



Building a Database

Creating a new site from scratch
with ARDI

Abstract

This document covers both the **how** and the **why** of creating an ARDI database for a sample site.

Copyright © Optrix Pty. Ltd, 2016-2023
All rights reserved.

No part of this publication may be distributed, or transmitted outside the organisation owning an ARDI license in any form or by any means, including photocopying, recording, or other electronic or mechanical methods, without the prior written permission of the publisher, except in the case of noncommercial uses permitted by copyright law.

About

What Does This Document Cover?

This document covers both the concepts behind ARDI and how you would create a simple ARDI database.

It's lengthy, because it covers *why* you make key design decisions, not just *how* you perform the tasks.

Assumed Knowledge

Some experience using the ARDI web interface may be useful, but is not required.

Resources Needed

The reader will need access to a blank ARDI server.

Table of Contents

Introduction	1
Logging In as an Administrator	Error!
Bookmark not defined.	
The Blue Widget Factory.....	3
Assets	6
Relationships.....	7
Creating Relationships	9
Properties	12
Creating Properties.....	14
Types.....	17
Creating Types	19
Adding Data to Types.....	21
Hierarchies.....	23
Creating Assets	26
The Water System.....	30
Property Flow.....	32
Small Relationships.....	34
Hydraulics	35
Bringing The Assets Together	36
Linking to Data	37
Digital Properties	44
Enumeration Properties.....	46
Adding Symbols	47
Colouring Your Properties.....	48
The Second Boiler	49
Making Bulk Changes.....	50
Enabling Augmented Reality.....	51
Making It 3D.....	52
Appendix – Fixing Common Errors	55
Cleaning Up Afterwards.....	57

Introduction

Why have such a big tutorial?

Although ARDI is not a particularly complicated system, it can take a little bit of time to fully understand what it's capable of and how it works.

The individual components and features are easy to explain, but it's hard to get an idea of how it all comes together into a complete ARDI system.

This is where the tutorial comes in – it will walk you through the process of creating a simple ARDI database and connecting it to data.

This should let you get an idea of how to perform all the important steps to both create and maintain your own ARDI database in the future.

Learning to think 'ARDI'

It may take quite a few pages before you're **doing** something in this tutorial. This document is just as much about *why* you are doing something as *what* you are doing.

The biggest hurdle in understanding how to make an ARDI database isn't how to add assets – is how to think about your system and prepare before you add your first asset to the system.

It takes some time for people – particularly engineers, who are used to thinking of the plant and process a particular way – to get comfortable with ARDI's focus on human-readability.

The tutorial tries to explain why we do something and how it works before we perform the steps required.

This tutorial also teaches best-practices. Some of the steps it goes through are quite complicated for such a small application. We go through these steps so you'll be armed with the proper techniques to make much larger ARDI databases with a minimum of time.

Is there a faster way to do this?

Yes. This tutorial concentrates on how to do all the processes *manually*, which is the slowest way to create your ARDI database.

In most real-world situations, you'd take advantage of ARDI's import features to dramatically speed up this process.

Actions To Follow

If you see a section that looks like this, it's highlighting a set of actions that you should follow on your own ARDI database.

Getting Started

Files You'll Need

This tutorial will use several components that are normally downloaded from the Internet.

These files can be downloaded from <https://ardi.com.au/downloads/training/TrainingBundle.zip>

The file includes a 3D model (**Site Builder Model.zip**) and a library of symbols (**Site Builder Symbols.ardisym**).

You'll need these on the system you're using to configure ARDI.

Offline Installation

If your ARDI server doesn't have access to the internet, you will have to find a way to install some ARDI components onto your ARDI server.

In the training bundle file (downloaded in the previous section), there's an **offline_installation** folder specifically for ARDI servers that can't connect to the Internet.

Copy the contents of this folder to your ARDI server, placing them in your ARDI installation directory.

SYSTEM	INSTALL FOLDER
WINDOWS	C:\Program Files (x86)\Opatrix\ARDI, or C:\ARDI
LINUX	/opt/ardi/

This will install the drivers, addins and sample data you'll use during the tutorial.

Signing In to ARDI

To begin, you'll need to log in to your ARDI system. This will have been setup by your system administrator and they will provide you with the server name and location.

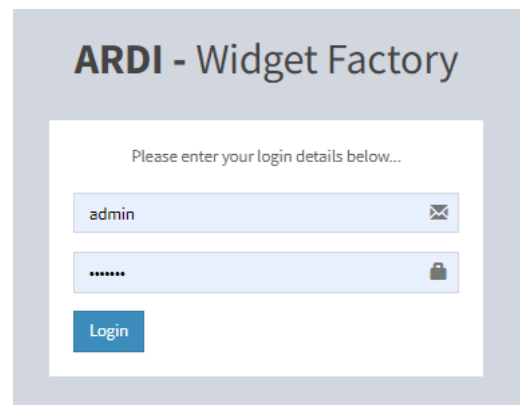
In a brand-new ARDI installation, the default username is **admin** and the password is **default** (case-sensitive).

We suggest changing this as soon as possible.

If your server is in guest mode (meaning it can be accessed without user authentication), you'll find the your user name – Guest – at the top-right corner of the page.

Click on **Guest** and hit the **Sign In** button that will pop up.

Then enter your username and password to begin.



The Blue Widget Factory

Introducing Our Plant

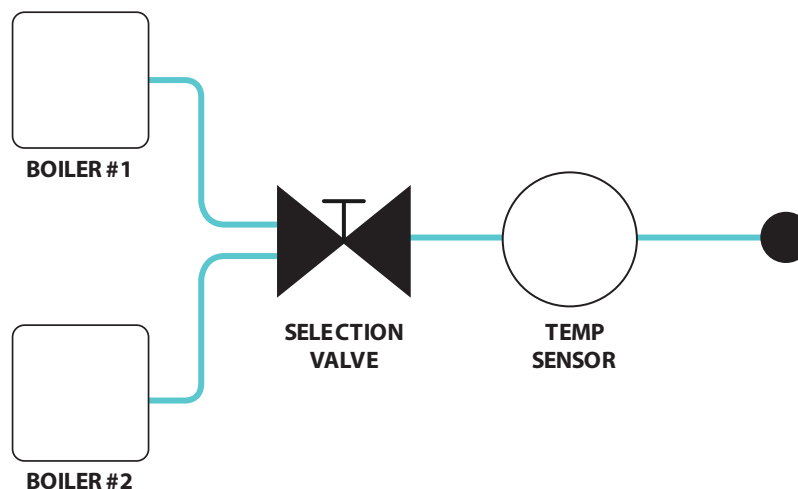
For our tutorial, we are going to create a database to represent the high-pressure steam system for the Blue Widget Factory.

Note that this is an extremely stylised version of the process. Many parts of a real steam system are missing, and the historical data we will use is completely made up.

We are using it simply as an example of a small but highly-connected system.

The Boilers

The steam system contains two boilers that burn natural gas to heat water to produce steam.



The two boilers are quite simply called **Boiler One** and **Boiler Two**.

The two boilers merge in a selector valve and pass the steam onto the 'Steam Outlet', which delivers steam to the rest of the plant (which we won't be concerned with in this tutorial).

What's Inside a Boiler?

Each boiler has several parts.

It has a **vessel**, which is the tank for the water.

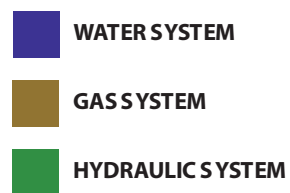
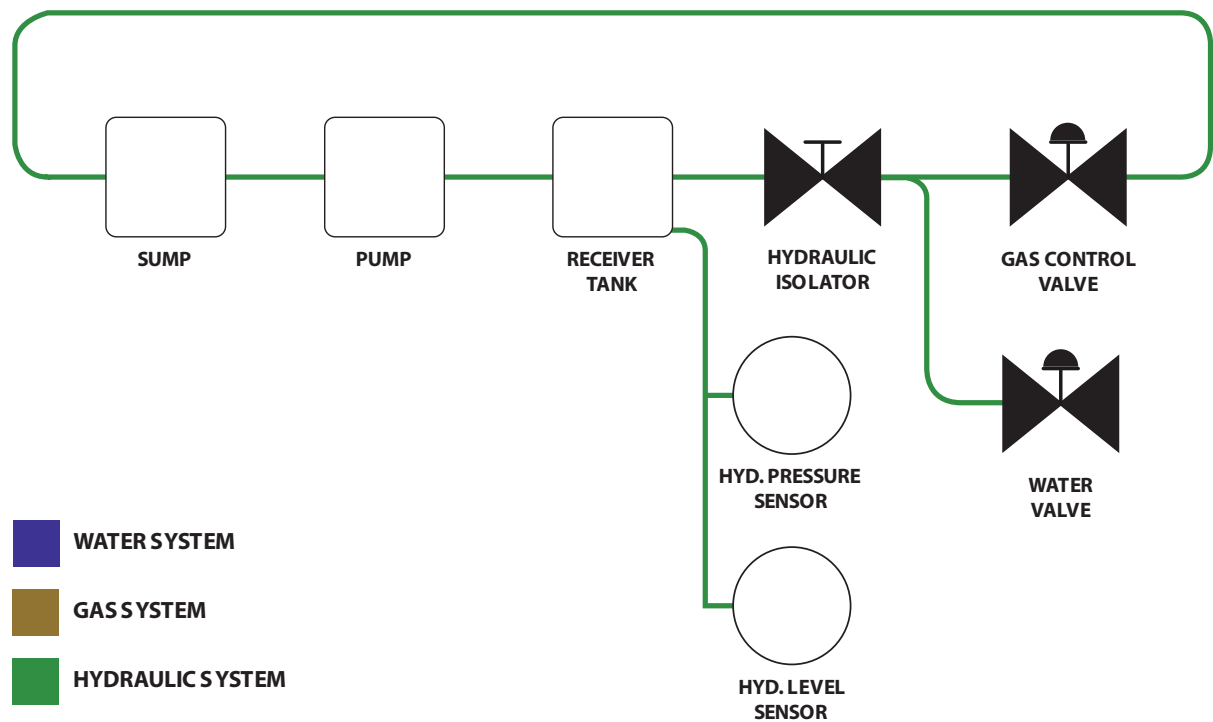
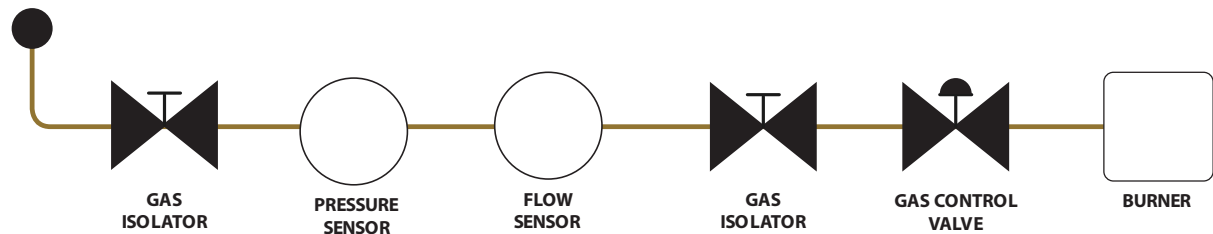
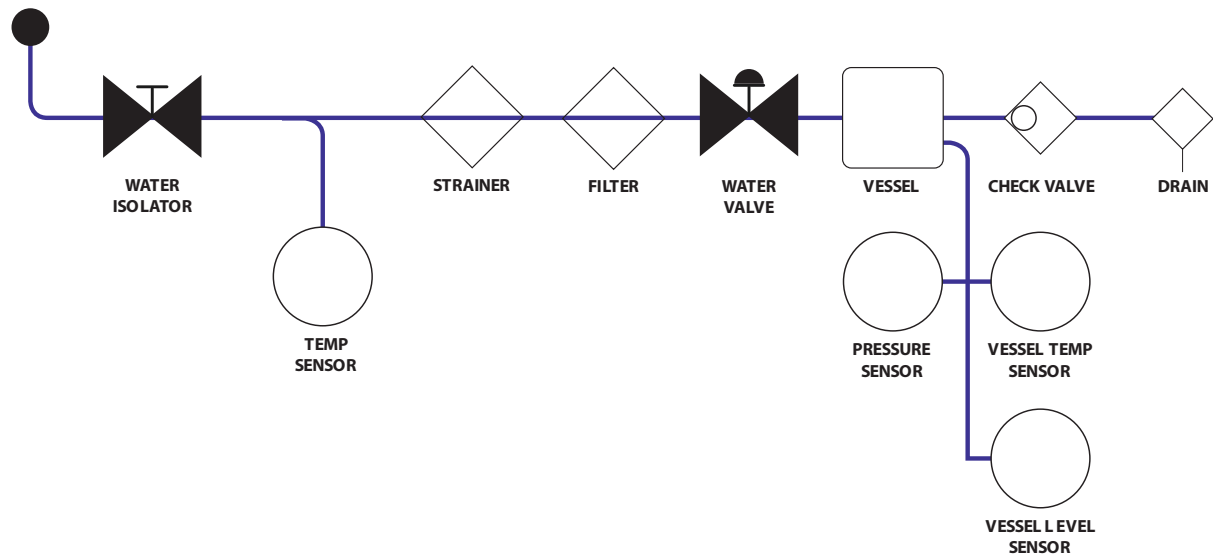
The vessel is filled by the **water system** – this takes water from your supply and makes sure the tank is always full.

Below the vessel is the **burner**, which burns natural gas to heat the water. It is fed by the **natural gas system**.

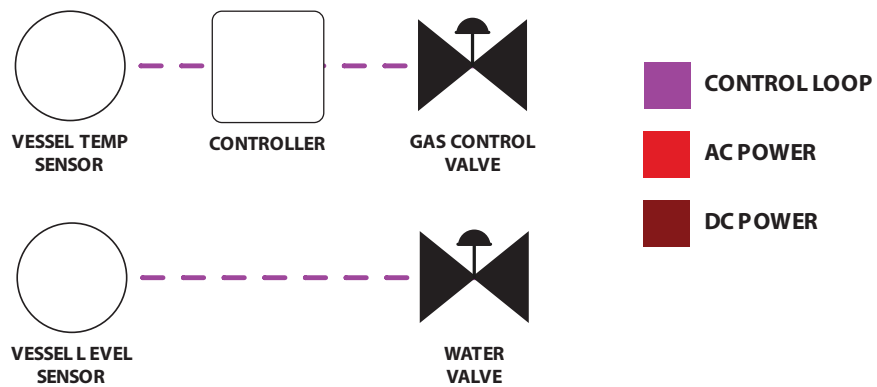
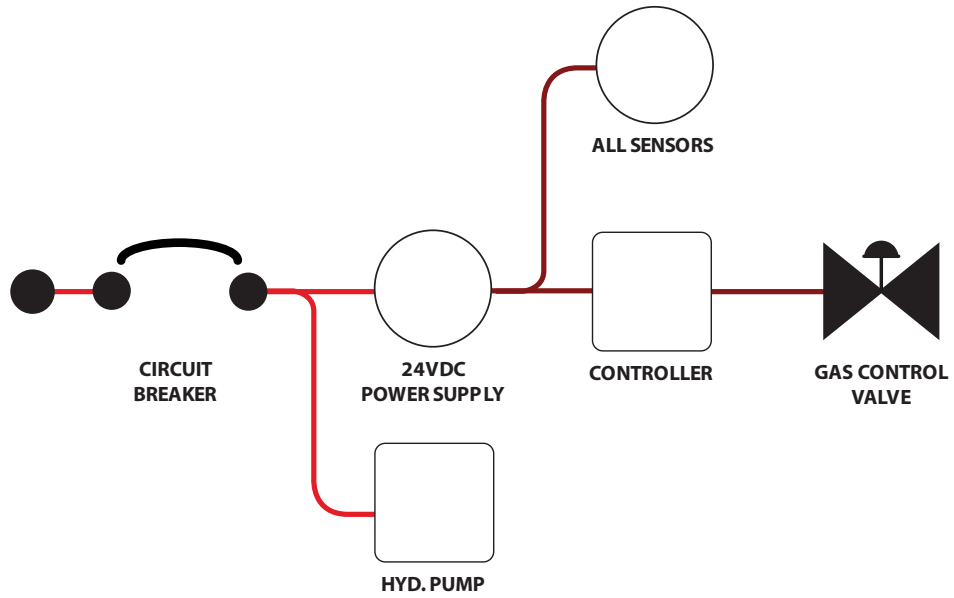
To control your temperature, we use a **controller** (which controls the amount of gas flow to keep the temperature and pressure of the vessel under control), and a **hydraulic valve**, which opens and closes to control the flow of gas. This means that we need an **electrical system** and a **hydraulic system** for each burner, too.

The diagrams for each of these are shown below.

P&ID Diagrams



Electrical & Control Diagrams



Assets

What Are Assets?

ARDI is focused on your *assets*.

In ARDI, an asset is basically anything. Machines, components, valves, data acquisition gear – even people. When an asset represents an object, it's called a *physical* asset.

However, an asset doesn't have to be a physical thing at all – an asset can represent something more abstract, like an area of your plant or production, a department in your organisation or a group of similar items. These are known as *logical* assets.

If you'd like to talk about it, it should be an asset in ARDI.

Physical Assets

Your physical assets are the easiest to identify. They represent actual pieces of plant hardware, such as sensors, machinery or buildings.

If you were to go back over our Blue Widget diagrams, you'll see that each shape represents a physical asset we would need to create in ARDI.

Logical Assets

Your logical assets are a little less obvious – normally you set these up when you design your *hierarchies*. This topic is covered later, but for now just know that some of your assets can represent *ideas* and *groups* of objects, such as 'The Hydraulic System'.

Properties and Media

As well as their own name, assets have *properties* and *media*. Properties are simply details about the asset - if the asset represents a temperature sensor, the properties attached to the asset could include...

- The current temperature,
- The manufacturer and model number of the device
- The last calibration date of the device

Media are files and web-links to further information that might be of use, such as...

- Procedures and safe work instructions
- Manuals and data-sheets
- Calibration and inspection certificates

Picking Some Assets

Now that we know what an asset **is**, we can choose some. From our drawings, we can assume that every one of our symbols will probably be a *physical* asset.

Relationships

What Are Relationships?

Relationships are connections between your assets.

Each relationship has a start point, an end point and a direction.

For example, a light switch feeds power to a light-bulb in the ceiling, so in this case the 'Power' relationship between the two moves from the switch to the bulb.

Like assets, there are two distinct types of relationship in ARDI.

Physical Relationships

Physical Relationships define actual, physical connections between assets, such as power cables, network connections or piping.

Process, wiring and pipework diagrams are a common source to identify physical relationships. They can also be used to represent 'transparent' connections – although a WiFi network can't be physical seen, it can still be considered a physical relationship because volumes of data move through it the same way that water moves through a pipe.

Logical Relationships

Logical Relationships describe non-physical associations, such as describing concepts such as...

- Control loops between assets
- The order of assets in a system
- The sequence that assets start up or shut down

These sorts of relationships are usually not obvious when you are out in the field, since there isn't the same visible connection as found in physical relationships.

Although they might seem minor, capturing the logical relationships between your equipment can be extremely powerful when it comes to visualising your data later.

Picking Your Relationships

Before we start creating our assets, it's a good idea to create our relationships.

Physical

Like physical assets, it's quite easy to pick out our physical relationships from any diagrams we have of our process.

For every distinct type of *exchange* (for instance, water flow, power flow, information etc.) between two assets, you should create a new relationship. For example, our Blue Widget has both cable and piping connecting the various components.

But we don't want relationships called 'Pipe' and 'Wire' – it's not the type of connection that matters, it's what is flowing *through* that connection.

For instance, there are several different items connected by wires. These include...

- AC Power (bright red)
- DC Power (dark red)
- Signals (purple dashed)

So keeping in mind that the relationship is about the **content** of the connection rather than the connection itself, our drawings suggest the following relationships...

- AC Power
- DC Power
- Natural Gas
- Water
- Steam
- Hydraulic Oil

Logical

Like logical assets, some of these are discovered as you start building your ARDI database and find assets where you'd like to provide more detail.

One logical relationship we will almost certainly need is **Control**.

Although the control signal is sent along a physical connection (ie. as a signal on a wire or Ethernet cable), the relationship itself bypasses all of the physical components involved.

For example, the real control signal might go through a remote I/O module, to the PLC, to a second remote I/O unit and out to the target device.

But the Control relationship doesn't include any of that detail. It simply says that 'A controls B'. As such, it's a good example of a logical relationship.

So now that we've decided which relationships we are going to need, it's time to actually create them.

Creating Relationships

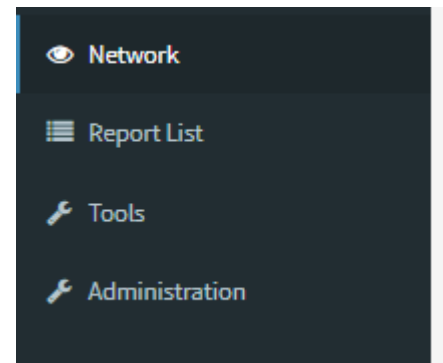
Opening the Admin Page

Let's finally get our hands dirty and start adding some relationships to ARDI.

To begin, we will need to make sure we are logged in as an administrator. If you're unsure how to do that, you should see *Signing in to ARDI* on page 2.

Then, you'll need to open the **Administration** page.

The Administration Page is where you access all the major functions of ARDI outside managing your assets.



Properties, relationships and data acquisition are all set up in Administration – it's where you customise your ARDI database to suit your particular application.

You can reach the Administration page by clicking on the *Administration* button in the sidebar (shown above).

Menu Shortcuts

In many places, we might give you a *menu shortcut* – a quick way of describing how you could navigate to a particular page.

For example, we might tell you to go to **Administration | Properties**, which means you should first go to the *Administration* button, then on that page you should press the *Properties* button.



To add new relationships, you'd need to open **Administration | Relationships**

The Relationship List

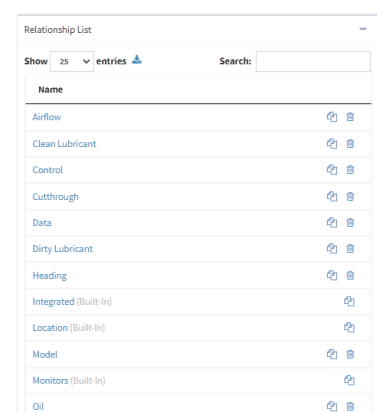
This will get you to the Relationship List.

Here you'll find all of the relationships you currently have in ARDI.

Since this is a new system, there won't be much here – often simply the **Location, Monitors, Integrated** and **Type** relationship. As we build your ARDI database, this list will start to get longer.

There are several different list pages in ARDI, and each of them work the same way.

- Edit an item by clicking on its name.
- Delete an item by clicking on the 'trash' icon to the right.
- Click on the two pieces of paper to duplicate the item.
- Type a name into the 'search' bar to search for a specific item
- Add a new item by pressing the 'Add New' button.



Adding a Relationship

Each relationship has several options you should fill in.

Name: A unique name for this relationship that indicates what it represents.

‘Signal’, ‘Hydraulic Oil’, ‘Air Supply’, ‘AC Power’ are all good names for a relationship.

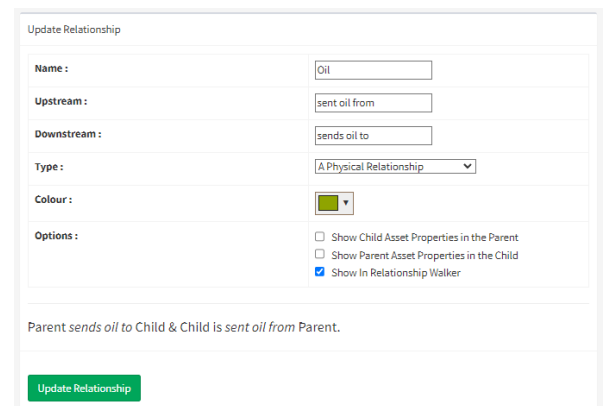
Upstream: The *Directional Name* (see below) for upstream assets.

Downstream: The *Direction Name* (see below) for downstream assets.

Type: Choose between *Physical*, *Logical* and *Hierarchy* (hierarchy relationships won’t be covered in this document, but are used to create alternative organisational structures for your assets).

Colour: The colour to use to draw or otherwise represent this relationship.

Options: Relationships have a few options to control how they (and the data in the assets they connect) will behave. For now, ignore these options.



Directional Names

Every relationship has two directional names. These are the words inserted between the parent and the child or the child and the parent when describing the relationship in plain English.

If the relationship is parent-to-child, the words are inserted without modification. If the relationship is child-to-parent, it will be prefixed by ‘is’.

In our AC Power relationship, we may want an upstream name of ‘Powered By’ and a downstream name of ‘Powers’.

When describing the relationship in English, ARDI will then say that...

- Asset A is powered by Asset B, or
- Asset B powers Asset A

When giving your relationship upstream and downstream names, it’s not a bad idea to consider adding the name of the media as well (ie. **AC Powered By / Supplied Cooling Water By**) as a way of differentiating it from other relationships.

You’ll be given an example of your directional names in-use when creating a new relationship.

Create Our Initial Relationships

Using the relationship list, create the following new relationships...

Option	Value
NAME	Water
UPSTREAM	Sent Water From
DOWNSTREAM	Sends Water To
TYPE	Physical

Option	Value
NAME	AC Power
UPSTREAM	AC Powered By
DOWNSTREAM	AC Powers
TYPE	Physical

Option	Value
NAME	DC Power
UPSTREAM	DC Powered By
DOWNSTREAM	DC Powers
TYPE	Physical

Option	Value
NAME	Natural Gas
UPSTREAM	Supplied Gas From
DOWNSTREAM	Supples Gas To
TYPE	Physical

Option	Value
NAME	Steam
UPSTREAM	Supplies Steam To
DOWNSTREAM	Supples Steam To
TYPE	Physical

Option	Value
NAME	Hydraulic Oil
UPSTREAM	Sent Oil From
DOWNSTREAM	Sends Oil To
TYPE	Physical

Properties

What Are Properties?

Properties are single points of information about your asset.

If you were interested in storing the model number of your assets, you'd create a single property called 'Model Number' to store that information. When you wanted manufacturer, you'd create one called 'Manufacturer'.

Every property is available across every asset in your ARDI system – however, not all assets have to have a value for the property. A single property might only be relevant to one asset, or it might only be relevant to a particular *type* of asset.

Property Types

Each property has a distinct *type*.

The type of a property effects where it appears, how you edit it and determines if it should appear on the Dashboard and in clients such as ARDI AR.

Common types include...

Type	Usage
Plain Text	Used to store textual information, such as model or serial numbers.
Digital/Boolean	Values that only normally have two states, such as <i>on</i> and <i>off</i> . For example, 1=On, 0=Off.
Discrete/Enumeration	Values that are discrete – round numbers that usually represent the mode or state of a machine. For example, 0=Off, 1=Forwards, 2=Backwards, 3=Error
Lookup	Discrete/Enumeration types are only designed for small numbers of options. A <i>Lookup</i> is built to handle extremely large lookup tables of hundreds or even thousands of possible values.
Measurement	Analogue measurements For example, <i>temperature</i> or <i>pressure</i> values.

Identify the Properties

The best way of picking what properties you will need is taking a look at the I/O list from your system – this will usually give you an idea of what values are being recorded from site.

In this case we don't have an I/O list, but we could make some educated guesses based on the different sensors we have available.

We have distinct sensors for...

- Level (water vessel and hydraulic tank)
- Flow (gas)
- Temperature (water and steam), and
- Pressure (gas, steam and oil)

Identify the Sub-Types & Media

Here's where things are a little unusual in ARDI.

In our example we are measuring pressure. But we are measuring that value on *three completely different* kinds of content.

We have *gas* pressure which should be between 1 and 3kPa, *steam* pressure which should sit somewhere between 1 and 1000kPa, and *hydraulic* pressure, which in our system should be somewhere between 0 and 50,000kPa.

While it's possible to cover each of these with a single 'pressure' property, there would be a number of problems with this approach.

- When looking at 'pressure' across the site, your hydraulic pressures would completely drown out your steam and gas pressures, making them invisible.
- ARDI is given a min and max value for analogue measurements – which would be horribly out of scale for gas and steam.
- There may be significant problems with *property flow* (a topic that will be discussed later).

So in every case where you have significantly different ranges – for example, these three pressure levels – you should separate them into their own distinct property.

We normally suggest splitting the property across your different *media* (ie. the contents of your pipes).

To let ARDI understand that the measurements are related, following this format...

<measurement name> - <media name>

This gives us...

- Pressure – Water
- Pressure – Gas
- Pressure – Oil
- Temperature – Water
- Temperature – Steam
- Flow – Gas
- Level

You'll notice that we created just the one *level* property, even though it's being used on two different media (water and hydraulic oil).

This is for three key reasons...

- It uses the same scale (0-100%) on both tanks.
- Level is a property that would often be looked at across many assets at once – seeing oil and water level at the same time wouldn't be disruptive, and
- Level isn't a property that would usually 'flow' (for details, see *property flow* on page X)

Although there are potentially many more properties to measure – status, speed, vibration and power-usage of the hydraulic pump for example – these will be more than enough to get us started.

Creating Properties

The Property List

You access the Property List from **Administration | Properties**. It works very much like *The Relationship List* on page 9.

Click on the **Add New Property** button to begin creating a new ARDI property you can use on your assets.

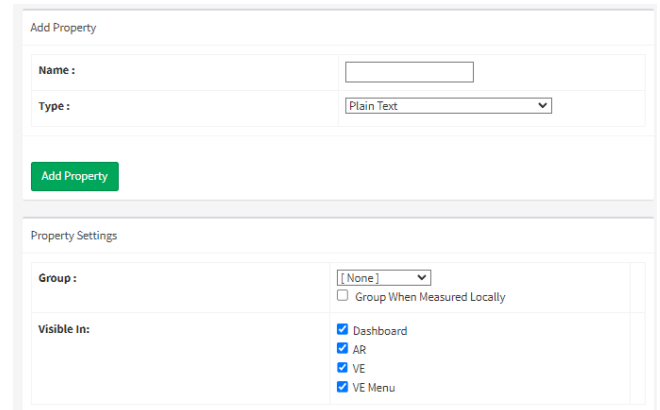
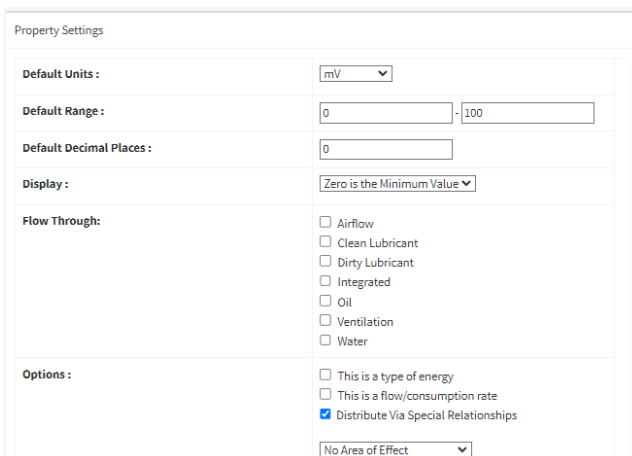
Initially, you're only given a few options for each of your properties. You can choose the...

Name: The human-readable name for your property.

Type: The type of property you'd like to create. The most common are *Plain Text*, *Measurement*, *Boolean* and *Enumerations*. See *Property Types* on page 12

Group: Some properties – often the ones that are less important or needed by fewer people – can be placed in groups. These properties will be hidden by default, but the user can click to expand the hidden properties at any time.

Visibility: Some properties might not be very relevant for some visualisations – daily totals and statistics are fantastic on a dashboard, but not very useful in 3D. These checkboxes allow you to remove properties from specific clients if you don't think they are going to be useful.

Type Options

Some property types come with additional options.

On the left you'll see a set of options that appear when you choose **An Analogue Value or Measurement** as the type of your property.

You can set the expected minimum and maximum values, the number of decimal places and units of measurement.

Options allow you to mark the property as a source of potentially hazardous energy, or a rate of flow and/or consumption.

And finally, you're given a list of all of the physical relationships in your ARDI system that this property can 'flow through'. For example, 'Temperature – Water' is a property that can flow through your 'Water Supply' relationship, since in most cases the asset immediately downstream from a temperature sensor is also getting water at the same temperature.

Consider the physical connections involved when checking any of these boxes. If your 'Water Supply' relationship reflects pipe, does the temperature and pressure measured at a given point normally pass on to the down-stream assets? If so, check the box.

Create Our Initial Properties

Option	Value
NAME	Pressure - Steam
TYPE	An Analogue Value or Measurement
UNITS	kPa
RANGE/POINTS	0-1100, 0 Decimal Points
FLows DOWN	Water

Remember that you can use the **duplicate** button to quickly copy the property you've already made – you can duplicate the first pressure to quickly create your other properties.

Option	Value
NAME	Pressure - Gas
TYPE	An Analogue Value or Measurement
UNITS	kPa
RANGE/POINTS	0-5, 0 Decimal Points
FLows DOWN	Natural Gas

Option	Value
NAME	Pressure - Oil
TYPE	An Analogue Value or Measurement
UNITS	MPa
RANGE/POINTS	0-50, 0 Decimal Points
FLows DOWN	Hydraulic Oil

Option	Value
NAME	Temperature - Steam
TYPE	An Analogue Value or Measurement
UNITS	Deg C
RANGE/POINTS	0-180, 1 Decimal Points
FLows DOWN	Steam

Option	Value
NAME	Temperature - Water
TYPE	An Analogue Value or Measurement
UNITS	Deg C
RANGE/POINTS	0-100, 1 Decimal Points
FLows DOWN	Water

Option	Value
NAME	Level
TYPE	An Analogue Value or Measurement
UNITS	%
RANGE/POINTS	0-100, 1 Decimal Points
FLOWS DOWN	
OPTIONS	

Option	Value
NAME	Flow - Gas
TYPE	An Analogue Value or Measurement
UNITS	Nm ³ /min
RANGE/POINTS	0-100, 1 Decimal Points
FLOWS DOWN	Natural Gas
OPTIONS	Flow Rate

Adding a new Unit of Measurement

You might have difficulty adding this last property, because you don't have a *unit of measurement* you can use.

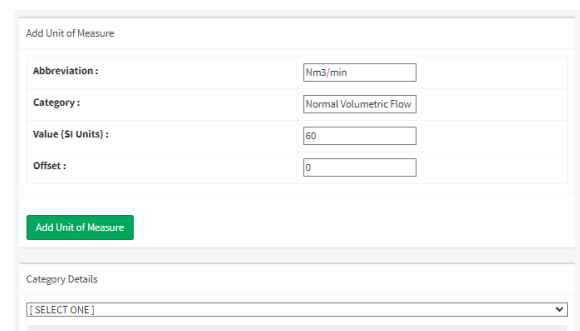
ARDI needs you to define each of your units, so that it can convert between them. You do this in **Administration | Units of Measurement**.

When creating a unit-of-measurement, you need to first choose a unique **abbreviation**. In this case, we want *normal cubic meters per hour*, or Nm³/min.

Then, you need a **Category**. Units of measurement fall into different categories, such as temperature, mass, speed etc.

A measurement can only be compared against other measurements of the same category – pressure only against pressure, for example.

You also need to pick a **value** and an **offset** for this value *compared to the base SI unit for the category*. The unit you pick as the base can be arbitrary, as long as its consistent. For example, we might decide that the base unit for Normal Volumetric Flow is 1Nm³/hr. In this case, our Nm³/min is worth 60 Nm³/hr. Since they share the same zero point, we don't need an offset (offset is primarily used for temperatures).



Types

What are Types in ARDI?

At their most basic, types are used to group similar assets together, making it easy to find assets that have common attributes.

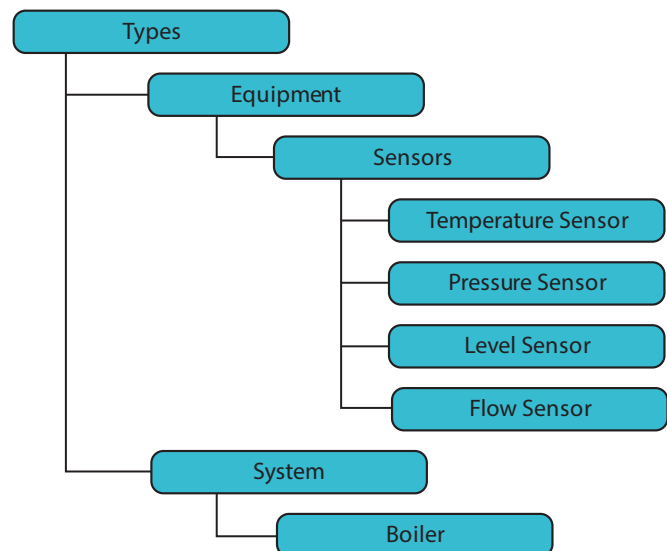
You can create a type called ‘Gas’ to group all of your natural gas-related equipment together, ‘Sensors’ to group your various types of measuring sensor, ‘Employee’ to mark staff members etc.

Types form their own unique hierarchy as well – you can have more specific types (ie. ‘Temperature Sensor’) under the more generic types (ie. ‘Sensor’).

On the right you’ll see an example of a very simple type hierarchy.

The first level simply contains **Equipment** and **System**. Equipment then contains **Sensors**, which in turn contains the different styles of sensor.

When you assign an asset to a type, it’s automatically assigned to all of the *parent* types – so an asset marked as a **Temperature Sensor** is also a **Sensor** and a piece of **Equipment**, too.



Equipment Types

At the very least, it’s well worth creating a list of Equipment types. This allows you to break up your assets into various categories based on what they **are**– for example, pumps, sensors or motors.

They allow you to group your various types of sensor together, and become even more powerful when you *add data to the types*.

Adding Data to Types

Types not only group your assets together, they also can be given properties and media, just like any asset. These are then available to *every asset that uses the type*. So if you add a ‘Oil Pressure’ property and a maintenance manual to the type called ‘Oil Pressure Sensor’, you’ll immediately have that property and manual available in every asset with the Oil Pressure Sensor type.

Looking at our earlier tree diagram, you’ll see where we have a ‘Equipment’ tag, with ‘Sensors’ underneath it (and several items under that, such as ‘Temperature Sensor’ and ‘Flow Sensor’).

We can add the property ‘Equipment Reference Number’ to the ‘Equipment’ type, and immediately it would appear on every single asset tagged with ‘Equipment’.

Not only that, but it would also appear in any asset tagged as ‘Sensor’, ‘Temperature Sensor’, ‘Pressure Sensor’ or ‘Flow Sensor’ – the types under Equipment. The parameters are inherited down the hierarchy.

Why use Types?

Types are almost overkill for this small application, but for larger applications they quickly become invaluable..

Single Point of Modification

If you update an instruction manual or specification sheet for a piece of equipment that you have in a number of locations across your plant, it can be a massively time-consuming task to go through each one, remove an old document and add the new one.

However, if your equipment assets are based on a type, all you need to do is update the document on the type itself. All of the assets that use it are updated at once.

Rapid Asset Creation

By creating a type for a commonly-used item (such as a temperature sensor or an isolation valve), you can create new assets of that type very easily. Rather than having to add an asset, find the properties that apply and add them one by one, you simply create the asset with the types you need and all of the properties, alerts and media drop into place instantly.

Easy Grouping & Discovery

You can use types to search for assets, both in the ARDI web interface and the *query language*. This way you can compare the performance of all of your pumps, wherever they are.

Self-Documenting

Types can help you document what assets actually *are*. This is discussed further in the following section.

Mixing Types

Because an asset can have more than one type, you can use them to describe not only what an asset **is**, but how it **operates**.

For example, a butterfly valve is a very common type of process valve found on plants around the world. They can be used either manually, as an on/off valve (they are always closed or always fully open), or as a control valve (they are carefully positioned between 0-100% open).

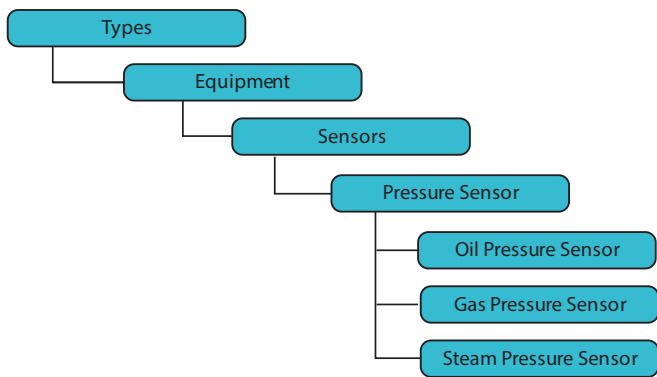
By mixing your types, you can have one type to describe the construction of the valve (ie, Gate or Butterfly Valve) and another to describe its function (Manual, Control or On-Off Valve). This can be very useful, as often control valves will have different properties than simple on-off valves.

Valves also have a 'failure state' – what happens to them when power is lost. You can also show this by adding an additional type. Thus when making a valve, you can say that it is a **Fail-Closed On-Off Butterfly Valve**.

You not only end up with a useful, human-readable description of the asset, but you can use these types to make sure that the correct data is available and that they appear in the appropriate dashboards and reports.

Splitting Types By Content

We usually split up our types the same way we did our properties. If we have three different types of pressure *measurement*, we usually have three different types of pressure sensor type.



Here we can see the different types of pressure sensor that match the different types of pressure measurement we created.

If you want to go further, you could even have types for the individual *brands* of sensor you have, to ensure the right documentation appears on each asset – but for most applications, that’s not required.

There are a number of other types of equipment you might want to add. You should consider making types for *everything*, just in case you add another one in the future.

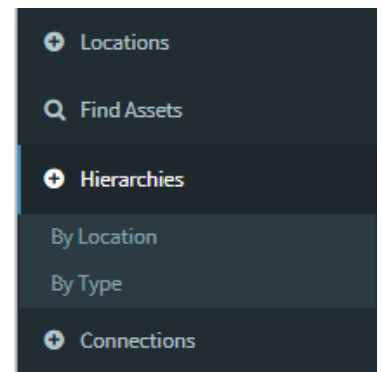
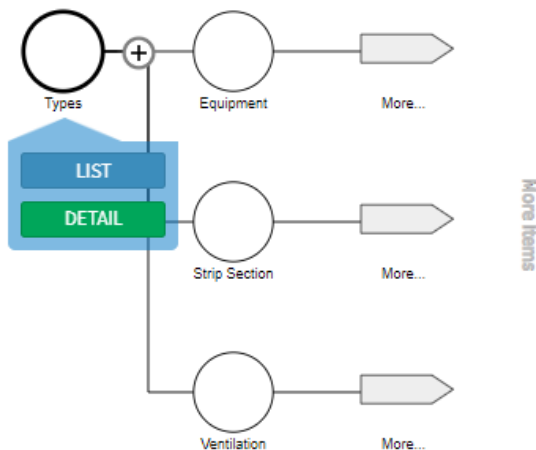
NOTE: You aren’t locked in to the properties that a type provides – you can add extra properties and even remove properties from the type on a per-asset basis.

Creating Types

The Type Hierarchy

You can find the type hierarchy in the ARDI sidebar.

Choose **Hierarchies** and then select **By Type**.

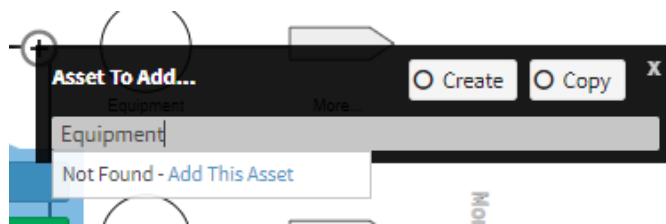


When your ARDI database is completely new, this is a *very* dull image. A single circle marked ‘Types’, with a plus sign.

It’s worth noting here that *types* are just another kind of *asset*, so what we will be doing here is exactly the same thing you will do later to add assets to ARDI.

To create a new type, press the plus-sign.

You'll be prompted to enter a name. Since we are starting the **Equipment** section, that's what you should type in.



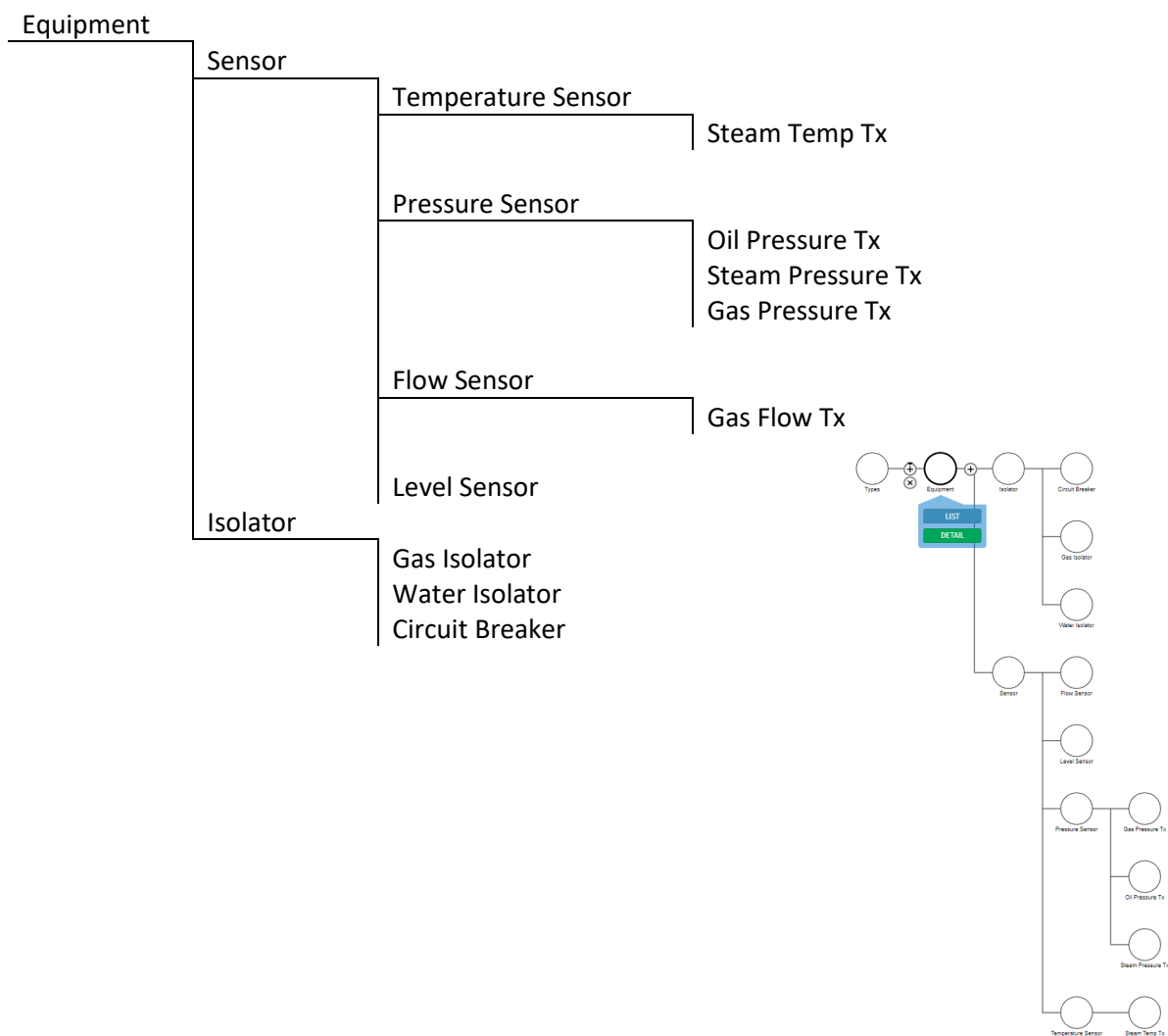
Since there isn't a type called Equipment yet, ARDI won't find it – click on *Add This Asset* to create your new type.

Once you hit the button, you'll be taken inside your new Equipment type.

From there, you can continue to press the plus button to fill in your type hierarchy.

Create Your Types

Continue by clicking on the parent and pressing '+' until you've created the type hierarchy shown below. You'll occasionally have to click back on the parent asset when creating several items under the one parent.



Adding Data to Types

Adding an ERN

Most pieces of physical equipment have an Equipment Reference Number, often also called a *tag number*.

We'd like to add a place-holder in each of our assets where we can fill that number in.

The easiest way to add the ERN to every piece of equipment is to add the property to the type.

Open the Details for the Equipment Type

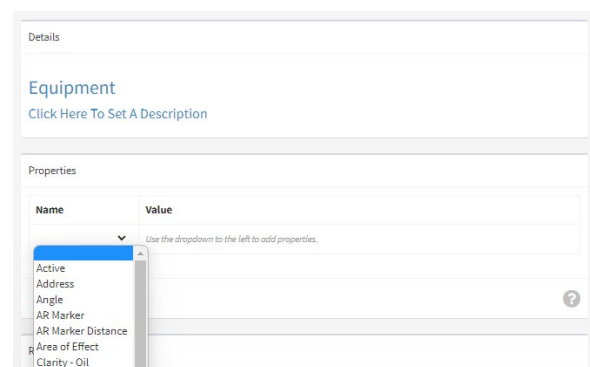
Go to **Hierarchies | Type** in the sidebar.

Click on **Equipment** and choose **Details**

You're now in the *details page* for the Equipment type.

To add a new property to a type or asset, find the drop-down list at the bottom of the *properties* section.

This will contain a list of all of the properties in your ARDI database, including those ones we made earlier.



Choose **Equipment Reference Number** from the list.

You'll be asked to enter a value. Since this is just a *type*, just leave it blank. This will add an empty ERN to each of your assets that use the Equipment type, or any of the other types under equipment - so it will appear for each of our types of transmitter.

Adding Properties to Transmitters

For each of your transmitter types, it's a good idea to add the property (or in some cases, properties) that you'll be reading.

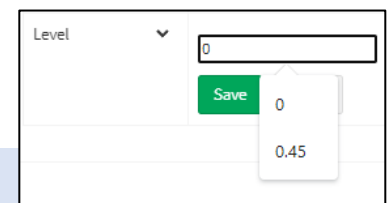
To set up our level sensor type, we'd go through the following steps...

Add the Level Property to the Level Type

Go to **Hierarchies | Type** in the sidebar, then navigate to **Equipment | Sensor | Level Sensor**

Press **Details**, open the droplist under Properties, and choose **Level**

When prompted, enter a value of 0 and hit **Save**



Substituting the appropriate property, repeat the steps above for each of the types we created.

When you edit these templates, you may notice that the initial property list isn't blank anymore. Since we are editing types that are under the Equipment type, they are inheriting the Equipment Reference Number property.

Is this a bit redundant?

For such a small application, making a full set of types will seem a bit of a waste of time. And you'd be right – until your application begins to grow.

Hierarchies

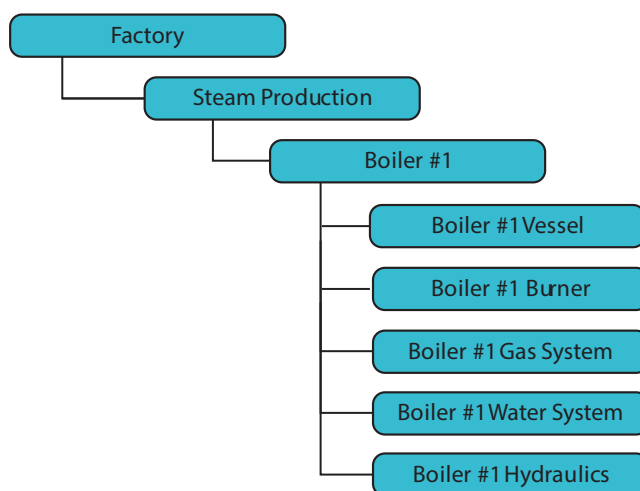
Building a Structure for your Assets

We've now figured out what our assets and properties will be, and created types for them.

Next, we need to figure out *where* they are going to be. We don't want to have a huge pool of assets without any structure – that would make it quite difficult to find anything.

Hierarchies put your assets into groups and sub-groups to make them much easier to navigate and understand.

Hierarchies form what are known as 'classification tree structures' – a tree that begins from a single point then branches out into sections and sub-sections, until you reach the assets at the end of the branches of the tree.



One of the most common hierarchies – and the most critical in ARDI – is the 'Location' hierarchy. It's the one included by default in your ARDI database, and one of the very few things in ARDI that can't be deleted.

The job of the Location hierarchy is to put each asset in a place that describes where it 'fits' inside your process.

This isn't just a **physical** location (you don't need to include that it's under the stairs or attached to the wall), but what *system* or *circuit* it might be part of. If you're using a

functional-location description for equipment on your site, you might already have a good idea of what your Location hierarchy might look like.

Where Should My Asset Go?

The Location hierarchy might be awkward for people when they first approach it. The key to the location hierarchy is *human readability*.

When deciding where to put assets in your Location hierarchy, the basic question is "How would I describe where this asset is in my process, to another person?"

If you wanted someone to isolate the gas to boiler 1, you might say "Go to the **factory** and turn off the **isolation valve** for the **gas system** on **boiler 1**."

This sentence has given you a lot of the key phrases you might need. If you're using words and phrases like *at, in, for the, inside* etc, these can be hints that you might want to include them in the Location hierarchy.

If you're using phrases like *under, on top of, connected to, fed by* etc, these are **not** good candidates for your Location hierarchy. They might be very valuable information and great candidates for other relationships. But they aren't ideal for Location.

Locations often fit a pattern similar to **Site -> Building -> System -> Subsystem(s) -> Containing Asset**, where you skip anything not relevant or redundant (ie. if you only have one building, you'd skip the building).

The Boiler #1 might be found in **Factory** (site) -> **Steam Production** (system) -> **Boiler #1**. In this case, we haven't included 'building' because the site only has a single building.

The level sensor inside the boiler vessel might be **Factory** (site) -> **Steam Production** (system) -> **Boiler #1** (Subsystem) -> **Vessel** (Encapsulating Asset) -> **Level Sensor**

Setting Up the Location Hierarchy

You open your Location hierarchy the same way we accessed our types – through the **Hierarchies** menu in the sidebar.

Choose **By Location** to open up the location hierarchy.

Our Location hierarchy is built up of *logical assets* – assets that often represent places, groups or concepts. There's usually one asset for the site as a whole, another for individual buildings, some representing systems and sub-systems etc.

Exactly the same way that we created our types (see *Creating Types* on page 19), by pressing the '+' button and typing in the name of the type we would like to create.

However, there's one difference to our workflow. Since we aren't creating types anymore, ARDI will pop up additional pages when creating assets.

This is where you can specify a few extra details about the asset you're creating.

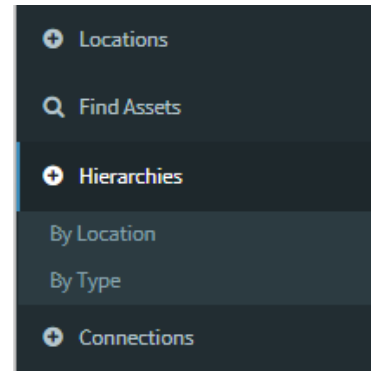
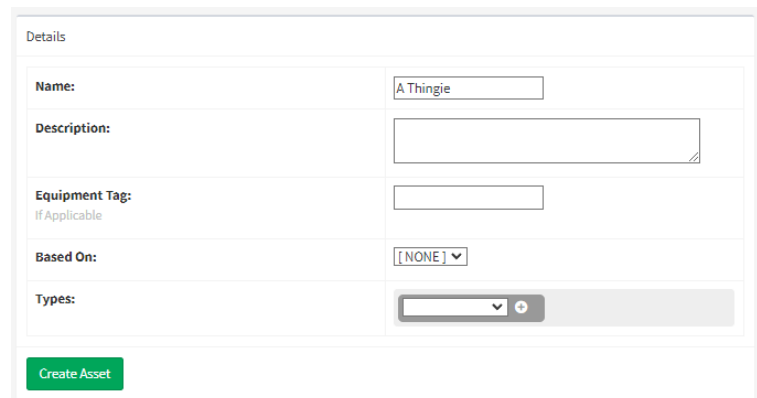
As well as just the name, you can also give your asset a **description**.

If your asset has an Equipment Reference Number, you can enter it in **Equipment Tag**.

You can also choose one or more **types** to assign to the asset.

When you press *Create Asset*, your asset will be created and placed in your hierarchy. When you make an asset *without* a type, you'll be taken straight back to the hierarchy to add more assets (an asset with no type is assumed to be a **logical** asset).

If you've chosen one or more types, it's assumed that your asset is a **physical** asset, and you'll instead be taken to the detail page so that you can fill in more data about the asset.

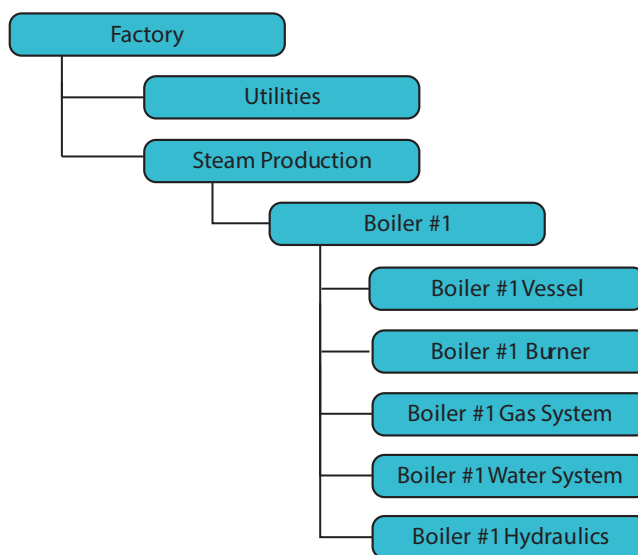
Create the Hierarchy

Using the '+' button, create the Location hierarchy shown on the right.

This will give structure to your ARDI system.

Once you've created the structure, we will **finally** be ready to start creating real assets.

You'll also notice we added an extra part of our hierarchy for *utilities* – this gives us a place to keep all of those assets that originate off-site.



Creating Assets

Choosing Where to Begin

Apart from your major mechanical components (which are usually featured as pieces of your Location hierarchy), it's best to build your assets in the order that they appear in your diagrams.

Choose one of your diagrams, start from the 'source' (assuming it has a source of energy or media) and follow the arrows until you've completed the full diagram.

Your best starting points are diagrams that are long (ie. have many assets) but are relatively straight and don't have many branches.

To start with, we are going to pick the Natural Gas Supply, simply because it is relatively straightforward.

Good Naming Practise

Note that while your asset names don't need to be unique, they should be informative – particularly when you get to making physical assets.

Calling something 'Temperature Sensor' is fine when you've only got one, but once you've got a few of them it's a better if the names of your physical assets contain enough information to figure out what its function is or what it's attached to.

So if your asset is part of a larger machine or system, we usually suggest adding that to the name. For example, "Boiler #1 Vessel Level Sensor" is much more informative than "Level Sensor". So we will be adding a little extra context to our asset names as we add them to ARDI.

The Gas Supply

Outside the Site

We start by opening our Location hierarchy and going to **Factory | Utilities**.

Hit '+' under there to create the *Natural Gas Supply*.

We don't need to give it any types.

Inside the Site

Now we can start creating the assets inside our site. Go to **Factory | Boiler #1 | Boiler #1 Gas System** in the Location hierarchy.

Press the '+' button and enter the name "Boiler #1 Main Gas Isolator". Press *Add This Asset*.

When prompted, enter the following details...

Option	Value
NAME	Boiler #1 Main Gas Isolator
ERN	BV1-G1
TYPE(S)	Gas Isolator

You might even find that the correct type has already been selected – ARDI automatically checks for any types that match with parts of your asset name.

When you hit **Create**, you'll then be taken to the detail page for that asset.

The Detail Page

Changing Names & Descriptions

You can change the name and description of the asset by clicking on either at the top of the page.

Type in the new name or description, and hit **save**.

Changing Properties

We've already explored the detail page when adding data to our types, but now we'll go into it in extra detail.

Name	Value
Equipment Tag:	STD-01
Power:	1,217.3 kW [0 - 3200, to 1 decimal place(s)]
Pressure - Force:	6.4MPa [0 - 10, to 1 decimal place(s)]
Reduction:	41.7% [0 - 100, to 1 decimal place(s)]
Use the dropdown to the left to add properties.	
Location:	This asset is part of Cold Rolling Mill. Position: 12.59, 11.99, 0 Scale: 7.67, 5.5, 37.73 Rotation: 0, 0, 0

You can add and set the value of your properties from the detail screen. You *add* properties by choosing the property name from the drop-down list.

You *change* property values by clicking on the value of the property in the list.

You *delete* properties by pressing on the trashcan icon to the right of the property value.

The pop-up property editor that appears when you change the value of a property might have a *see more* button. This allows you to adjust settings that are less common (such as the minimum or maximum range of an analogue value).

Adding Relationships

Although it's easier to add relationships from the diagram, there is a way to edit an assets relationship here.

You'll find a menu on the details page taking you to other information about the asset. These buttons include *Relationships*, *Alerts*, *Media*, *Flow Control*, *Data Sources* and more.

Choose *Relationships* to see a list of all the relationships the asset has. For the moment, there's just the one – the fact that your *Boiler #1 Main Gas Isolator* is part of the *Boiler #1 Gas System*

To add a new relationship, start typing the name of the other asset in the **Select an Asset** space. Once you've typed in the first few characters, it should pop up a list of matches. Simply click on the correct match.

Then choose which relationship you want to create, and which direction the relationship flows in, before hitting **Add New Relationship**.

Connect the Isolation Valve to the Gas Supply

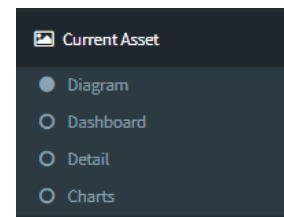
Start typing **Natural** and then choose **Natural Gas Supply** from the popup list.

Choose **Supplies Gas To This Asset** under **Natural Gas** and press **Add New Relationship**

Checking Our Diagrams

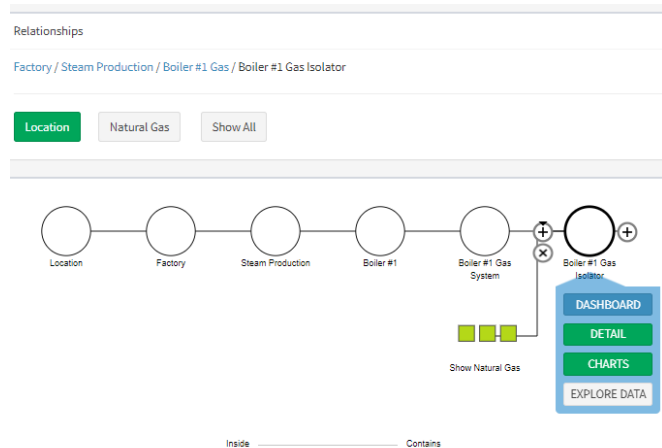
Before we continue, we will double-check our diagrams to make sure everything is the way it's supposed to be.

In the side-bar, under *Current Asset*, you'll find buttons that take you between the Dashboard, Detail and Diagram for your selected asset. Press **Diagram**.



On the right, you'll see an example of what the *Location* diagram for the *Boiler #1 Gas Isolator* should look like.

You can see that it's part of the *Boiler #1 Gas System*, which is in turn part of *Boiler #1*, which is in the *Steam Production* circuit of the *Factory*.



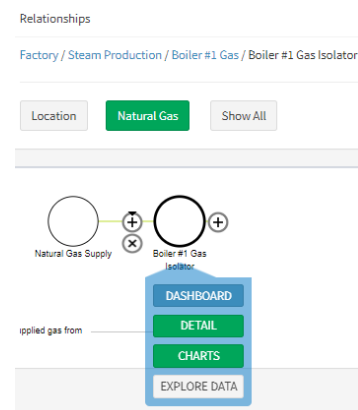
Above the diagram itself, you'll see three buttons. These are used to control which relationship you're looking at.

Change the relationship from **Location** (the current option) to **Natural Gas**.

This will give you a much smaller diagram, showing the flow of natural gas from the *Natural Gas Supply* to the *Boiler #1 Gas Isolator*.

Once we've created the rest of the gas system assets, we will come back here to fill in the rest of this relationship.

But for now, return to the **Location** relationship, so we can finish adding each of the assets to the *Boiler #1 Gas Supply*.



The Rest of the Gas System

Repeat the steps to create your assets for the gas system. These include...

Option	Value
NAME	Boiler #1 Gas Pressure Tx
ERN	PT1-A
TYPE(S)	Gas Pressure Tx

Option	Value
NAME	Boiler #1 Gas Flow Meter
ERN	FT1-A
TYPE(S)	Gas Flow Meter

Option	Value
NAME	Boiler #1 Secondary Gas Isolator
ERN	BV1-G2
TYPE(S)	Gas Isolator

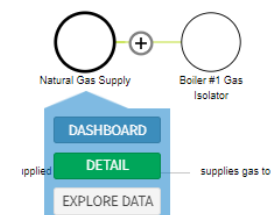
Option	Value
NAME	Boiler #1 Gas Control Valve
ERN	CV1-G1
TYPE(S)	

Filling in the Gas Relationship

Now we'll finish off the Natural Gas relationship. Head back to the **Natural Gas Supply** by searching or by clicking through your hierarchy.

Make sure you are looking at the *Natural Gas* relationship, not *Location*.

The diagram should look similar to the one on the right, with just the Natural Gas Supply and the Boiler #1 Gas Isolator.

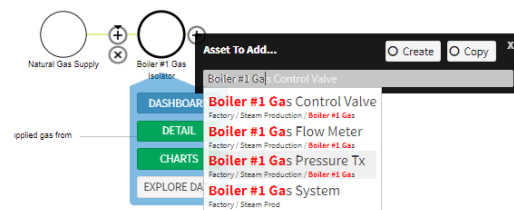


We are going to use our natural gas drawing as a guide and connect each of the assets together on the natural gas circuit.

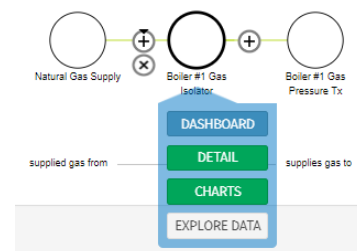
To do this, we click on Boiler #1 Gas Isolator and press the '+' button, just as if we were making a new asset.

Start typing in the name of the next asset in the natural gas line – the **Boiler #1 Gas Pressure Tx**. Once you get a few characters in, it should appear.

Click the asset in the list to add it as the next item in-line along the natural gas circuit.

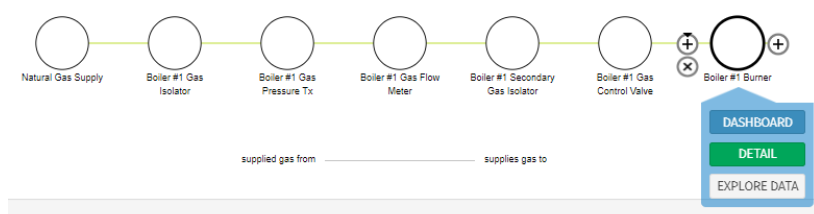


Note that the Boiler #1 Gas Isolator will still be selected, so you'll need to click on the pressure transmitter to continue the natural gas line further.



Finish the Relationship

Continue to click to the end and press '+' until you've created a complete line that runs from the **Natural Gas Supply** to the **Boiler #1 Burner**.



The Water System

Creating Water Assets

Next, we'll add our water assets the same way that we added our natural gas assets.

Find The Water System

Open the Location hierarchy and go to **Factory / Steam Production / Boiler #1 / Boiler #1 Water System**

We add each of the assets in the list, with two exceptions.

The **Boiler #1 Vessel Level Tx** and the **Boiler #1 Vessel Temp Tx** aren't added to the water system – instead, we place them *inside* the vessel. You'll usually put assets inside other assets when....

- They are mounted directly on or are physically inside another asset, and
- The asset you're putting *inside* is measuring something about the asset *outside*.

In this case, both the level and temperature transmitters are measuring the contents of the tank, so it makes sense that they should be considered **part of** the tank.

Product Transformation

Another thing to consider is the fact that your boiler vessel is where a *product transformation* takes place. Water and heat go *in* to your vessel, but what comes *out* is **steam**.

As such, you should clarify this in your asset names. The pressure and temperature readings from the tank will be the pressure and temperature of your *steam*.

We'll also change the relationship we use when we arrive at the tank. The **Water** relationship will go in, and the **Steam** relationship will go out.

Add The Assets

Go through and add each of the assets to the **Boiler #1 Water System** and the **Boiler #1 Vessel**.

When finished, you should have a set of assets with similar names to those below.

Boiler #1 Steam Check Valve

Boiler #1 Steam Drain

Boiler #1 Water Strainer

Boiler #1 Vessel Pressure Sensor

Boiler #1 Water Control Valve

Boiler #1 Water Filter

Boiler #1 Water Isolator

Boiler #1 Water Temp Tx

Water and Steam Relationships

Like before, it's quickest to go back to the source of a relationship to build it.

Find the Water Supply

Find the Location **Factory / Utilities** and add a new asset called **Water Supply**

Next, we want to make sure we are looking at the **Water** relationship. There isn't a button available for it, but we can make it appear by clicking *Show All* and choosing **Water** from the list.

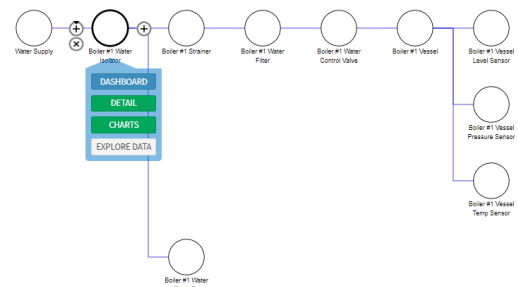
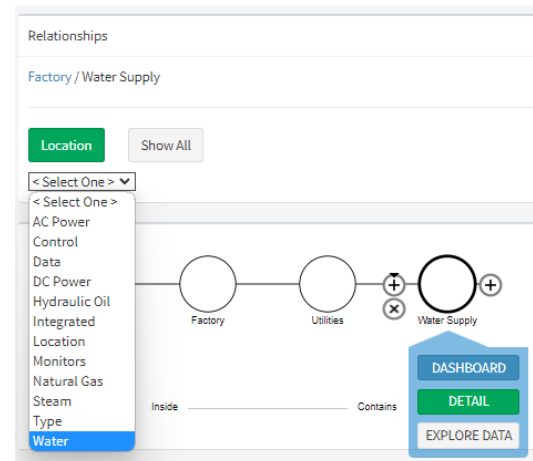
Once we've done that, we can use the '+' button to connect our assets together one-by-one in the order shown on the diagram.

Unlike our first relationship, the water relationship has *forks* – some objects that have more than one down-stream asset.

When you are done, it should look something like the image to the right.

It's important to note that you will only see the **Boiler #1 Water Temp Tx** if you have selected an asset *before it* in the water flow. The diagrams for equipment *after* the fork don't include assets that are down alternate paths.

Now that we've finished with our water, it's time for **steam**.

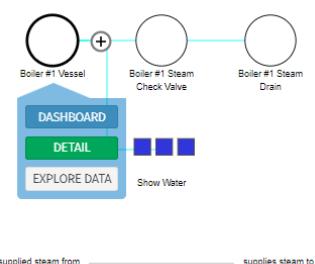


Switch to Steam

Select the **Boiler #1 Vessel**, choose **Show All** from the relationship list, and pick **Steam**

Once you've switched to the Steam relationship, you'll be able to add those extra down-stream assets.

You'll notice that the diagram includes indicators to show you where assets have multiple different relationships in-and-out. This helps with navigation, and to ensure people don't think you've reached a dead-end.



Property Flow

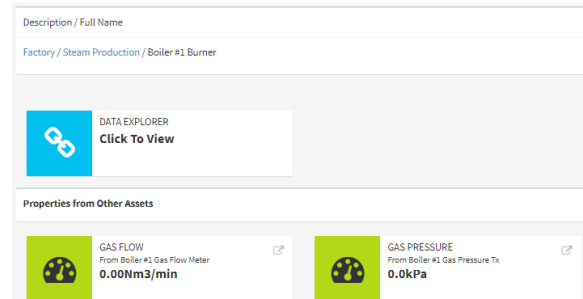
What is Property Flow?

ARDI relationships aren't just decorative. Properties can be made to flow between assets connected by specific relationships.

For example, try visiting the dashboard for **Boiler #1 Burner**.

You can see it on the right. Although the burner itself doesn't have any I/O, it still has two visible properties. The **pressure** and the **flow rate** of the natural gas that is being fed into it.

This is because we checked the 'Flows Down Natural Gas' property when creating it.



It's also one of the major reasons why we split our properties based on what type of medium they are. If you only had a single 'Pressure' property, it would be very confusing when you reached a hydraulically-powered valve. Which pressure is which?

By splitting them up into distinct Air and Hydraulic pressures, you can see both readings, and they won't interfere with one-another.

But while property flow is working well in our natural gas line, there are a couple of issues with it in our water circuit.

Water Property Issues

Water Temperature Not Flowing

If we look through our water assets – for example, the **Boiler #1 Water Control Valve** – we don't have any properties.

Although there's a temperature transmitter up-stream from the control valve, it's behind a fork in the line. Because of this, ARDI can't assume that the temperature measurement is relevant to the valve.

Level & Steam Data Not Flowing

Another issue arises from our vessel. It would be good to see the steam pressure and temperature when looking at the check valve – but because those measurements are being taken from instruments *inside* the vessel rather than in the vessel itself, they aren't flowing.

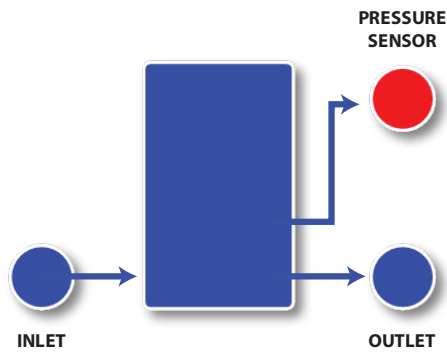
The Monitors Relationship

The solution to this is the **Monitors** relationship.

The Monitors relationship takes a little bit of explaining.

But basically, if asset **A** has a 'Monitors' relationship that points to asset **B**, it means that the measurement(s) being taken in asset **A** *belong to* asset **B**. By 'belong', we don't just mean that you can see them, but they are treated as if they apply to asset **B**, including being able to flow the property value further downstream.

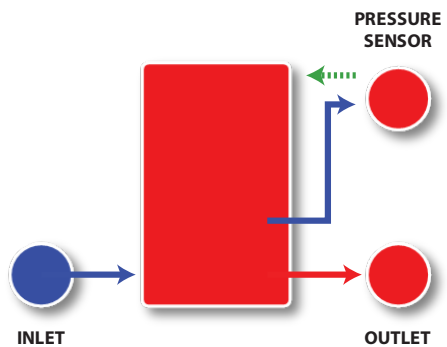
Let's focus on the Boiler #1 Vessel to demonstrate.



Here you have our existing vessel. We have water going in, steam going out, and a pressure sensor inside.

But the pressure isn't flowing into the outlet, because the pressure is being read at the sensor – the outlet isn't downstream from the sensor.

But what if we tell ARDI that the pressure sensor *monitors* the vessel?



When we create a Monitors relationship between the pressure sensor and the vessel, a selection of the properties (explained below) are copied across as if they belong on the vessel.

This doesn't just mean that the properties will appear when looking at the *vessel*, but the properties will also flow to all of the down-stream assets, delivering the steam pressure to our other assets.

Properties That Are Copied

Not every property is copied across as part of the Monitors relationship, since complex machines might have a variety of measurements, only some of which are relevant to the monitored device.

To be copied as part of the Monitors connection, the property either has to...

- 1) Be present on the monitored asset (ie. both the pressure sensor and the vessel should have the **Pressure – Steam** property), or
- 2) Flow through a relationship that the monitored asset has

Because of this, **Pressure – Steam** will automatically copy from the sensor to the vessel, because Pressure – Steam can flow down the Steam relationship.

But even if you had a Monitors relationship, **Level** won't automatically copy from the level transmitter. You'd need to add the Level property to the vessel to let ARDI know it's OK to copy the value.

Fixing the Issues

You can make the water temperature flow correctly by adding a monitors relationship between **Boiler #1 Water Temp Tx** and **Water Supply**.

You can make the steam properties flow by adding a monitors relationship between **Boiler #1 Vessel Pressure Tx**, **Boiler #1 Vessel Level Tx** and **Boiler #1 Vessel Temp Tx** and the vessel (ie. each of the named sensors should be directly connected to the vessel by a monitors relationship – they should not be connected to each-other.).

Finally, add the **Level** property to the vessel to make sure that the level transmitter reading is available on the vessel itself.

Small Relationships

Adding Electronics

You should hopefully be ready to add the electrical assets on your own now.

The only complication is that we didn't set up anywhere to keep them. We *could* simply create them directly under the boiler, but that might be messy.

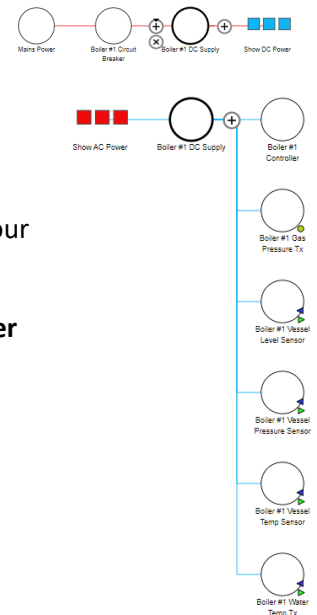
It's probably best to create a **Boiler #1 Electrical System** under Boiler #1 to keep our electronic parts in.

Once you've created that, you need to add the **Boiler #1 Circuit Breaker**, the **Boiler #1 DC Supply** and the **Boiler #1 Controller**.

We will also need a **Mains Power** under **Utilities** to represent the power-grid.

Finally, begin connecting your power. Like water turning to steam, your AC power *transitions* to DC power once it reaches the power supply.

- Create an AC power relationship from Mains Power, through the Circuit Breaker and into the Power Supply.
- Create a DC power relationship out of the Power Supply to the Controller. If you'd like to be accurate, you could also run DC power from the supply to all of your different sensors and the burner itself



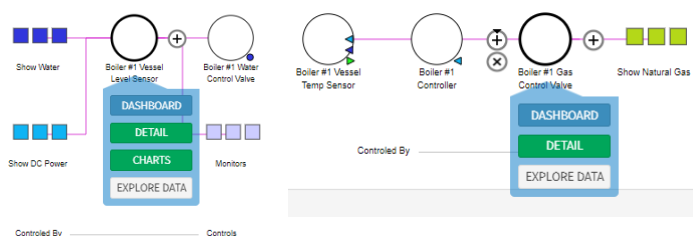
Taking and Making Control

Unlike power, gas and most of the other physical relationships, *logical* relationships often don't have a single source.

The **Control** relationship is used to highlight when assets directly influence each-others operation, such as how the temperature of the tank controls the valve to increase or decreases the flow of gas into the burner.

We have two main control loops in the system – temperature control (adjusting gas flow to control temperature) and level control (opening the water valve to fill the tank when it is getting low).

Logical relationships are added the same way as physical ones – go the start of the relationship, and work forwards.



Hydraulics

Avoiding Cyclic Relationships

The last diagram is the one for our hydraulic system.

The main difference between this relationship and the others we've seen is that this one is **cyclic**. The oil goes through the pump, into the tank, to the two valves, and then *back into the tank*.

ARDI is *capable* of handling cyclic relationships, but they should be avoided.

The main reason for this is that the entire concept of 'upstream' and 'downstream' becomes meaningless if your relationship is an endless loop.

Instead, we suggest creating a *second relationship* for the returning or re-circulating product. This has several advantages...

- It clearly explains (to the person reading the ARDI relationships) the *reason* for unexpected connections. When browsing your relationships, it can be confusing to see one asset with two 'Water' outputs – but it's much clearer when one is 'Water' and the other is 'Recirculated Water'.
- It ensures that ARDI can tell what is 'up' and 'down' stream from given assets
- It will in many cases prevent data from flowing incorrectly.

We suggest adding "Recirculating", "Return" or other description of why the cycle exists to the name of the second relationship. In this case, **Return Hydraulic Oil** would be a great name.

In most cases it should be relatively easy to find a place to transition between the two relationships. With a hydraulic system like this one, the transition to *Hydraulic Oil* happens when it enters the pump, and it turns into *Return Hydraulic Oil* after the oil has been used to operate the valves. Effectively, when the oil becomes **ready to use** and when it becomes **unusable**.

Normally, both relationships should share a similar colour with a different shade. For example, the return hydraulic oil can be a darker shade of the same colour you used for the normal hydraulic oil.

Create the Return Hydraulic Oil Relationship

Add it under **Administration | Relationships**.

You can feel free to build these assets as practise, but it's not strictly necessary. The only item we will need for later in the guide is the **Boiler #1 Hydraulic Pump**.

Bringing The Assets Together

Heating It Up

A sign of a good ARDI database is that it's *navigable*.

Although there are always exceptions, you should be able to use your relationships to follow your process, and – as long as assets are in some way connected – be able to get to almost any asset from any asset.

Right now, there's a bit of a gap in our flow. You can follow the natural gas system all the way to the burner, but then it simply stops.

But in real life, there *is* a connection between the burner and the vessel. There aren't wires or pipes involved, but there's certainly an exchange of energy – the burner *heats* the vessel. And we should capture that with a relationship.

Create the Heating Relationship & Use It

Create a new relationship called 'Heating' and use it to connect the **Boiler #1 Burner** to the **Boiler #1 Vessel**.

That change is simple, but it makes our system *much* more navigable. We can now start at the natural gas supply and work our way out through the steam outlet, or retrace our steps to find out where the water we are heating is coming from.

Linking to Data

Binding Properties

Now that we've set up our relationships, properties, types, structure and our assets, we're up to the final phase of making a functional ARDI system.

We need to connect our properties to external data.

Although the process isn't difficult, there are a couple of concepts that you need to understand before we begin.

Per-Property Connections

ARDI's properties are connected to external information on a property-by-property basis. This means that a single asset in ARDI might have ten properties, with 5 of them coming from the SCADA system, three coming from the maintenance system, one coming from an Excel spreadsheet, and the last being a fixed value.

Live and Historical Connections

A single ARDI property may be connected to a different source for *live* data than it is for *history*. In many cases your source of history data might be too slow (or resource-hungry) to constantly poll for the most recent information.

This way, you can source your live information from a quick-update source (such as OPC-UA, MQTT, a key-value store like REDIS or MEMCACHE etc.) while getting your history from slower services (SQL servers, historians etc.).

In some cases – such as when people use industrial historians like OSI PI – the one source is perfectly fine for both live data and history.

The process of connecting an ARDI property to live or historical data is called *binding*.

Drivers and Sources

To get information from an external source, you're going to need two things in your ARDI configuration. A **driver**, and a **source**.

A driver manages the communication for a particular protocol. For example, you might have an *OPC-UA driver* to read data from an industrial server, or a *Text-File Driver* to read information out of CSV files.

But the driver just looks after your connections. To actually connect to a *particular* source of data (ie. a specific OPC-UA server or a CSV file), you need to create a *source*.

Like everything else in ARDI, your data sources are assets too. However, these contain the connection details they need to pass the driver so you can read information.

If you haven't already, unzip the sample files to a folder on the ARDI server.

In there, you'll find a file called **BlueWidgetData.csv**, which contains some made-up data for our machines. We're going to bring that data into ARDI.

Installing Drivers & Addons

First, we will need to *install* the driver we need.

ARDI is modular – you only install those drivers and addons that you need (and are licensed for).

Although you can download and install drivers manually, most ARDI servers can download new modules through the **Extension Addon**. Note that this *requires an internet-connected ARDI server*.

For offline installations, you can install the training package manually to your ARDI server.

Instructions can be found at <https://ardi.com.au/training-package>

Enable the Extender Addon

Go to **Administration | Addons**

Find the checkbox next to **Extender**, then press **Save Addons**

Extender

You've now enabled the Extension Management addon, which allows you to download *other* addons from the ARDI web store.

Open the Extender Addon

Go to **Administration | ARDI Extensions** for a list of the available addons.

After a few moments, you will be shown a list of the available addons and drivers. Click on any of the options and press *Update/Install* to add them to your ARDI server.

You'll want to download three different components...

TextCycle Driver

Data Explorer

Site Builder Sample Data

Enable the Data Explorer

Return to **Administration | Addons** and check the box for the *Data Explorer*. Then hit **Save Addons**.

Adding a Driver

Go to **Administration | Drivers** to see the list of active drivers. Currently, there won't be any.

If you ever have issues with drivers, the **Status/Diagnostics** button can help you access diagnostic information about them.

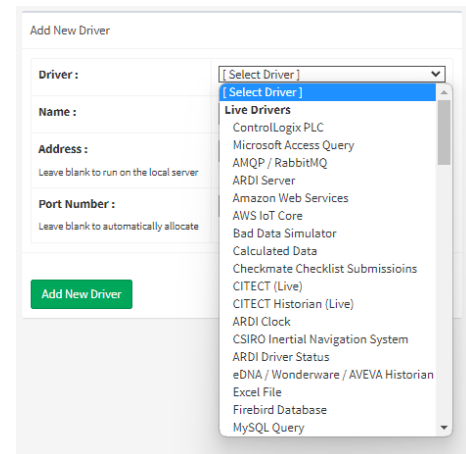
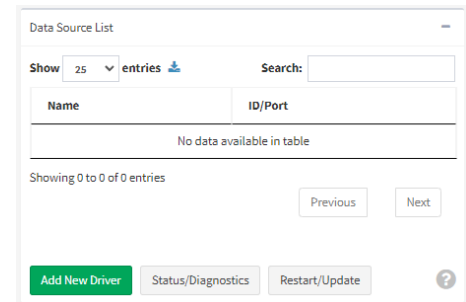
Press **Add New Driver** to start the driver setup.

For most applications, all you will need to do on this page is select a driver from the list at the top – all of the other fields will be automatically filled in or can be ignored.

Please do not put in the address for your data-source in the address column.

Putting the data-source address here is a common mistake, but it can cause significant problems. The address and port number shown here are advanced options for when you want to run ARDI drivers on machines *other than the ARDI server*.

Unless you've been advised by your Partner or understand exactly what you're doing, we advise leaving the *Address* and *Port Number* options blank.



Create a TextCycle Live & Historical Driver

Choose **Text Cycle** from the **Live Drivers** section, then press **Add New Driver**

Go back into the Add New Driver page and do it again, this time from the **Historical Drivers**.

It may take a few moments to create the new drivers – in the background, ARDI is creating new services that start up with your PC to acquire data.

The Text Cycle driver takes a simple tab-delimited or CSV text file and makes it repeat infinitely. It makes it easy to put together a quick demonstration system that has meaningful data in it that you can use to test your reporting and analytics.

Adding a Source

Now that we've added the driver, the second part is the **source**.

Data sources are assets, just like anything else on site. So we will need to create a new asset to represent our source of information.

Let's call it **Database**, and we'll place it under the Factory in the Location hierarchy.

Open the Data Source Details

Open the details for your **Database** asset. At the bottom of the page, press **Data Source(s)**.

You can use this page to add Live, Historical and Event data sources.

We want our Database to give us both live and historical data, so we will create one of each.

Press the **Create** button next to *Actual – Live*

In the next page, choose *Cycling Text File Data* from the **Driver** list.

The Driver Settings section will now update with options that are specific to your chosen type of driver.

In this case, we want to press the **Use Training Data** button, and leave the **Delimited** option set to *Tab Delimited*.

Then press **Save Data Source Settings**.

Do It Again

Go through the same process, but this time for **Actual – History** instead of **Actual – Live**.

The last thing we might want to do is increase the update-rate for our live data source.

By default, ARDI scans the live data source quite slowly (just in case there are performance issues with the data source).

Pressing **Timing / Rate** on the live source will allow you to adjust the update rate for live data. The cycling text driver is very quick, so we suggest using 1 second.

You've now set up both the drivers and the data source that you need. It's time to start binding properties to our data.

Actual - Live	Text File: Path To The File With 1 second updates	Reconfigure	Timing / Rate
Actual - Historical	Text File @	Reconfigure	Advanced

Binding Live Data

The sample data file contains several channels of data with the following names...

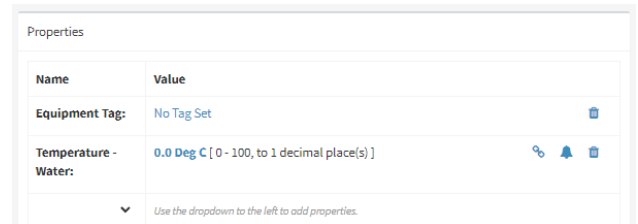
B1_WATER_TEMP	B1_OIL_LEVEL	B1_BURNER_RUN
B1_WATER_LEVEL	B1_PUMP_ON	B1_GAS_FLOW
B1_STEAM_TEMP	B1_PUMP_CURRENT	B1_GAS_VALVE_PV
B1_STEAM_PRESSURE	B1_PUMP_VOLTAGE	B1_GAS_PRESSURE
B1_WATER_VALVE		

To bind them to a property, we choose one of these data channels and find the corresponding asset.

For example, let's start with the incoming water temperature, which is the first item in our list (B1_WATER_TEMP).

To bind it to data, we...

- Search for the asset (**Boiler #1 Water Temp Tx**)
- Open the details for the asset
- Find the property we want to bind and press the 'link' button (the picture of two chain-links).

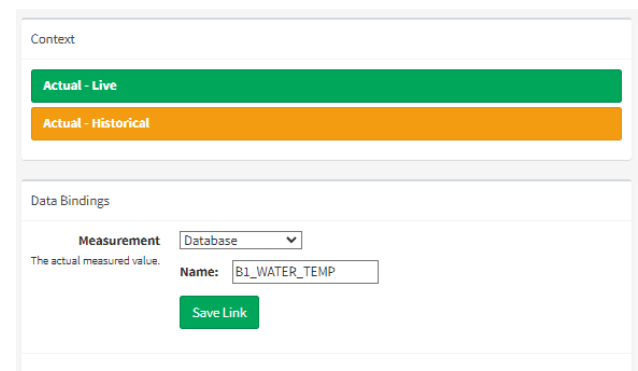


The page that opens allows you to choose if you're connecting the point to *live* or *historical* data.

Leave *live* selected, and in the drop-list next to **Measurement**, choose *Database* (the name of the source of our information).

You'll then be asked for the name of the point you want to bind to. Enter **B1_WATER_TEMP**

And finally, press **Save Link**.



Binding Historical Data

You follow the same process for binding your historical data, but in our case we can use a shortcut.

When you have an data source that has both a live *and* a historical driver of the same type, a button will appear saying **Copy Live Data Binding**.

This allows you to simply copy the live binding over to the historical one, saving us from having to type anything in again.

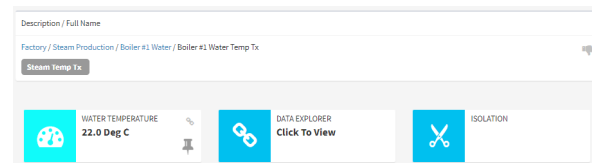
Check Data is Flowing

ARDI Dashboard

Let's check that our data is flowing through to ARDI.

Go to the **Dashboard** for Boiler #1 Water Temp Tx.

You should see that the *Water Temperature* property now has a value – usually around 22 degrees C.



If you continue to watch, you might notice some subtle fluctuations in the value – the number on this page updates live.

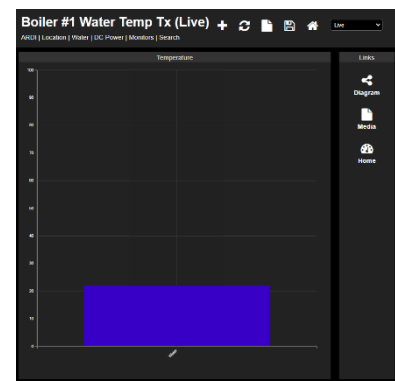
Data Explorer

Next, press the **Data Explorer – Click To View** button to open the Data Explorer.

This will open a vertical bar-chart of all of the assets properties. Again, you can see the steady 22 degrees.

This lets us know that our live information is working, but now we need to double-check our history.

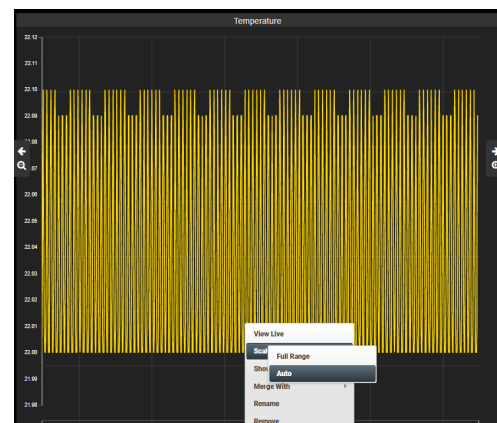
In the top-right corner, open the droplist where it says *Live* and change the value to *Last 30 Minutes*



Initially, the chart you'll be shown is very flat – this is because the chart is automatically scaled to the ARDI range for the property.

You can right-click in the empty space on the chart to open a menu, then choose **Scale | Auto** to zoom in and see the cycling temperature variations we are getting.

Press the 'Back' button in your browser twice to get back to the normal ARDI dashboard for your asset.



Bulk Importing your Data Bindings

The manual method for setting up your data bindings is very slow.

What we usually suggest is that you manually bind **one** point, then you use exporting and importing to do the rest.

Follow the steps below to bind the other points.

- Go to **Administration | Export Data Bindings** to download a CSV file containing all your bindings.
- Open the file in your preferred spreadsheet program, such as Microsoft Excel.
- Copy/Fill-Down the lines so you have enough to bind each of the assets you've created.
- Update the **Name** and the **Actual Live Address** for each point.
- If you're using the same data source for both points, change the **Actual History Address** for line 2 to be '=G2', so it copies the value from the live point, and fill that down to each row.
- Save the file in a place you'll remember.
- In ARDI, go to **Administration | Import Data Bindings** and drag in the CSV file you saved in the step above.
- Hit 'Import' – ARDI will check the file for validity, and if everything is fine you'll be able to push 'Import' again to complete the process.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	ERN	Name	Type	Property/Name	Node/ID	Actual Live Source	Actual Live Address	Actual Live Transform	Actual History	Actual History Address	Actual History Transform		
2		Boiler #1 Water Temp Tx	Property	Temperature - Water	measurement	Database	B1_WATER_TEMP		Database	B1_WATER_TEMP			
3													
4													
5													
6													

Import Issues

Reviewing Import Files...

ARDI is checking your import files for issues - please wait a few moments for the inspection to finish.

All tests passed successfully.

Press the 'Import' button below to begin the import of your spreadsheet file(s).

Digital Properties

Identifying our Digitals

The sample data contains three digital properties – they are always simply 1 or 0.

CHANNEL	MEANING	VALUE
B1_BURNER_RUN	If a flame is detected in the burner.	0 = No Flame 1 = Flame
B1_WATER_VALVE	If the water valve is open	0 = Closed 1 = Open
B1_DRAIN	If the steam drain is open	1 = Closed 0 = Open

We can add these points to ARDI the same way we added our other properties earlier, but this time we need to make sure that we use the **Digital/Boolean** type.

For these properties, we don't need to pick a maximum, minimum or unit of measurement – we just need to define human-readable text to translate the number into a state.

We would need two properties – one named **Flame Detected** and one named **Valve State**.

Dealing With Scale Problems

In this case, we've got a small issue with the meaning of the digital signal.

The water valve uses a value of 0 to indicate 'closed', and a value of 1 to indicate 'open'.

The drain valve does the *opposite*.

Note that we always suggest that problems with data – such as scaling issues, incorrect units etc. are always dealt with at the source, rather than in ARDI.

However, this specific issue is often not a problem. The '1' state of a digital often indicates that the value is **not safe**, or **not the resting position**. As such, the meaning can change between pieces of equipment.

There are two possible ways to deal with this.

- Create two different 'open' properties – one for each different I/O pattern.
- Transform the value before displaying it.

Having two different properties to represent open/closed can be a little confusing to the user, and makes it difficult to look at valve status across the entire plant.

The second option is the most attractive.

In ARDI, a **transform** changes an value that is being read from a data source. These can be used to perform a number of adjustments, such as re-scaling or sanity-checking.

Adding a Transform to a Binding

First, we will need to create our properties and data bindings before we can create a transform.

Create the **Valve State** property in Administration | Properties, and let's have '1' meaning Open and '0' meaning Closed.

Next, add this property to both valves – the **Boiler #1 Steam Drain** and the **Boiler #1 Water Valve**.

Bind the new property on the water valve. Because the water valve uses the same values that our type does, we don't need to do anything special for it. Remember to also copy the live binding to history.

But when binding the Steam Drain, we'll need to do things slightly different.

- Setup the *live* binding of the property.
- of the properties to the channels in our sample data – the Steam Drain is linked to **B1_DRAIN** and the Water Valve to **B1_WATER_VALVE**

Before we copy the live data bindings over to the historical, we want to set up a transform. So press the **Edit Transform** button under the binding information.

The Transform Editor

There are several different options when transforming your data. Each is enabled by clicking on the '+' button next to their name to expand their options.

Note that these transformations occur in the order they appear in the list. So if you were to perform a Low Cut, a Scale and Rounding, they would occur in exactly that order – because that is the order they appear in the list.

If you want to remove a step in your transform, simply collapse it by pressing the '+' button again. There is no need to erase any information – just closing it and pressing 'Save' is enough.

In our case, we can use the **scale** transform to convert our number.

The scale transform needs you to specify the *input* min & max, and the *output* min and max. To invert a digital number, set the **input** min to 0 and max to 1, and the **output** min to 1 and max to 0, then hit **Save Changes And Return**.

Enumeration Properties

Identifying our Enumerations

The hydraulic pump is a complex machine that we haven't really focused on yet. Our sample data has several channels of information that might be of use – **B1_PUMP_VOLTAGE** and **B1_PUMP_STATUS** are both analogue values, while **B1_PUMP_STATUS** is an *integer*.

Unlike the plain 0/1 of a digital, the pump has four different possible statuses, which are...

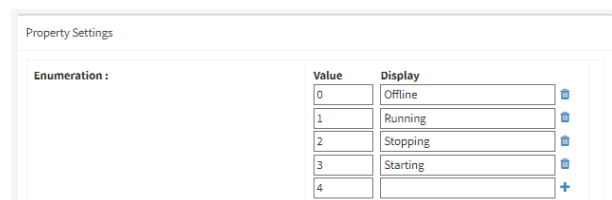
VALUE	MEANING
0	Offline
1	Running
2	Stopping
3	Starting

To make this work we can create a property called **Pump Status**, and its type will be a **Discrete Value/Enumeration**.

Entering Possible Values

You define what each numeric value means when creating an enumeration.

You can press the trash-can icon to get rid of an option, or press the '+' button to add one.



Value	Display
0	Offline
1	Running
2	Stopping
3	Starting
4	

NOTE: The line with the '+' button will not be saved. If you need four different options, there should be *five* lines, the last of which is empty.

Adding Symbols

What Are Symbols

Although the ARDI diagrams are good, they don't provide a lot of insight into what your assets *are*.

Using *symbols*, you can replace those boring circles in your ARDI diagrams with images that show people what each asset represents, without people having to read or click.

ARDI provides a **Symbol Library** where you can download individual symbols, but to save time we've included a set of them in a pre-made ARDI Symbol File.

Importing Symbols Into ARDI

Go to **Administration | Symbols** and press **Add New Symbol**.

Choose your .ardisym file, and press **Add Symbol(s)**.

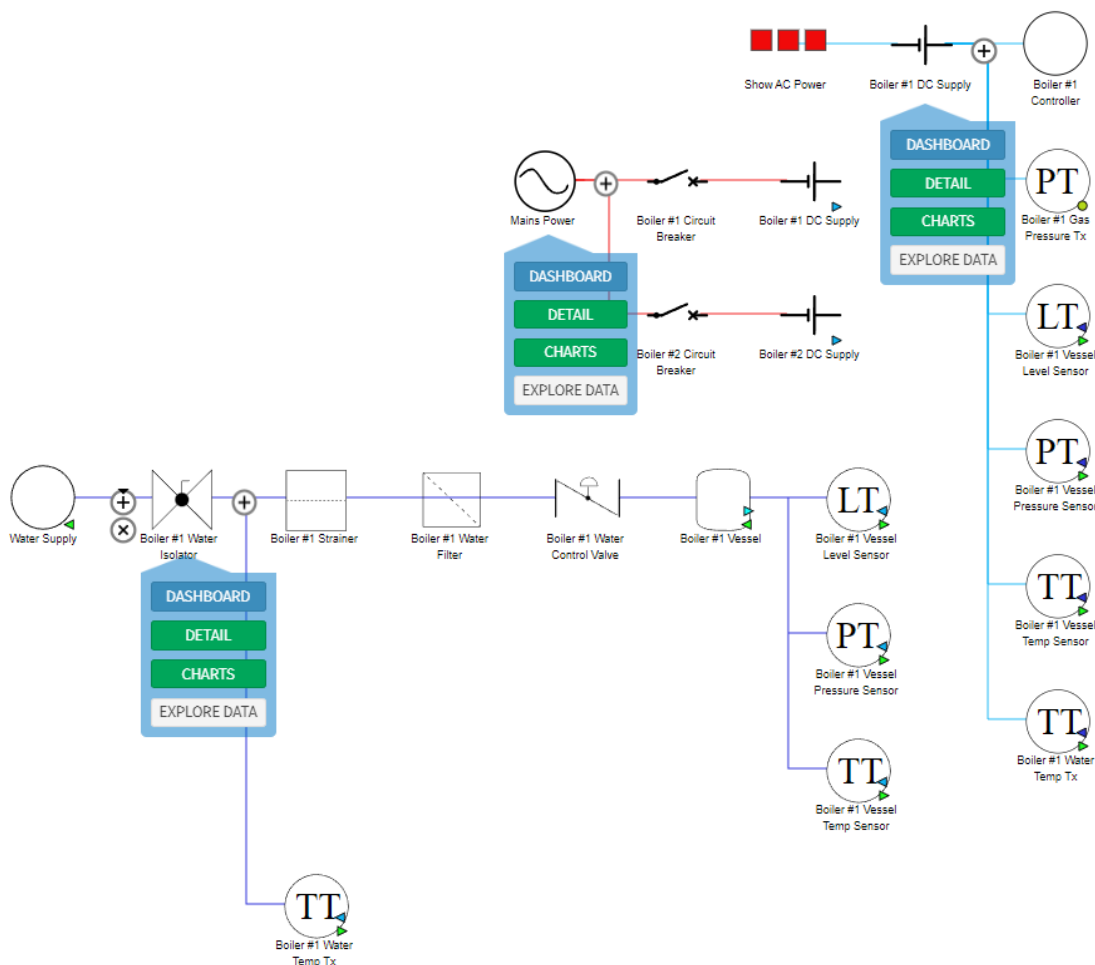
This will give you a set of symbols that are useful for the system we've just designed.

Assigning Symbols to Assets

Your symbols are just another kind of property that an asset might have.

To assign a symbol, you simply add the **Symbol** property to an asset, or even better to a *type*.

Symbols are organised into a small (three-tier) hierarchy, so to find the symbol for a *circuit breaker*, you'd look in *Electrical | Switches*.



Colouring Your Properties

It's often helpful to give your properties a colour code so you can very quickly communicate if values are expected, or a problem.

This is a purely optional step in your ARDI configuration, but it's very common to set up colours for your digital and enumeration properties to control how they appear in visualisations.

Flow - Gas			
Level			
Pressure - Gas			
Pressure - Oil			
Pressure - Steam			

In **Administration | Properties**, you might notice that some of your properties include a small camera icon. This indicates that the property has *display options*.

Click on the camera to customise the colours for each of your different states, or set a gradient for your analogue values.

Value	Colour	Icon/Image
Stopped		
Warning		
Stopping		
Starting		

Display Settings

Range Type: Fixed

Default Range: -

Colour Scheme:

When Values Cross Zero:
 When Negative
 When Zero
 When Positive

Minimum Span:

Other:
 Fade Items Without This Property
 Invert Scale When Negative

Save Settings Remove Gradient

The Second Boiler

The Sequence Relationship

It's time to make the second boiler. But first, we will introduce the *sequence* relationship.

If you ever have pieces of equipment that are very similar and side-by-side, feed one-another, load share or are hot-spares for one-another, they'll probably have a particular order.

In this case, we have Boiler #1 and Boiler #2, both of which supply steam.

Before we make the second boiler, we should create a 'sequence' relationship we can use to connect the two.

This is a *logical* relationship. A sequence relationship helps you control the order your assets will appear in when creating reports, dashboards and other visualisations, as you might often find that alphabetical sorting isn't particularly useful.

Create the Sequence Relationship

Make the logical **Sequence** relationship in **Administration | Relationships**.

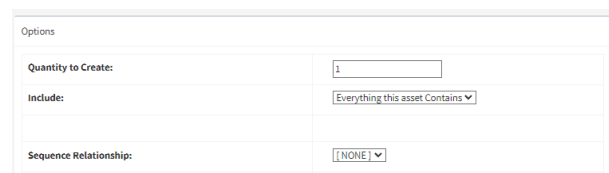
How to Duplicate a System

So we've now got to do everything again?

Luckily, no. ARDI has the ability to duplicate assets. This duplicates not only the asset itself, but *everything inside that asset* in the Location hierarchy.

Follow the steps below...

- Open the details page for **Boiler #1**
- Open **Asset Tools**
- Press **Duplicate This Asset**
- Choose the number of new assets we want (1)
- Choose the *Sequence* relationship under **Sequence Relationship**
- Press **Duplicate Asset**



Options	
Quantity to Create:	<input type="text" value="1"/>
Include:	<input type="text" value="Everything this asset Contains"/>
Sequence Relationship:	<input type="text" value="[NONE]"/>

This duplicates your entire boiler – all major components including all of the gas, electrical and hydraulic assets we have set up so far.

It also duplicates all of your relationships.

In the case of an *internal* relationship – one that begins and ends with assets that are part of the asset being duplicated – the relationship is copied exactly. This means that in your new duplicate, your new hydraulic pump will be connected to your new hydraulic tank and won't have anything to do with the old one.

With an *external* relationship – where one end is inside the asset and the other is outside – you will end up with all of your duplicated assets connected to the external one. So when we duplicate our boiler, the Natural Gas Supply will be connected to each boiler separately.

What Gets Changed

A few changes may occur to your asset during duplication.

Most importantly, if you have any assets with a '#' in their name followed by a number, the number will be incremented.

This is why we called our first boiler 'Boiler #1'. During duplication, ARDI looks for the number and increases it on any duplicates. This means our new boiler will be created with the name 'Boiler #2'. We won't even have to go in and rename it.

As an advanced feature, ARDI can also attempt *automatic binding* on your properties. However, this is only limited to sites that have very consistent channel naming and requires some configuration that is beyond the scope of this tutorial.

For that reason, we'll probably have to do the export-and-import trick to bind each of these new asset properties to data.

Making Bulk Changes

You can use your relationships to perform large-scale changes to your ARDI database.

Let's say that we need to rename our assets. Instead of 'Boiler', they'd like to call it 'Steam Generator', and instead of '#1' they'd prefer 'A' and 'B'.

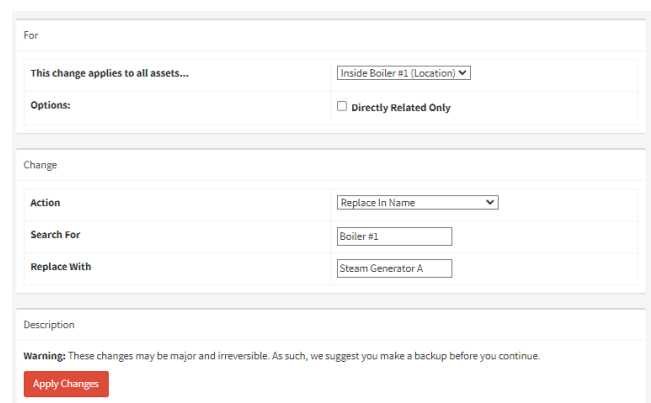
Obviously we don't want to go through the assets manually. This is where the Bulk Changes tool is useful.

Because of the risk of doing significant damage to your ARDI configuration, we suggest backing up (using **Administration | Backup**) before using the Bulk Changes system.

While you don't need to follow these steps right now, it's good to know the function exists.

The steps are...

- Find a common asset (such as a type or place in the Location hierarchy) that all of the target assets share. For instance, **Boiler #1**
- Go to the details of that asset, click **Asset Tools** and **Change Assets Under/After This One**.
- This change will modify everything downstream on a particular relationship. It's usually Location, but you can change this if needed.
- Pick an **Action**, such as *Replace In Name*
- For this action, you can fill in a 'Search For' and a 'Replace With'.
- Press **Apply Changes** to perform the update.



The screenshot shows the 'Change Assets Under/After This One' interface. It has a 'For' section with a dropdown menu set to 'Inside Boiler #1 (Location)'. Below that, there's an 'Options' section with a checkbox for 'Directly Related Only'. The 'Change' section includes an 'Action' dropdown set to 'Replace In Name', a 'Search For' input field containing 'Boiler #1', and a 'Replace With' input field containing 'Steam Generator A'. At the bottom, there's a 'Description' section with a warning message: 'Warning: These changes may be major and irreversible. As such, we suggest you make a backup before you continue.' and a red 'Apply Changes' button.

Enabling Augmented Reality

Download and Enable the Addon

You can download the AR addon the same way you installed your drivers and the Data Explorer addon. Just remember to enable it in **Administration | Addons**.

Setup Markers

ARDI-AR uses augmented reality markers to identify your equipment. To let ARDI know which asset belongs to which marker, you need give your assets the **AR Marker** property and fill in the ID number of the marker.

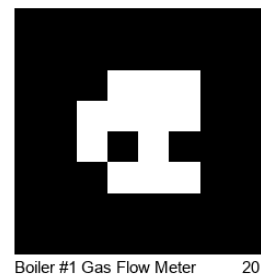
Go to the details page for an asset you'd like to attach a marker to (in this case, the gas flow meter), drop-down the property list, choose **AR Marker** and enter a number – in the example, we've picked marker 20.

Test the Marker

As long as you have a camera, you can test the marker out right now – even if you're on PC.

First, go to the dashboard for your asset, and click the new **Test AR** button.

You'll be shown an image of your marker. You can also download markers, either plain (with no text) or annotated with the name of the asset and ID number.



Print the marker or open it on a second device (such as your phone).

Then, go to **AR** in the sidebar. You'll start Augmented Reality. Simply lift up the marker so the camera can see it – the marker should quickly be identified. You can then touch the name of the asset to access your ARDI information.

Web-based Augmented Reality requires a security certificate to function. Your ARDI server includes one by default, but it is *self-signed*, meaning you have to install the other half of the certificate on your device.

Working in Tight Spaces

It is possible to add more than one asset to a single AR marker. This is helpful when assets are closely backed together or behind locked doors.

When you have multiple assets using the same marker, we suggest adding the **AR Marker Distance** property. This tells ARDI where the asset is in relation to the marker. It allows ARDI-AR to overlay pointers on top of each of the assets of interest.

Making It 3D

Importing the 3D Model

We've provided a very simple 3D model of the vessel and the water system, just to show you the basics of getting ARDI information into ARDI-VE.

There are two main ways to build up your site in 3D. You can build it **piece by piece**, where each asset is placed individually, or **all-in-one**, where your model has many different parts that are already connected together.

All-in-one construction allows you to work with your existing 3D models (if any), and will usually be more accurate and faster to set up. However, they do require someone to create custom 3D models for you.

Piece-by-piece construction is more time-consuming – you've got to add a number of different objects into ARDI. But there are advantages - you can use our 3D library of objects, so that you don't actually need to create 3D models. Your system is also much easier to change.

In this example, we have an **all-in-one** model containing several pieces of a boiler.

Creating an Environment

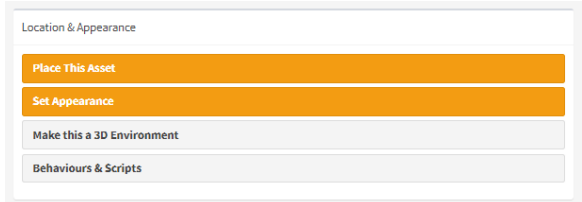
Before you can do anything, you need to create an *environment*. This creates the 'world' your assets appear in. You can use satellite photography, drawings or maps as the base of the environment, or you can simply create an empty world.

We are going to do that last one – create an empty world we can place our assets into.

Go to the **Factory**, open the details page and find **Placement & 3D** in the menu.

Then press **Make This a 3D Environment**.

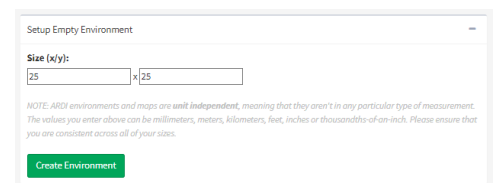
You'll be prompted to choose how you'd like to create the 3D environment. You can upload images, SVG and DXF files to create individual *layers* of your space, upload a pre-made environment from the ARDI Environment Creator tool, or simply build an empty space.



Choose **Create An Empty Environment**.

The last step is choosing the *size* of your environment. ARDI doesn't care which units you use, as long as you're consistent with them. This number could be in meters, feet or fathoms.

Leave the default size of 25x25, and press **Create Environment**.



Uploading a 3D Model

Next, we need to give ARDI a 3D model of our boiler.

Go to **Administration | 3D Models**

Press **Add New Model**

Drag the *Site Builder Model* that is part of the tutorial files into the form and then press **Add Model**.

Setting Appearance and Position

Position

Our model contains a number of parts – the vessel and pieces of the water system. The 3D model needs to be attached to an asset, and when doing all-in-one models, it's best to assign the model to the object furthest up the Location hierarchy.

In this case, we should assign the model to **Boiler #1**

Go into the details for Boiler #1, find **Placement & 3D** again, and press **Place This Asset**.

For now, we will just place this object in the middle of the scene. Set the position to **12.5, 12.5, 0**.

Leave the rotation at **0,0,0**

Set the scale to **1,2,2.2**. Normally you won't have such a good idea of the size of the model and will need to adjust it later in ARDI-VE, but we've measured it for you in this tutorial.

Hit **Save Details**

Appearance

Click on **Set Appearance** to pick the 3D model this asset should use.

Like symbols, 3D models are arranged in a hierarchy. You'll find the model for the system in **ARDI / Examples / Boiler / Boiler Water System**.

From the **Skin** droplist, choose *Default*. A single 3D model might have more than one appearance, and this list lets you choose from several of them.

Press **Save** when you are done.

Linking Assets & Model Parts

When you are doing piece-by-piece models, your job would now be finished. But since we are doing an all-in-one model – where there are several different pieces that need to be linked to ARDI assets – there's one more step left.

When you return to *Placement & 3D*, you'll have new options available under the heading **Assign Asset To Model Part**.

There's a droplist of options next to each model part where you can choose the asset that matches the part. This list is made up of the assets inside Boiler #1. If you can't see the asset in the list, press the ellipsis (three dot) button and you'll be able to choose an asset from anywhere in your ARDI system.

Note that there are pieces of the model that do not need assets connected to them, such as the *Stand* and *Pipework*.

Assign Asset to Model Part

Model Part	Asset Selection
Stand:	[Not Bound]
Tank:	[Not Bound]
Pipework:	[Not Bound]
Water Isolation Valve:	[Not Bound] Boiler #1 Boiler #1 Vessel Boiler #1 Burner Boiler #1 Gas Isolator Boiler #1 Gas Pressure Tx Boiler #1 Gas Flow Meter Boiler #1 Secondary Gas Isolator Boiler #1 Gas Control Valve Boiler #1 Water Isolator Boiler #1 Water Temp Tx Boiler #1 Strainer Boiler #1 Water Filter Boiler #1 Water Control Valve Boiler #1 Vessel Pressure Sensor Boiler #1 Steam Drain
ValveHandle:	
Coarse Filter:	
Fine Filter:	
Water Control Valve:	
Water Temp Tx:	

Designing Your Own System

In this example, you've worked with *our* system, but now it's time to do something with one of yours.

There are a few notes about things to decide on when you're building your own ARDI database.

Decide On The Level Of Detail

It's important to figure out what level of detail you're interested in. That will largely depend on who is going to use the system, and how much information they need.

Each group of users will have their own needs, and ARDI can be extended to suit a wide range of them.

Does your electrical department want to know where assets get their power?

Do the maintenance people want to be able to identify assets by the maintenance IDs rather than the ones the engineers use?

Do you want to include all of the passive equipment, like manual valves, filters and perhaps even pipework? Or can you ignore all of these and concentrate on only the assets that have live data?

And in particular, how detailed are the relationships going to be? Since you build your own relationships, you could have relationships representing the individual 4-20mA signals in your plant, Ethernet and Wi-Fi connections, BUS cables and PLC cards. But unless these are actively needed by your users, it can be wasteful to put in excessive detail.

Working with Complex Assets

You'll also have to be consistent in your approach to complex machines.

Some assets – particularly those with electric motors such as pumps and conveyors – can be more complex than they seem.

An industrial pump has the actual pump body, the electric motor that turns the pump, and the electric drive that powers the pump.

In most applications, a single pump would be considered *two* assets, since the drive is often physically distant from the pump and motor, which are connected to one-another.

But if you're working with a maintenance crew that treats the motor separately from the rest of the pump, you might need to consider it *three* distinct pieces.

Or if you simply don't care about maintenance people and field technicians and are making your ARDI database purely for engineers or data analysis, you could simply treat the whole pump as a single asset.

Working with your Partner

Your ARDI Partner will usually be available to ask questions and provide ARDI support, and there are resources online to help you get the most out of your ARDI system. If you're stuck or unsure, just ask!

Appendix – Fixing Common Errors

Renaming An Asset

Nobody's perfect – we don't get the asset names right first time, every time.

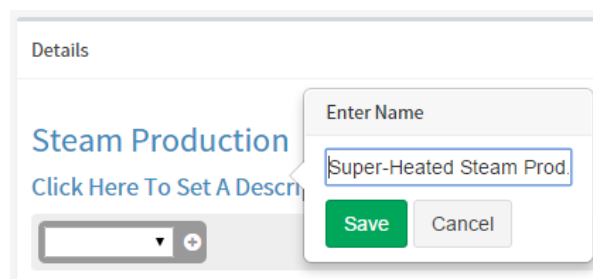
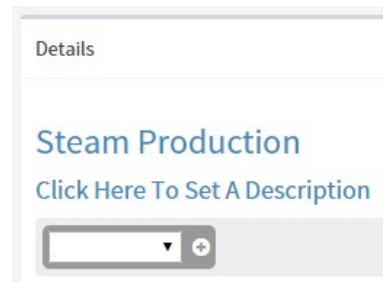
To change the name of an existing asset, you first find it (either via Search or the Location hierarchy) and go to the Details page for the asset.

At the top of the page you'll see the name and description of the item, shown in blue.

All you need to do to rename the item (or to change the description, or to change *any* property shown on the page) is to click on it.

A popup will appear, allowing you to change the value to something new.

Hit 'Save' to finalise your changes.

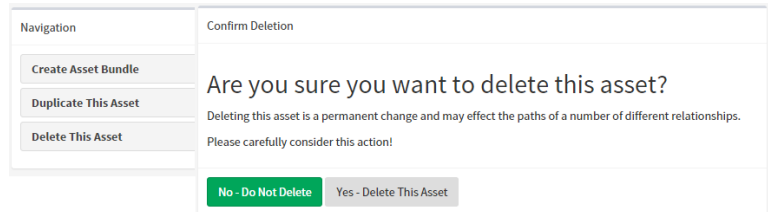


Deleting An Asset

The delete button is kept deep inside the asset Details, since inadvertently deleting an asset can cause serious issues in a live system.

In order to erase an existing asset, you'll need to go to the **Asset Tools** tab of the **Details** page.

Here you've got the option of deleting an asset permanently. Note that you will be asked to confirm the action.



You should make sure that your asset has been removed from all relationships. However, if still

connected when an asset is deleted ARDI will attempt to *fill the gap*.

For each relationship, all assets that were upstream from the asset will be connected to all assets downstream from it.

So if you began with the relationship...

A -> B -> C

...and then chose to delete 'B', you'll be left with

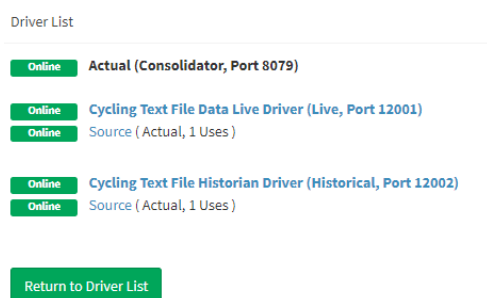
A -> C

This means that deleting a parent asset – such as Boiler #1 – will **not** delete any of the assets inside it or connected to it.

If you do need to delete multiple assets, this can be done using the Bulk Changes feature.

No Data Available

If you've got issues with your live or historical data, visit **Administration | Drivers | Status/Diagnostics** to see a summary of the state of your various ARDI drivers and sources.



In the example on the left, all of the elements are working correctly.

The items at the top of the list are the *consolidators*, which are responsible for capturing and distributing live data.

Below those are the *drivers*, and directly under each driver is a list of the *sources* that driver is servicing.

Driver Running, Source Stopped

If the driver is green but the source is yellow, you can click on the name of the source. A popup list will appear – you can choose **Reconnect** from that list.

If that fails, continue to the next step.

Driver Not Running

If the driver is yellow or the fix above didn't resolve problems with a source, press **Return To Driver List** and then click **Restart / Update**. This will unload and reload all of your ARDI drivers.

Cleaning Up Afterwards

When you've completed this tutorial and you'd like to return the database to its original state, you can visit **Administration | Databases** and hit the 'Reset' button. After confirming the action by typing in a confirmation code, this will reset your database to its empty state.