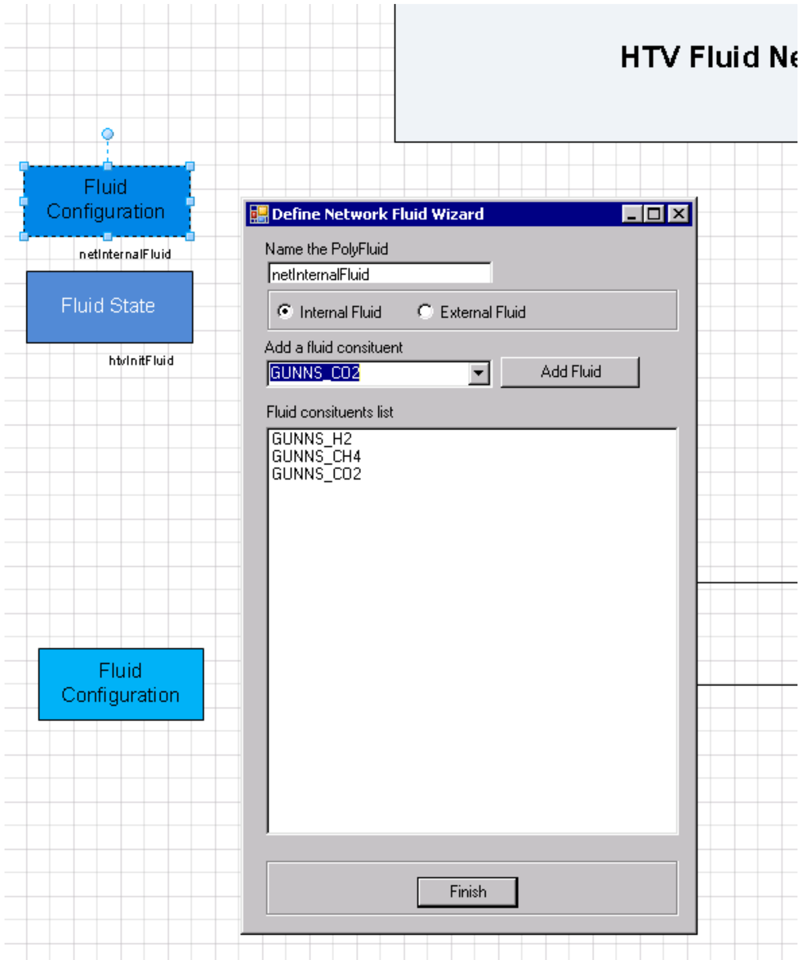
## Configuring Fluids and Defining Fluid States

The next key part to creating GUNNS fluid networks is the process of configuring an internal network fluid, and then building up different states of this fluid to be used as initial conditions for fluid nodes or certain kinds of links. To demonstrate this we will start by creating a model of the HTV Cabin. The cabin is essentially just a control volume that contains a volume of air. In GUNNS, we tend to use a fluid tank class as a model for our pressurized cabins. This is done because the tank class has some additional nice to have features over the standard GunnsFluidCapacitor (which would also work as a simple first-cut model). We will not dive into these additional features much in this tutorial. It is enough to know that certain useful malfunctions are present in the tank class along with a way to take heat flux impinging on the tank shell. This is useful if you want to build a model that is capable of interacting with a thermal aspect.

The first step in this process will be to define a Fluid Configuration. This shape can be found in the Fluid Stencil. It goes by the name of Fluid Config. The shape is a representation of a PolyFluid Configuration Data class. See the PolyFluid class documentation for more information. For our purposes, it is enough to know that every GUNNS fluid network must have atleast one internal fluid configuration. This is used to tell the fluid nodes what kind of fluid constituents are flowing through them. It is basically just the list of all possibile fluid consitutents that will be present in the fluid network. The rule of thumb is that if you anticipate needing to model the amount of Carbon Monoxide in the air of the HTV Cabin, for instance, then you need to include that constituent in the internal fluid configuration of your network.

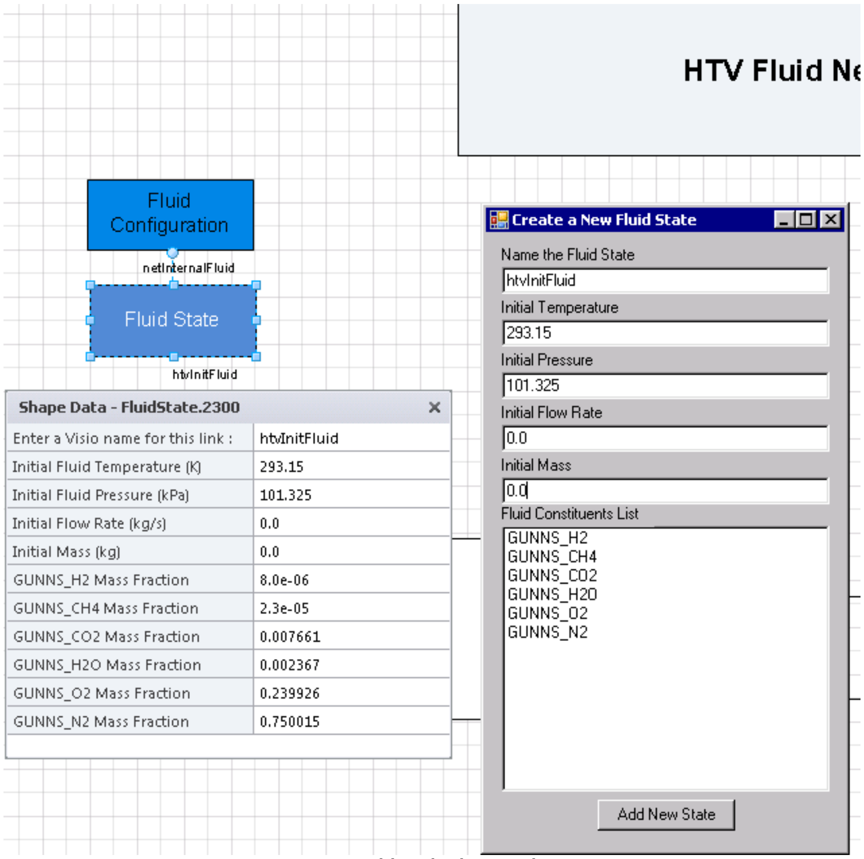
To define a fluid configuration, drag and drop the Fluid Config shape onto the drawing. As soon as the shape drops onto the drawing, a window will pop-up that will help you configure the fluid. We immediately see the name and whether or not the configuration is for an internal or external fluid. Right away, we select internal fluid. All GUNNS networks must have one and only one internal fluid. They can, however, have as many external fluid configurations as they need. This tutorial will not dive into how to use external fluid configurations. They are used to help set up network to network fluid interfacing. See our guide on GUNNS Fluid Network Interfacing to learn more. It is a GUNNS TS21 ECLSS standard to name the internal fluid netInternalFluid. This is done to create a level of consistency across all the GUNNS TS21 ISS ECLSS networks. The name entered into this field becomes the object name of the internal fluid in the GUNNS network. Since every fluid network must always have one, we decided to standardize the name. If you want, you can follow our convention.

The next step is to add fluid constituents. To do this select the drop down menu, and scroll to find the fluid consituent you would like to add. It is another convention to add these constituents in the order of increasing mass fraction. So, the fluid constituents that most often make up a small percentage of the fluid should be added first. If you mess up adding fluid constituents, you can just select finish, then delete the shape that you added, and then repeat the process of dropping a new fluid configuration shape onto the drawing. In our HTV example, we want to model the following gas constituents of Air: H2, CH4, CO2, H2O, O2, N2. To do that, we just select GUNNS_H2, then select add fluid, and then repeat for CH4, CO2, H2O, etc...




**How to Add A Fluid Configuration Shape**

Once a fluid configuration is defined for the GUNNS fluid network, you can start adding Fluid State shapes. A fluid state shape represents exactly what it sounds like, a state of the internal fluid. To be more specific, it is an instance of the PolyFluid Input Data class. See the PolyFluid class for more details .In our example, we want to define an initial state for the HTV Cabin. To do this, we drag a Fluid State shape onto the drawing. Another GUI window pops up and asks for things like the initial temperature, pressure, flow rate, and mass. It also repeats the fluid constituents list which it gets from the internal fluid configuration. It's too late to edit the constituent list. If you aren't satisfied with it at this point, you can close the GUI window, delete both the Fluid State shape and the Fluid Config shape and start over. After you set up the initial temperature, pressure, etc, you can select Add New State and the changes will be reflected in the Fluid State shape. Select the Fluid State shape and view the shape data to verify your inputs were correctly received by your new Fluid State shape. Also, at this point, you can edit all the fluid mass fractions for your fluid constituents for your state. View the picture below to see what values we used to set up our htvInitFluid state.




**How to Add A Fluid State Shape**

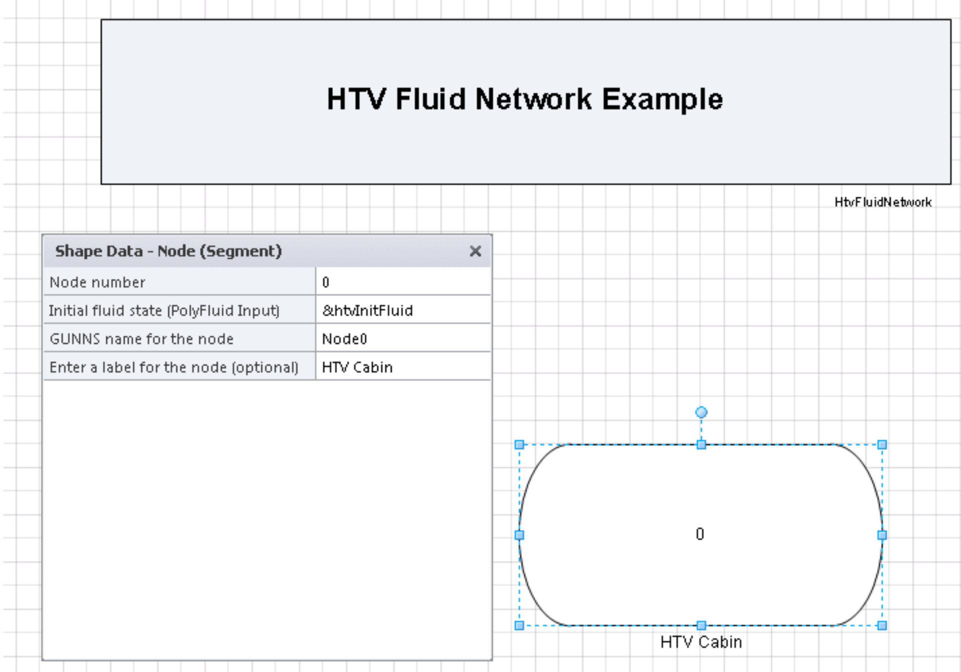
## Utlizing Fluid States in Fluid Nodes

The next step in creating our initial model of the HTV cabin is to create our first fluid node, and then connect up and configure a cabin model. Start by dragging and dropping a Node (Segment) shape onto the drawing. The Node (Segment) shape is the exact same as the Node. The only difference is in the graphical representation. Node (Segment) shapes are typically used to represent large volumes. Things like the HTV Cabin or ISS Node 2, for example. Since this is the first node being added to drawing, you will notice that GunnShow assigns it the node number of 0. GunnShow keeps track of the number of nodes on the drawing, and attempts to help you automatically number the nodes as you go. You can see the node count for the active drawing by looking at the GunnShow ribbon. You will notice that you cannot manually edit the Node number field. There is a way to do this should you encounter a situation where you need to. See the page on Manual Node Number Management LINK HERE. In theory, if you are careful as you add new nodes, you should not need to manually edit the number. In practice, it is common to do many copy and paste operations on nodes. This sometimes makes it difficult for GunnShow to properly keep track of the node numbers. It is a good idea to always immediately check your node numbers as soon as they are added, and be sure that you do not have duplicates. If you encounter a situation where you must do manual node number edits, see the page mentioned above.

Let's review the Fluid Node's shape data more carefully.

1. Node number: This is the GunnShow auto-generated node number. It is filled in for you automatically when you drop a node shape on the drawing. It does this by taking the current node count, adding one to it, and then placing that number into the Node's shape data. GunnShow locks this field to prevent easy editing. The field can be manually edited should you need to by viewing the procedure here ... LINK HERE.
2. Initial fluid state (PolyFluid Input): This is the fluid state that this fluid node will be initialized to. To properly fill it out, you must give the field a reference to a properly configured FluidState shape that exists in the drawing.
3. GUNNS name for the node: This is the name given to the node in the network manager's node enumeration list. By default we use Node# (with # representing whatever number is in the Node number field). The user can specify a custom name here if they like.
4. Enter a label for the node (optional): This field allows you to add a label to the node. Sometimes, if a fluid node represents a physical volume (like the HTV Cabin node) it is useful to add a label to the node. This field does not affect the network in any way. It is just an additional graphic sometimes used for making your drawing more descriptive.

With that information in mind, you can now configure your newly added HTV Cabin node. We gave our HTV Cabin node a reference to our initial fluid state "&htvInitFluid". You may need to use a different value if you named your initial fluid state something different. See the image below to help verify your inputs.

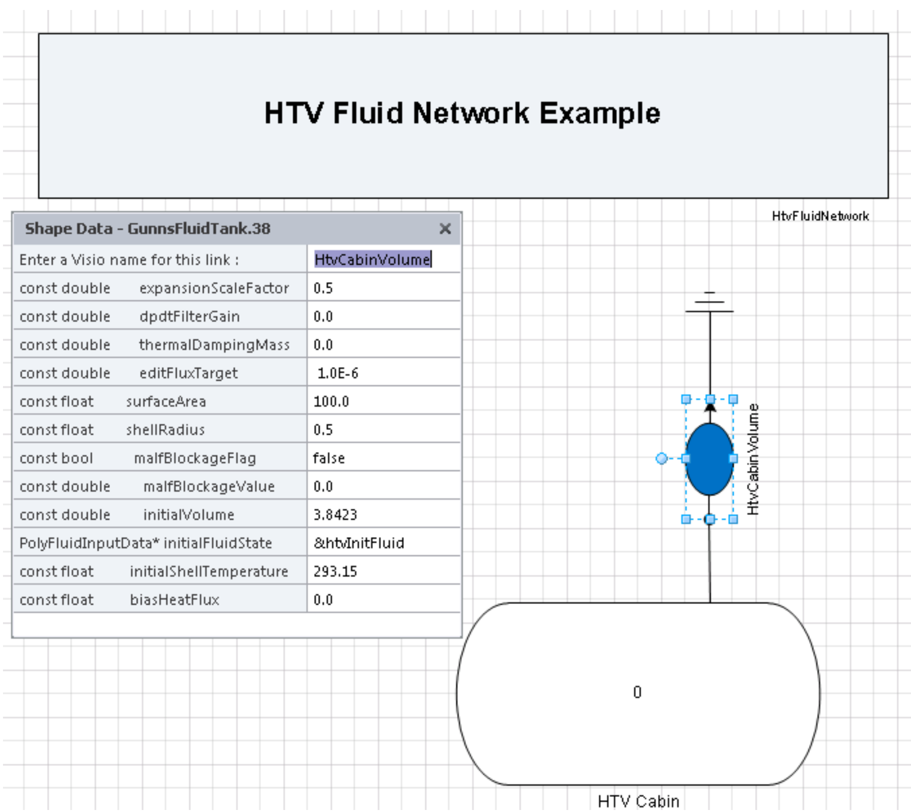



**Configuring Your First Fluid Node**

## Adding Our First Fluid Link

Now that we have an HTV Cabin node, we'd like to add capacitance (or volume) to the node. You'll notice that in the FluidState shape that we never specified the volume of the fluid as an input parameter. GUNNS uses capacitive links in order to give nodes their capacitance. In fluid networks, a GunnsFluidCapacitor, or any class that derives from it, is used to give a fluid node its initial volume. During run-time the capacitive link manages this volume and can make updates to it, and that information is then passed back to the fluid node it is connected to and consequently, the core solver. For example, to give our HTV Cabin node volume, we will use a GunnsFluidTank link which is derived from a GunnsFluidCapacitor. Refer to each class's documentation for more information on these two links.

Select the Tank in the FluidSymbols stencil and then drag and drop it on to your drawing. Often times, new GunnShow users will drag and drop links onto their drawing and then have no idea how to properly hook it up and configure it. Early GunnShow users would either ask more seasoned GunnSlingers, or they would refer to already completed drawings to see how to properly hook up the link they just added. Since then, GunnShow has begun to ramp up its documentation efforts, and has come up with a better solution. Visit the how to view GunnShow link documentation guide for more information. LINK HERE. This tutorial will assume you know how to get to the information on how to hook up and configure any GunnShow link. In any case, view the picture below to see how we connected up and configured the Tank. The special things to note are the initialFluidState field, and the way the link is connected to a Ground Node. This tutorial assumes you are familiar with Ground Nodes, and that you will look to the Tank link's documentation to learn why it is hooked up the way it is. If there are any doubts as to what certain shape data fields are for the link (dpdtFilterGain for example), we again urge you to view the Tank link's documentation. Remember, each link in GunnShow is a GunnsBasicLink derived object. Some are fairly straight forward to set up and use. Others, however, require time and research in order to properly understand them and put them to good use. It is our hope that they are all one day documented well enough that an engineer with little prior knowledge of GUNNS can quickly read up on a link and then begin to put it to use in a network with GunnShow.

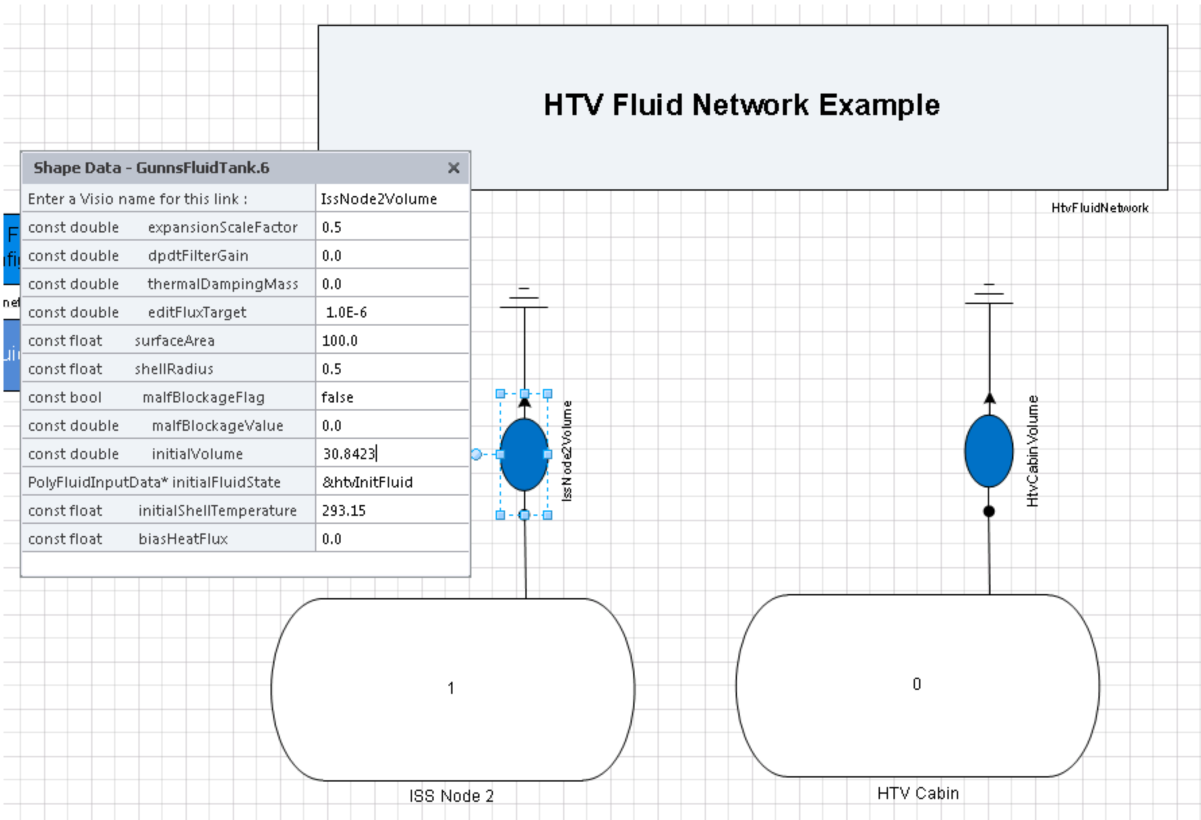


**Configuring Our HTV Cabin Volume Link**

## Using Copy-Paste To Speed Up Network Creation

We now have a properly confgured model of the HTV Cabin volume. In fact, we have an entire network. At this point, if you'd like to, you should be able to export this network and then get it running in a sim. See our documentation on how to do this. LINK HERE. Obviously it's a pretty boring network at this point. Let's use the copy-paste feature of Visio to quickly change that.

We'd like to add in the stub model of the ISS's Node 2. This model can use the same Tank, Node, and connections that the HTV Cabin used. Obviously we will have to configure the ISS Node 2 stub model slightly differently than the HTV Cabin model. To do this very quickly, we will use copy-paste. Use your mouse to select your Ground Node, HTV Cabin Link (along with its connectors), and the HTV Cabin Node. Then use ctrl-c, ctrl-v (or obviously right-click copy, right-click paste) to copy and paste the nodes, link, and connectors. From here, all you should have to do is arrange the ISS Node 2 link, nodes, and connections the way you want them, and then change the names, labels, and volume parameters. You can make a separate FluidState for ISS Node 2 if you like. Just follow the procedure from the Utilizing Fluid States in Fluid Nodes section. In our example, we will just initialize Node 2 to the same initial fluid state as the HTV. View the picture below to see how we configured our copy-pasted ISS Node 2.

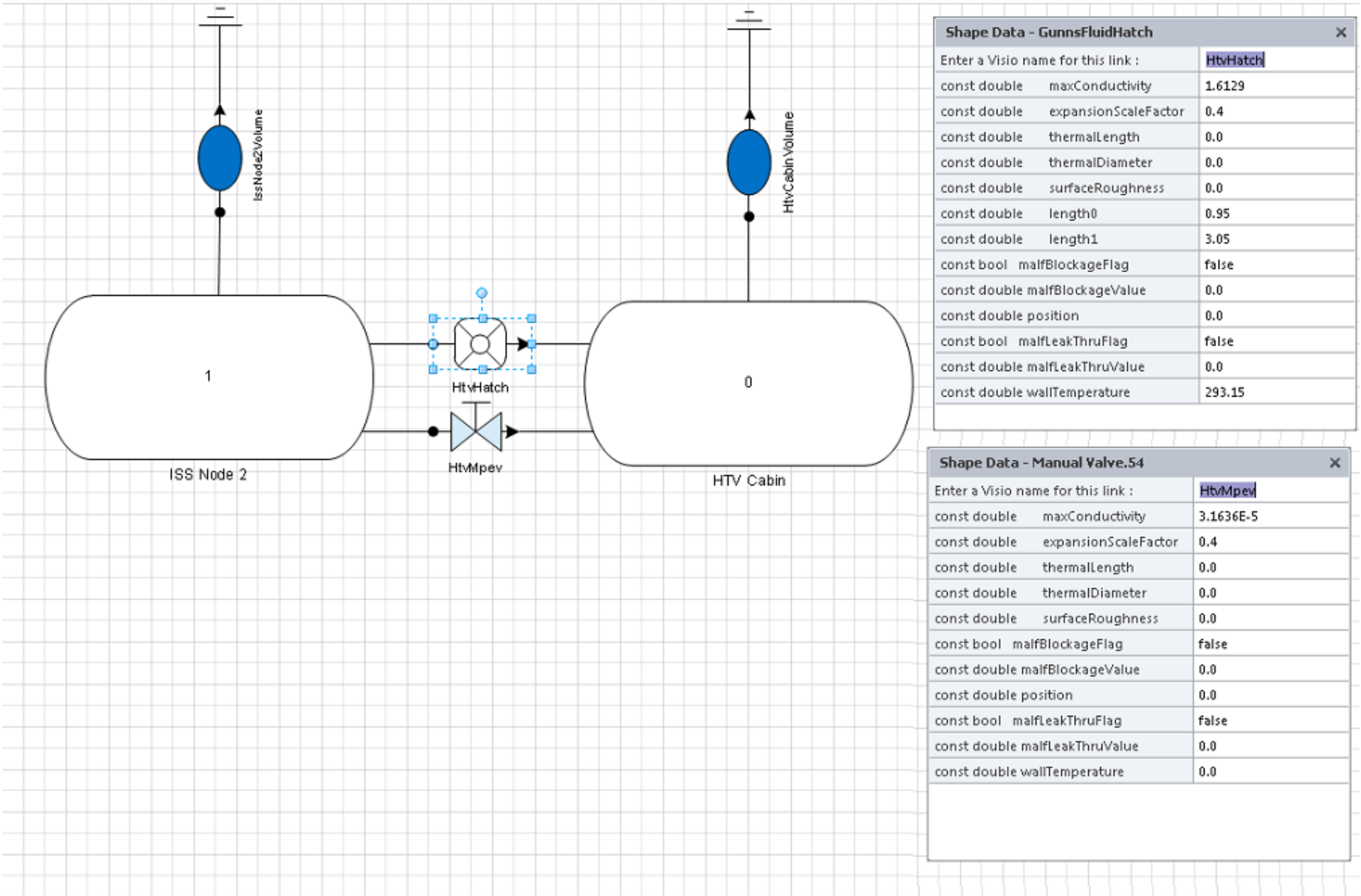



**Copying and Pasting An ISS**

Obviously copy-pasting is a nice feature to use for building GUNNS networks quickly in GunnShow. It is especially useful for rapidly implementing redundant systems. Be warned however, with great power comes great responsiblilty. If you use copy-paste to copy many nodes and many links, please be sure to verify all of your node numbers were incremented correctly! Use the manually editing node numbers procedure if they were not. LINK HERE.

## Adding A Hatch and Manual Pressure Equalization Valve

Again, at this point, we have another complete network. This time with two nodes, and two capactive links. It is still a boring network, however, because the two volumes have no way of interacting with one another. They are currently isolated nodes in the network. Let's change that by adding two links that will allow air to flow from one node to the next. The type of GUNNS fluid link that allows for flux in response to a change in potential is called a GunnsFluidConductor. The fluid conductor class derives from GunnsBasicConductor. Any link class that derives from GunnsFluidConductor or GunnsBasicConductor has the ability to add their own implementation to modify the flux through the link. This can depend on the conditions of the nodes that it is connected to and on its own internal state. The Hatch and Manual Valve GunnShow shapes are objects of the GunnsFluidHatch and GunnsFluidValve class respectively. Both are derived classes of GunnsFluidConductor. View the class documentation for more details. We will use the two links to model a Hatch and a Manual Pressure Equalization Valve that will allow air flow between Node 2 and HTV. Like we have been doing, drag and drop each shape on to your drawing (use the Hatch and Manual Valve shapes from the FluidSymbols stencil). See the image below to see how each link was connected up and configured.



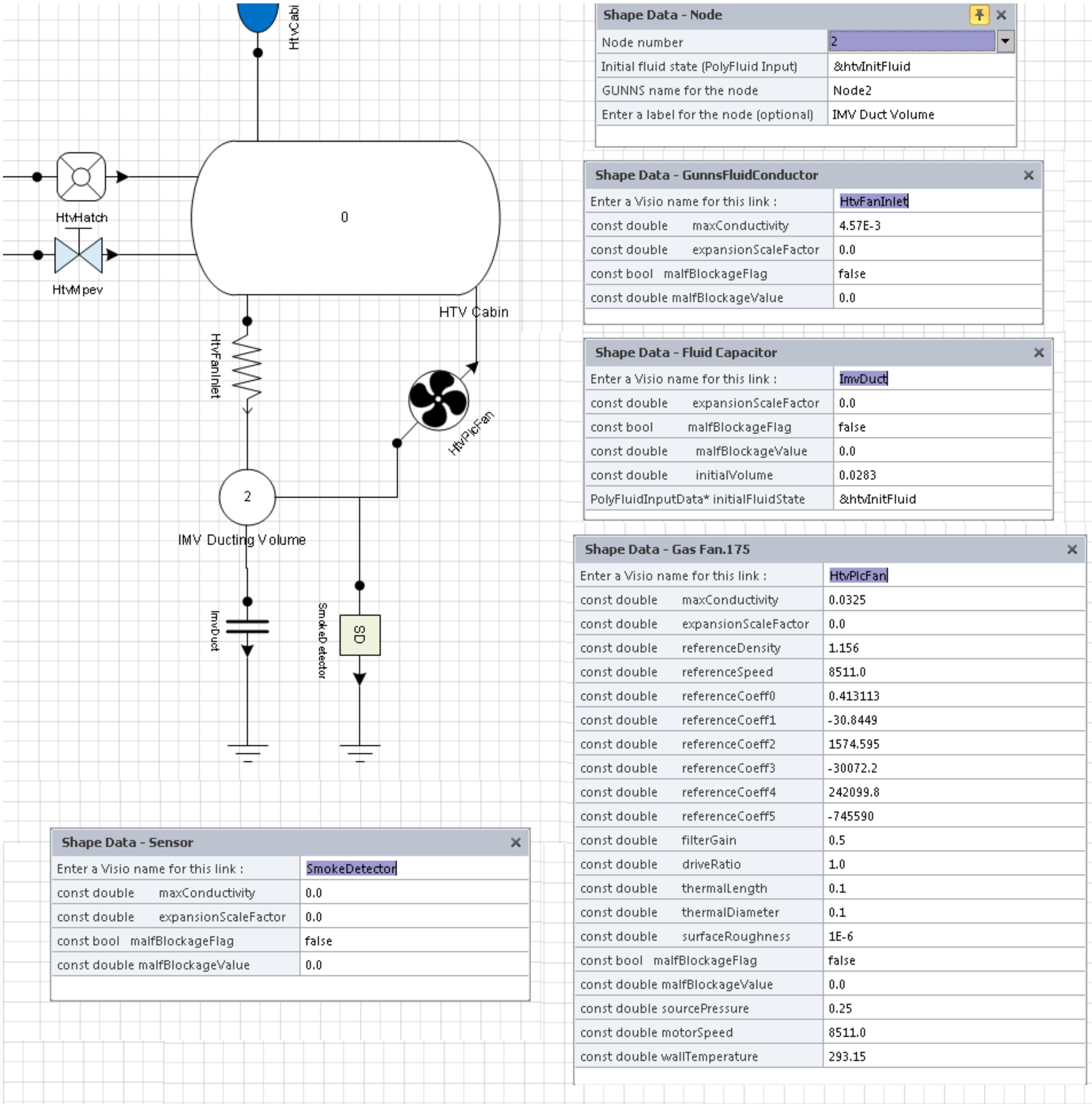

**Configuring A Hatch And MPEV**

Our network is finally starting to take shape. At this point, you should be able to get this into a Trick sim, and play around with the HTV and Node 2 pressures, along with the MPEV and Hatch position and see flow from one node to the next. While you're at it, take a look at the flux and flow rate terms in the Hatch and MPEV. Also observe how temperature varies in response to an expansion or compression of air in the nodes. Use Trick-View to see how the names you gave the links in Visio directly map to the name they have in the sim. Take a look at all of the internal state variables for your links. Many of these can be changed on the fly.

## Adding An Air Circulation Loop

The next key piece of our HTV ECLSS model is an air circulation loop. In more complex ECLSS systems air circulation loops are commonly used within an ARS (Air Revitilization System). This is a loop that air is forced to flow around that does things like cool the air, sample the air for toxins and fire products, scrub the air of pesky fluid constituents, etc... In our example, we keep our air loop simple. Its only components are a GunnsFluidConductor link to model the ducting, a capacitive node to model the volume of the ducting, a GunnsGasFan link to model the HTV Cabin Fan, and a GunnsFluidSensor to model the Smoke Detector. Some of these links you should already be pretty familiar with. The GunnsFluidConductor and GunnsFluidCapacitor have already been discussed in this guide. The new links here are the GunnsGasFan (shape name Gas Fan) and the GunnsFluidSensor (shape name Sensor). The Sensor is a very simple link. It is basically a GunnsFluidConductor configured to allow no flow through it (exception to this when it is used as a delta-pressure sensor). It exists primarily to provide an easy interface for getting at the fluid properties of the fluid nodes it is connected up to.

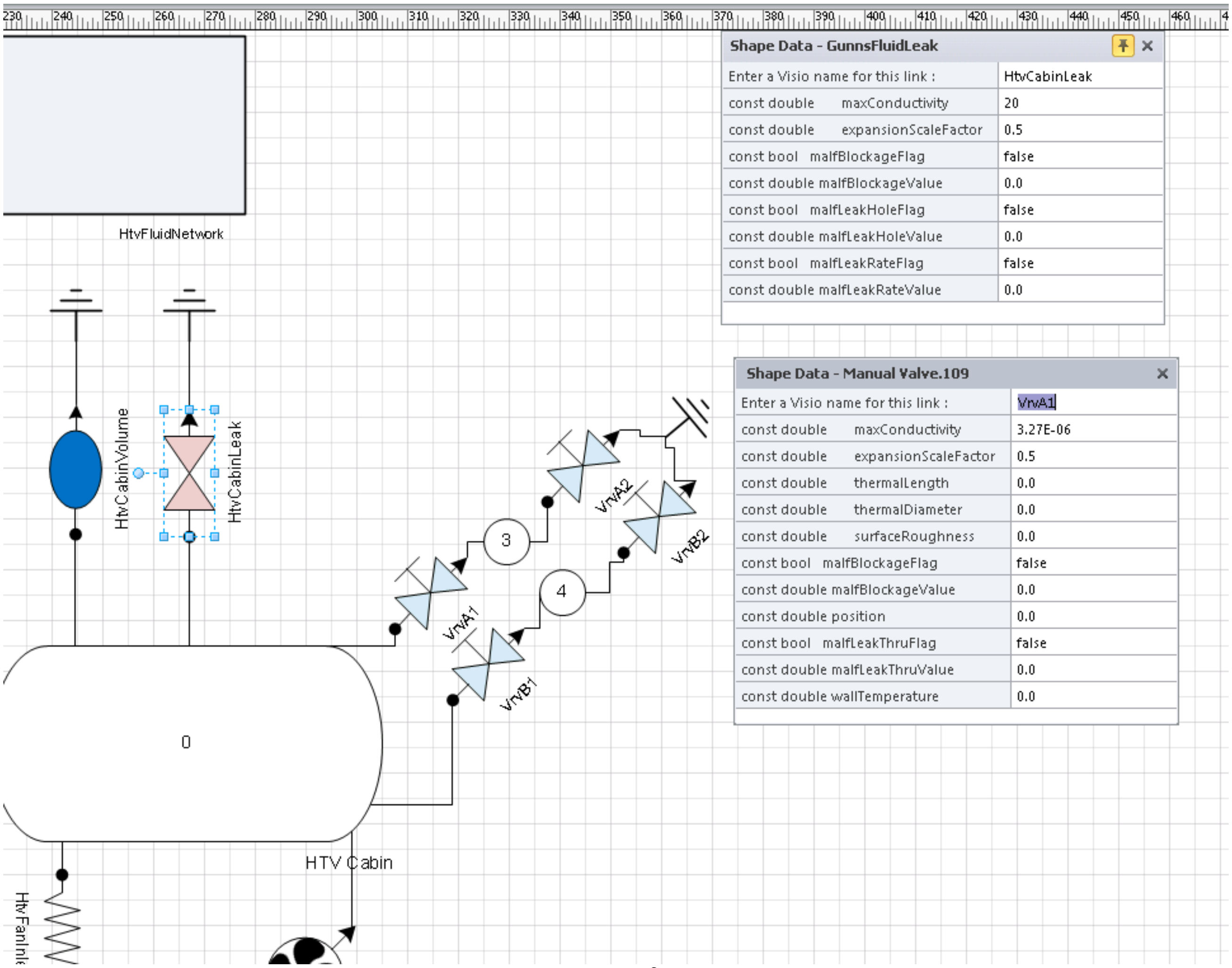
The Gas Fan shape is a much more complex shape. It is a derived class from GunnsFluidPotential. So far we have discussed capacitors and conductors. Now it is time to briefly discuss potentials. All potential links in GUNNS derive from GunnsBasicPotential. They are used to create potential differences between the nodes that they are connected up to. In the fluid world, this means a delta-pressure. The Gas Fan is a somewhat involved link that creates the right delta-pressure between the two nodes it is connected up to in order to produce a flow rate through the fan. Much more detail can be found by looking at the GunnsGasFan documentation. View the image below to see the configuration values we used for our HTV example.




**HTV Air Loop Configuration**

## Adding A Vent Relief System and A Cabin Leak

We're nearing completion of our HTV model. The last pieces to add are four Vent-Relief Valves and a model to represent an HTV Cabin leak. The Vent-Relief Valves are simply another set of Fluid Conductor's. We will use the same Manual Valve shape that we used for the MPEV. The Cabin leak model is also a simple Fluid Conductor that is hooked up to the HTV Cabin and Ground. Ground is taken to represent space in this instance. Ground is at zero potential, so it works as a pretty good representation of the space environment. Follow the established steps and see the image below to complete your HTV ECLSS model.




**HTV Air Loop Configuration**

## Conclusion

At this point you should be able to click the Export button in the GunnShow ribbon, and get a C++ generated GUNNS manager class that is a representation of your HTV model. If you get an export error, be sure to check the GunnShow log file for further information. Most errors on export are easily fixed. If you struggle to understand the error, be sure to let the GunnShow developers know.

From here, the next step is to verify your model within the Trick simulation framework. This process is generally an iterative one. As you attempt to verify certain expected system behaviors, you will inevitably notice shortcomings in your model. Then you go back to the GunnShow drawing and start tweaking parameters, swapping low-fi links for high-fi links, changing link connections, etc... Each time you make a change to the drawing be sure to save and then re-export before re-attemping to build your Trick sim.

At this point, you are hopefully somewhat comfortable with how you might model a real-life fluid system in GUNNS. In any case, this type of modeling can be tricky. GUNNS and GunnShow were designed to be tools to make the process easier, but they will not solve everything for you. It is important to understand what is happening within the GUNNS solver and the associated link models that the GunnShow user is connecting together to form a system. Be sure to reference the GUNNS link and solver documentation when you have trouble. You may have come into this tutorial a little GUNN-shy, but hopefully you are leaving feeling more confident as a young-GUNN. Best of luck in your future GUNNS modeling. Reach for the sky!

**DEVELOPERS:**

| Michael Moore | Nick Kaufman | Carlo Bocatto |
|---|---|---|
| *E-mail: michael.m.moore@usa-spaceops.com* | *E-mail: nicholas.kaufmann@nasa.gov* | *E-mail: carlo.t.bocatto@nasa.gov* |